FeynCalc 3.0 Manual

Tools for Feynman diagram calculations

Rolf Mertig

August 26, 1998

Copyright ©1998
Mertig Research & Consulting
Oosterpark 77
NL – 1092AT Amsterdam
http://www.mertig.com

This notebook describes features of FeynCalc 3.0.5.1

■ Table of Contents

You can select the Cell bracket of this "Table of Contents" section and copy it to a new notebook which should be saved in the same directory as this one.
One possibility to jump back from a hyperlink is to use the "<<Back" button from the Help Browser.

ShortOverview
LoadFeynCalc
A0
Amplitudes
AnomalousDimension
Anti5
AntiCommutator
AntiQuarkField
Apart1
Apart2
Apart3
B00
B0
B11
B1
BackgroundGluonVertex
C0
Calc
CA
Cases2
CF
ChangeDimension
ChargeConjugationMatrixInv
ChargeConjugationMatrix
ChiralityProjector
Chisholm
Collect2
Combine
CommutatorExplicit
Commutator
ComplexConjugate
ComplexIndex
Contract
Convolute
CovariantD
D0
DataType
DeclareNonCommutative
DeltaFunction
DeltaFunctionPrime
Dimension
DiracGammaCombine
DiracGammaExpand
DiracGamma
DiracGammaT
DiracMatrix

DiracOrder
DiracReduce
DiracSigmaExplicit
DiracSigma
DiracSimplify
DiracSimplify2
DiracSlash
DiracTrace
DiracTrick
DotSimplify
EpsChisholm
EpsEvaluate
Epsilon
Eps
ExpandPartialD
ExpandScalarProduct
Explicit
ExplicitPartialD
Factor2
FAD
FCE
FCI
FermionSpinSum
FeynAmpDenominatorCombine
FeynAmpDenominator
FeynAmpDenominatorSimplify
FeynAmpDenominatorSplit
FeynAmp
FeynCalc2FORM
FeynCalcExternal
FeynCalcInternal
FeynRule
FieldStrength
FORM2FeynCalc
FourDivergence
FourLaplacian
FourVector
FreeQ2
FromTFi
FRH
FunctionalD
FVD
FV
GAD
GammaEpsilon
GammaExpand
GA
GhostPropagator
GHP
GaugeField
GluonGhostVertex
GluonPropagator
GluonVertex
GSD

GS
Hill
HypergeometricAC
HypergeometricIR
HypergeometricSE
HypExplicit
Integrate2
Integrate3
IntegrateByParts
InverseMellin
Isolate
Kummer
Lagrangian
LCD
LC
LeftPartialD
LeftRightPartialD
LeftRightPartialD2
LeviCivita
LorentzIndex
Map2
MetricTensor
MomentumCombine
MomentumExpand
Momentum
MTD
MT
Nielsen
NTerms
OneLoop
OneLoopSimplify
OPEDelta
OPEi, OPEj, OPEk, OPEl, OPEm, OPEn, OPEo
OPESumExplicit
OPESum
OPESumSimplify
Pair
PartialD
PaVe
PaVeOrder
PaVeReduce
PlusDistribution
Polarization
PolarizationSum
PolarizationVector
PowerSimplify
PropagatorDenominatorExplicit
PropagatorDenominator
QuantumField
QuarkField
QuarkGluonVertex
QuarkPropagator
RightPartialD
RussianTrick

ScalarProduct
ScalarProductCancel
SD
SelectFree
SelectNotFree
Series2
SimplifyDeltaFunction
SimplifyPolyLog
SOD
Solve2
SO
SPD
Spinor
SpinorUBar
SpinorU
SpinorVBar
SpinorV
SplittingFunction
SP
SumP
SumS
SumT
SUNDelta
SUND
SUNF
SUNIndex
SUNSimplify
SUNT
SUNTrace
Symmetrize
Tdec
Tensorfunction
TFi
TIDL
TID
ToDistribution
ToHypergeometric
ToLarin
ToTFi
TrickMandelstam
Trick
Tr
Twist2GluonOperator
Twist2QuarkOperator
Uncontract
UnDeclareNonCommutative
Write2
Evaluationtimeandmemoryusage

■ Short Overview

FeynCalc3.0 is a *Mathematica* 3.0 package for algebraic calculations in high energy physics. Tools are provided for Lorentz index contraction, Dirac algebra manipulation, color factor calculation, automatic Feynman rule derivation, general noncommutative algebras as well as various look–up tables for Feynman parameter integrals, convolutions and Feynman rules. Furthermore special translation facilities are provided to change the FeynCalc syntax to and from FORM syntax. Optimized FORTRAN generation is possible by using Isolate and Write2.

The more important functions for input of objects like p^μ, γ^ν, $g^{\mu\nu}$, $\varepsilon^{\mu\nu\rho\sigma}$, etc. (abbreviations in parentheses) are: DiracMatrix (GA), DiracSlash (GS), FourVector (FV), LeviCivita (LC), MetricTensor (MT), Spinor, SUNDelta, SUNF, ScalarProduct (SP).

The more important functions for manipulations are:

Contract, Tr, DiracSimplify, DiracTrick, Calc, Convolute, FeynRule , Integrate2 , FeynCalc2FORM , PaVeReduce , SimplifyPolyLog , SUNSimplify , Isolate , Collect2 , OneLoopSimplify , Series2 , ScalarProductCancel

FeynCalc3.0 is still compatible with FeynArts2.2 if the FeynArts function ToFA1Conventions is used on the generated amplitudes. The functions OneLoop and OneLoopSum for calculation of Standard Model (like) 1–loop diagrams still work in FeynCalc3.0 but they have not been developed much further compared to the FeynCalc1.0 version. While OneLoop and OneLoopSum still work fine for self energy, triangle and box diagrams not too complicated the reader should be aware of the package FORMCalc which is more effective for the adventurous researcher calculating thousands of diagrams involving four external gauge bosons. Notice however that FORMCalc does not reduce the Passarino–Veltman integrals to scalar integrals. Also FORMCalc is limited to the 't Hooft Feynman gauge. FeynCalc3.0 provides the possibility to reduce Passarino–Veltman integrals to scalar integrals algebraically by the function PaVeReduce .

On a PentiumII 300 Mhz computer with 64 MB of RAM running *Mathematica* 3.0 for Linux this notebook needs about 11 minutes to evaluate and less than 16 MB of memory.

FeynCalc automatically loads any tarcer*.mx file from the HighEnergyPhysics directory.
For a description of TARCER see hep–ph/9801383 or the published version in Computer Physics Communications 111 (1998) 265–273.

For more detailed information concerning FeynCalc functions you can investigate most of the source code in the FeynCalc directory in the HighEnergyPhysics directory. The location of the HighEnergyPhysics directory is stored upon installation in the fc.m file and you can get its value from the global variable $FeynCalcDirectory. A complete list of FeynCalc objects is stored in the variable $FeynCalcStuff. Not all functions are documented in this notebook. Some are either very special ones (e.g. for tools for 2–loop QCD diagrams originating in twist–2 OPE) or still experimental.

The γ^5 scheme used in OneLoop is the naive one, i.e., $\{\gamma^5, \gamma^\mu) = 0$ in 4 and D dimensions. See also ToLarin

The metric used is the one from Bjorken and Drell (+−−−).

For suggestions please email the author: Rolf@Mertig.com.
Notice that usually no support is given. However, especially if you find bugs, send a repeatable clear example to the above email address.

■ Load FeynCalc

```
absolutestarttime = AbsoluteTime[]
```

$3.113150656000000 \times 10^9$

This defines typesetting rules for q1 and q2:

```
q1 /: MakeBoxes[q1, fmt_] := SubscriptBox["q", "1"];
```

```
q2 /: MakeBoxes[q2, fmt_] := SubscriptBox["q", "2"];
```

Load FeynCalc:

```
<< HighEnergyPhysics`fc`
```

loading TARCER tarcer23.mx

FeynCalc 3.0.5.1. Type ?FeynCalc for help or visit www.mertig.com

loading TARCER tarcer23.mx

FeynCalc 3.0.5.0. Type ?FeynCalc for help or visit www.mertig.com

■ A0

■ Description

A0[*m*^2] is the Passarino - Veltman one – point integral A_0.

```
Options[A0]
```

{A0ToB0 → True}

■ See also: B0, C0, D0, PaVe.

■ Examples

By default any A_0 is expressed in terms of B_0's.

```
A0[m^2]
```

$B_0(0, m^2, m^2)\, m^2 + m^2$

```
A0[m^2, A0ToB0 → False]
```

$B_0(0, m^2, m^2)\, m^2 + m^2$

According to the rules of dimensional regularization $A_0(0)$ is set to 0.

```
A0[0]
```

0

```
A0[SmallVariable[M^2]]
```

0

■ Amplitudes

■ Description

Amplitudes[name] is a database of Feynman amplitudes.
Gauge → 1 means 't Hooft Feynman gauge; Polarization → 0 gives
unpolarized OPE – type amplitudes, Polarization → 1 the polarized ones.

```
Options[Amplitudes]
```

{Dimension → *D*, Gauge → 1, QuarkMass → 0, Polarization → 1}

■ See also: FeynAmp.

■ Examples

This is the amplitude of a gluon self energy diagram.

```
Amplitudes["se1g1"]
```

$$-\frac{g_s^2\, V^{\nu\rho\sigma}(-p,\ p-q,\ q)\, V^{\mu\alpha\beta}(p,\ q-p,\ -q)\, g^{\alpha\rho}\, g^{\beta\sigma}\, \delta_{cd}\, \delta_{ef}\, f_{ace}\, f_{bdf}}{(p-q)^2\, q^2}$$

```
Explicit[%]
```

$$-\frac{1}{(p-q)^2\, q^2}(g_s^2\, g^{\alpha\rho}\, g^{\beta\sigma}\, (-p^\alpha\, g^{\beta\mu} - q^\alpha\, g^{\beta\mu} + 2\, g^{\alpha\mu}\, p^\beta - g^{\alpha\mu}\, q^\beta - g^{\alpha\beta}\, p^\mu + 2\, g^{\alpha\beta}\, q^\mu)$$
$$(p^\gamma\, g^{\rho\sigma} - 2\, q^\gamma\, g^{\rho\sigma} + g^{\gamma\sigma}\, p^\rho + g^{\gamma\sigma}\, q^\rho - 2\, g^{\gamma\rho}\, p^\tau + g^{\gamma\rho}\, q^\tau)\, \delta_{cd}\, \delta_{ef}\, f_{ace}\, f_{bdf})$$

This is the amplitude for graph 6.2 from the paper Z. Phys C **70:** 637–654, 1996.

```
FeynAmp[q1, q2,      EpsEvaluate[Trick[Explicit[Amplitudes["gg2"]]]]]
```

$$\int \frac{d^D q_1}{(2\pi)^D} \int \frac{d^D q_2}{(2\pi)^D} \Big((2\, i\, g_s^4\, \epsilon^{\lambda_1\, \lambda_5\, \Delta\, q_2}\, (-p^\gamma\, g^{\lambda_7\, \lambda_{10}} + 2\, q_1{}^\nu\, g^{\lambda_7\, \lambda_{10}} + 2\, g^{\nu\lambda_{10}}\, p^{\lambda_7} - g^{\nu\lambda_{10}}\, q_1{}^{\lambda_7} - g^{\nu\lambda_7}\, p^{\lambda_{10}} - g^{\nu\lambda_7}\, q_1{}^{\lambda_{10}})$$
$$(p^\mu\, g^{\lambda_1\, \lambda_{11}} - 2\, q_2{}^\mu\, g^{\lambda_1\, \lambda_{11}} - 2\, g^{\mu\lambda_{11}}\, p^{\lambda_1} + g^{\mu\lambda_{11}}\, q_2{}^{\lambda_1} + g^{\mu\lambda_1}\, p^{\lambda_{11}} + g^{\mu\lambda_1}\, q_2{}^{\lambda_{11}})$$
$$(-2\, q_1{}^{\lambda_5}\, g^{\lambda_7\, \lambda_{12}} + q_2{}^{\lambda_5}\, g^{\lambda_7\, \lambda_{12}} + g^{\lambda_5\, \lambda_{12}}\, q_1{}^{\lambda_7} - 2\, g^{\lambda_5\, \lambda_{12}}\, q_2{}^{\lambda_7} + g^{\lambda_5\, \lambda_7}\, q_1{}^{\lambda_{12}} + g^{\lambda_5\, \lambda_7}\, q_2{}^{\lambda_{12}})$$
$$(-p^{\lambda_{10}}\, g^{\lambda_{11}\, \lambda_{12}} - q_1{}^{\lambda_{10}}\, g^{\lambda_{11}\, \lambda_{12}} + 2\, q_2{}^{\lambda_{10}}\, g^{\lambda_{11}\, \lambda_{12}} - g^{\lambda_{10}\, \lambda_{12}}\, p^{\lambda_{11}} +$$
$$2\, g^{\lambda_{10}\, \lambda_{12}}\, q_1{}^{\lambda_{11}} - g^{\lambda_{10}\, \lambda_{12}}\, q_2{}^{\lambda_{11}} + 2\, g^{\lambda_{10}\, \lambda_{11}}\, p^{\lambda_{12}} - g^{\lambda_{10}\, \lambda_{11}}\, q_1{}^{\lambda_{12}} - g^{\lambda_{10}\, \lambda_{11}}\, q_2{}^{\lambda_{12}})$$
$$(1-(-1)^m)\, (\Delta\cdot q_2)^{m-1}\, f_{ac_5c_{11}}\, f_{bc_7c_{10}}\, f_{c_5c_7c_{12}}\, f_{c_{10}c_{11}c_{12}})/(q_1^2\, q_2^2\, q_2^2\, (q_2-p)^2\, (q_1-q_2)^2\, (q_1-p)^2)\Big)$$

■ AnomalousDimension

■ Description

AnomalousDimension[name] is a database of anomalous dimensions of twist–2 operators.

```
Options[AnomalousDimension]
```

{Polarization → 1, Simplify → FullSimplify}

■ See also: SplittingFunction.

■ Examples

■ Polarized case

```
SetOptions[ AnomalousDimension, Polarization → 1]
```

{Polarization→ 1, Simplify→ FullSimplify}

$\gamma^{(0)}_{NS,qq}$ polarized

```
AnomalousDimension[gnsqq0]
```

$$C_F \left(8\, S_1(m-1) + \frac{4}{m} + \frac{4}{m+1} - 6\right)$$

$\gamma^{(0)}_{S,qg}$ polarized

```
AnomalousDimension[gsqg0]
```

$$\left(\frac{8}{m} - \frac{16}{m+1}\right) T_f$$

$\gamma^{(0)}_{S,gq}$ polarized

```
AnomalousDimension[gsgq0]
```

$$C_F \left(\frac{4}{m+1} - \frac{8}{m}\right)$$

$\gamma^{(0)}_{S,gg}$ polarized

```
AnomalousDimension[gsgg0]
```

$$\frac{8\, T_f}{3} + C_A \left(8\, S_1(m-1) - \frac{8}{m} + \frac{16}{m+1} - \frac{22}{3}\right)$$

$\gamma^{(0)}_{PS,qq}$ polarized

```
AnomalousDimension[gpsqq1]
```

$$16\, C_F \left(\frac{1}{m+1} + \frac{3}{(m+1)^2} + \frac{2}{(m+1)^3} - \frac{1}{m} - \frac{1}{m^2} + \frac{2}{m^3}\right) T_f$$

$\gamma^{(1)}_{NS,qq}$ polarized

AnomalousDimension[gnsqq1]

$$-\left(\frac{16\,S_1(m-1)}{m^2}+\frac{16\,S_1(m-1)}{(m+1)^2}+\frac{16\,S_2(m-1)}{m}+\frac{16\,S_2(m-1)}{m+1}-24\,S_2(m-1)+32\,S_{12}(m-1)+32\,S_{21}(m-1)+\right.$$
$$\left.\frac{32\,\tilde{S}_2(m-1)}{m}+\frac{32\,\tilde{S}_2(m-1)}{m+1}-32\,\tilde{S}_3(m-1)+64\,\tilde{S}_{12}(m-1)-\frac{40}{m}+\frac{40}{m+1}+\frac{16}{(m+1)^2}+\frac{8}{m^3}+\frac{40}{(m+1)^3}+3\right)$$
$$C_F^2-N_f\left(\frac{80}{9}\,S_1(m-1)-\frac{16}{3}\,S_2(m-1)-\frac{8}{9\,m}+\frac{88}{9(m+1)}-\frac{8}{3\,m^2}-\frac{8}{3(m+1)^2}-\frac{2}{3}\right)C_F-$$
$$C_A\left(-\frac{536}{9}\,S_1(m-1)+\frac{88}{3}\,S_2(m-1)-16\,S_3(m-1)-\frac{16\,\tilde{S}_2(m-1)}{m}-\frac{16\,\tilde{S}_2(m-1)}{m+1}+\right.$$
$$\left.16\,\tilde{S}_3(m-1)-32\,\tilde{S}_{12}(m-1)+\frac{212}{9\,m}-\frac{748}{9(m+1)}+\frac{44}{3\,m^2}-\frac{4}{3(m+1)^2}-\frac{16}{(m+1)^3}+\frac{17}{3}\right)C_F$$

$\gamma_{S,qg}^{(1)}$ polarized

AnomalousDimension[gsqg1]

$$8\,C_F\,T_f\left(\frac{2\,S_1^2(m-1)}{m}-\frac{4\,S_1^2(m-1)}{m+1}-\frac{2\,S_2(m-1)}{m}+\frac{4\,S_2(m-1)}{m+1}+\frac{14}{m}-\frac{19}{m+1}-\frac{1}{m^2}-\frac{8}{(m+1)^2}-\frac{2}{m^3}+\frac{4}{(m+1)^3}\right)+$$
$$16\,C_A\,T_f\left(-\frac{S_1^2(m-1)}{m}+\frac{2\,S_1^2(m-1)}{m+1}-\frac{2\,S_1(m-1)}{m^2}+\frac{4\,S_1(m-1)}{(m+1)^2}-\frac{S_2(m-1)}{m}+\right.$$
$$\left.\frac{2\,S_2(m-1)}{m+1}-\frac{2\,\tilde{S}_2(m-1)}{m}+\frac{4\,\tilde{S}_2(m-1)}{m+1}-\frac{4}{m}+\frac{3}{m+1}-\frac{3}{m^2}+\frac{8}{(m+1)^2}+\frac{2}{m^3}+\frac{12}{(m+1)^3}\right)$$

$\gamma_{S,gq}^{(1)}$ polarized

AnomalousDimension[gsgq1]

$$4\left(\frac{4\,S_1^2(m-1)}{m}-\frac{2\,S_1^2(m-1)}{m+1}-\frac{8\,S_1(m-1)}{m}+\frac{2\,S_1(m-1)}{m+1}+\frac{8\,S_1(m-1)}{m^2}-\frac{4\,S_1(m-1)}{(m+1)^2}+\right.$$
$$\left.\frac{4\,S_2(m-1)}{m}-\frac{2\,S_2(m-1)}{m+1}+\frac{15}{m}-\frac{6}{m+1}-\frac{12}{m^2}+\frac{3}{(m+1)^2}+\frac{4}{m^3}-\frac{2}{(m+1)^3}\right)C_F^2+$$
$$32\,T_f\left(-\frac{2\,S_1(m-1)}{3\,m}+\frac{S_1(m-1)}{3(m+1)}+\frac{7}{9\,m}-\frac{2}{9(m+1)}-\frac{2}{3\,m^2}+\frac{1}{3(m+1)^2}\right)C_F+$$
$$8\,C_A\left(-\frac{2\,S_1^2(m-1)}{m}+\frac{S_1^2(m-1)}{m+1}+\frac{16\,S_1(m-1)}{3\,m}-\frac{5\,S_1(m-1)}{3(m+1)}+\frac{2\,S_2(m-1)}{m}-\frac{S_2(m-1)}{m+1}+\right.$$
$$\left.\frac{4\,\tilde{S}_2(m-1)}{m}-\frac{2\,\tilde{S}_2(m-1)}{m+1}-\frac{56}{9\,m}-\frac{20}{9(m+1)}+\frac{28}{3\,m^2}-\frac{38}{3(m+1)^2}-\frac{4}{m^3}-\frac{6}{(m+1)^3}\right)C_F$$

$\gamma_{S,gg}^{(1)}$ polarized

```
AnomalousDimension[gsgg1]
```

$$4\left(\frac{8\,S_1(m-1)}{m^2}-\frac{16\,S_1(m-1)}{(m+1)^2}+\frac{134}{9}\,S_1(m-1)+\frac{8\,S_2(m-1)}{m}-\frac{16\,S_2(m-1)}{m+1}+\right.$$
$$\left.4\,S_3(m-1)-8\,S_{12}(m-1)-8\,S_{21}(m-1)+\frac{8\,\tilde{S}_2(m-1)}{m}-\frac{16\,\tilde{S}_2(m-1)}{m+1}+4\,\tilde{S}_3(m-1)-\right.$$
$$\left.8\,\tilde{S}_{12}(m-1)-\frac{107}{9\,m}+\frac{241}{9\,(m+1)}+\frac{58}{3\,m^2}-\frac{86}{3\,(m+1)^2}-\frac{8}{m^3}-\frac{48}{(m+1)^3}-\frac{16}{3}\right)C_A^2+$$
$$32\,T_f\left(-\frac{5}{9}\,S_1(m-1)+\frac{14}{9\,m}-\frac{19}{9\,(m+1)}-\frac{1}{3\,m^2}-\frac{1}{3\,(m+1)^2}+\frac{1}{3}\right)C_A+$$
$$8\,C_F\left(-\frac{10}{m+1}+\frac{2}{(m+1)^2}+\frac{4}{(m+1)^3}+1+\frac{10}{m}-\frac{10}{m^2}+\frac{4}{m^3}\right)T_f$$

$\gamma_{S,gg}^{(1)}$ polarized (different representation)

```
AnomalousDimension[GSGG1]
```

$$4\left(-\frac{m(m(m(m(48\,m(m+3)+469)+698)+7)+258)+144}{9\,m^2\,(m+1)^3}+\right.$$
$$\left.\frac{8\,S_2'\left(\frac{m}{2}\right)}{m(m+1)}-S_3'\left(\frac{m}{2}\right)+\frac{2\,(m(67\,m(m+1)^2+144)+72)\,S_1(m)}{9\,m^2\,(m+1)^2}-4\,S_2'\left(\frac{m}{2}\right)S_1(m)+8\,\tilde{S}(m)\right)C_A^2+$$
$$32\,T_f\left(\frac{m(m+1)\,(3\,m(m+1)+13)-3}{9\,m^2\,(m+1)^2}-\frac{5\,S_1(m)}{9}\right)C_A+\frac{8\,C_F\,(m(m(m(m(m(m+3)+5)+1)-8)+2)+4)\,T_f}{m^3\,(m+1)^3}$$

Check that all odd moments give the same for the two representations of $\gamma_{S,gg}^{(1)}$.

```
Table[(%-%%)/.OPEm->ij, {ij,1,17,2}]
```

```
{0, 0, 0, 0, 0, 0, 0, 0, 0}
```

■ Anti5

■ Description

Anti5[exp] anticommutes all γ^5 in exp to the right. Anti5[exp,n] anticommutes all γ^5 n times to the right. Anti5[exp,–n] anticommutes all γ^5 n times to the left.

The naive γ^5 scheme is used.

■ See also: DiracOrder, DiracSimplify, DiracTrick.

■ Examples

```
DiracMatrix[5, μ]
```

$\gamma^5\,\gamma^\mu$

```
Anti5[%]
```

$-(\gamma^{\mu}\ \gamma^{5})$

```
Anti5[%, -1]
```

$\gamma^{5}\ \gamma^{\mu}$

```
DiracMatrix[5, α, β, γ, δ]
```

$\gamma^{5}\ \gamma^{\alpha}\ \gamma^{\beta}\ \gamma^{\gamma}\ \gamma^{\delta}$

```
Anti5[%, 2]
```

$\gamma^{\alpha}\ \gamma^{\beta}\ \gamma^{5}\ \gamma^{\gamma}\ \gamma^{\delta}$

```
Anti5[%%, Infinity]
```

$\gamma^{\alpha}\ \gamma^{\beta}\ \gamma^{\gamma}\ \gamma^{\delta}\ \gamma^{5}$

```
Anti5[%, -Infinity]
```

$\gamma^{5}\ \gamma^{\alpha}\ \gamma^{\beta}\ \gamma^{\gamma}\ \gamma^{\delta}$

In the naive γ^5 – scheme D – dimensional γ – matrices anticommute with γ^5.

```
GAD[5, μ]
```

$\gamma^{5}\ \gamma^{\mu}$

```
Anti5[%]
```

$-(\gamma^{\mu}\ \gamma^{5})$

Setting the global variable $FCT to False restores the default typesetting rules for the noncommutative multiplication operator Dot[].

```
$FCT = False;
```

```
GAD[5, μ]
```

$\gamma^{5}.\gamma^{\mu}$

```
Anti5[%]
```

$-\gamma^{\mu}.\gamma^{5}$

```
$FCT = True;
```

■ AntiCommutator

■ Description

AntiCommutator[x, y] = c defines the anti–commutator of the noncommuting objects x and y.

■ See also: Commutator, CommutatorExplicit, DeclareNonCommutative, DotSimplify.

■ Examples

This declares a and b as noncommutative variables.

```
DeclareNonCommutative[a, b]
```

```
AntiCommutator[a, b]
```

$\{a, b\}$

```
CommutatorExplicit[%]
```

$a b + b a$

```
CommutatorExplicit[AntiCommutator[a + b, a - 2 b ]]
```

$(a - 2\,b)\,(a + b) + (a + b)\,(a - 2\,b)$

```
DotSimplify[AntiCommutator[a + b, a - 2 b ]]
```

$2\,a^2 - 4\,b^2 - a b - b a$

```
DeclareNonCommutative[c, d, c̃, d̃]
```

Defining {c,d} = z results in replacements of c.d by z–d.c.

```
AntiCommutator[c, d] = z
```

z

```
DotSimplify[ d . c . d ]
```

$d z - d^2\,c$

```
AntiCommutator[d̃, c̃] = z̃
```

$\tilde{z}$

```
DotSimplify[ d̃ . c̃ . d̃ ]
```

$\tilde{d}z - \tilde{c}\,\tilde{d}^2$

```
UnDeclareNonCommutative[a, b, c, d, c̃, d̃]
```

```
AntiCommutator[c, d] = .
```

```
AntiCommutator[d̃, c̃] = .
```

■ AntiQuarkField

■ Description

AntiQuarkField is the name of a fermionic field. AntiQuarkField is just a name with no functional properties. Only typeset rules are attached.

■ See also: QuantumField, QuarkField.

■ Examples

```
AntiQuarkField
```

$\overline{\psi}$

■ Apart1

■ Description

Apart1[expr, x] is equivalent to Apart[expr, x], but it is faster than Apart if explicit complex numbers (I's) are in exp.

■ See also: Apart2, Apart3.

■ Examples

```
t = 1/4*(1 - x)^4*x*(21 + 13*x + 3*x^2) -
1/24*11*I*Pi*(1 - x)^4*x*(21 + 13*x + 3*x^2) +
((1 - x)^4*x^2*(21 + 13*x + 3*x^2))/(36*(-1 + x)^4) -
(I*Pi*(1 - x)^4*x^2*(21 + 13*x + 3*x^2))/
(12*(-1 + x)^4) - (7*(1 - x)^4*x^3*(21 + 13*x + 3*x^2))/ (48*(-1 + x)^4) + (3*I*Pi*(1
- x)^4*x^3*
(21 + 13*x + 3*x^2))/(8*(-1 + x)^4) +
(11*(1 - x)^4*x^4*(21 + 13*x + 3*x^2))/(24*(-1 + x)^4) -
(3*I*Pi*(1 - x)^4*x^4*(21 + 13*x + 3*x^2))/
(4*(-1 + x)^4) - (49*(1 - x)^4*x^5*(21 + 13*x + 3*x^2))/
(144*(-1 + x)^4) + (11*I*Pi*(1 - x)^4*x^5*
(21 + 13*x + 3*x^2))/(24*(-1 + x)^4) +
1/4*(1 - x)^4*x*(21 + 13*x + 3*x^2)*Zeta2 +
11/24*(1 - x)^4*x*(21 + 13*x + 3*x^2)*Log[x] +
1/4*I*Pi*(1 - x)^4*x*(21 + 13*x + 3*x^2)*Log[x] +
((1 - x)^4*x^2*(21 + 13*x + 3*x^2)*Log[x])/
(12*(-1 + x)^4) - (3*(1 - x)^4*x^3*(21 + 13*x + 3*x^2)*
Log[x])/(8*(-1 + x)^4) +
(3*(1 - x)^4*x^4*(21 + 13*x + 3*x^2)*Log[x])/
(4*(-1 + x)^4);
```

```
Timing[a1 = Apart1[t, x];]
```

{0.72 Second, Null}

```
Timing[a2 = Apart[t, x];]
```

{3.53 Second, Null}

```
a1 - a2 // Expand
```

0

```
Clear[t, a1, a2]
```

■ Apart2

■ Description

Apart2[expr] partial fractions FeynAmpDenominator (and FAD's).

■ See also: FAD, FeynAmpDenominator.

■ Examples

```
FAD[{q, m}, {q, M}, q - p]
```

$$\frac{1}{([(q-p)^2])([q^2-m^2])([q^2-M^2])}$$

```
Apart2[%]
```

$$\frac{1}{(m^2-M^2)((q^2-m^2)(q-p)^2)} - \frac{1}{(m^2-M^2)((q^2-M^2)(q-p)^2)}$$

```
InputForm[FCE[%]]
```

```
FAD[{q, m}, -p + q]/(m^2 - M^2) -
  FAD[{q, M}, -p + q]/(m^2 - M^2)
```

■ Apart3

■ Description

Apart3[expr,x] is equivalent to Map2[Factor2,Collect2[Apart1[expr,x],x]].

■ See also: Apart1.

■ Examples

```
t = 1 / 4 (1 - x) ^4 x (21 + 13 x + 3 x^2) - 11 / 24 I π (1 - x) ^4 x (21 + 13 x + 3 x^2) +
   ((1 - x) ^4 x^2 (21 + 13 x + 3 x^2)) / (36 (-1 + x) ^4) -
   (I π (1 - x) ^4 x^2 (21 + 13 x + 3 x^2)) / (12 (-1 + x) ^4) -
   (7 (1 - x) ^4 x^3 (21 + 13 x + 3 x^2)) / (48 (-1 + x) ^4) +
   (3 I π (1 - x) ^4 x^3 (21 + 13 x + 3 x^2)) / (8 (-1 + x) ^4) +
   (11 (1 - x) ^4 x^4 (21 + 13 x + 3 x^2)) / (24 (-1 + x) ^4) -
   (3 I π (1 - x) ^4 x^4 (21 + 13 x + 3 x^2)) / (4 (-1 + x) ^4) -
   (49 (1 - x) ^4 x^5 (21 + 13 x + 3 x^2)) / (144 (-1 + x) ^4) +
   (11 I π (1 - x) ^4 x^5 (21 + 13 x + 3 x^2)) / (24 (-1 + x) ^4) +
   1 / 4 (1 - x) ^4 x (21 + 13 x + 3 x^2) Zeta2 + 11 / 24 (1 - x) ^4 x (21 + 13 x + 3 x^2) Log[x] +
   1 / 4 I π (1 - x) ^4 x (21 + 13 x + 3 x^2) Log[x] +
   ((1 - x) ^4 x^2 (21 + 13 x + 3 x^2) Log[x]) / (12 (-1 + x) ^4) -
   (3 (1 - x) ^4 x^3 (21 + 13 x + 3 x^2) Log[x]) / (8 (-1 + x) ^4) +
   (3 (1 - x) ^4 x^4 (21 + 13 x + 3 x^2) Log[x]) / (4 (-1 + x) ^4);

Timing[a1 = Apart3[t, x];]

{1.05 Second, Null}

Timing[a2 = Apart[t, x];]

{3.52 Second, Null}
```

```
a1 - a2 // Expand
```

0

```
Clear[t, a1, a2]
```

■ B00

■ Description

B00[pp, m_1^2, m_2^2] is the Passarino-Veltman B_{00} -function, i.e., the coefficient function of g^(μ ν). All arguments are scalars and have dimension mass^2.

```
Options[B00]
```

{BReduce → True}

■ See also: B0, B1, PaVe.

■ Examples

Remember that SP[p] is a short hand input for ScalarProduct[p,p], i.e. p^2.

```
SP[p]
```

p^2

```
B00[SP[p], m^2, M^2]
```

$$\frac{1}{3} B_0(p^2, m^2, M^2)\, m^2 + \frac{1}{18}(3\, m^2 + 3\, M^2 - p^2) + \frac{1}{6}\left(B_0(0, M^2, M^2)\, M^2 + M^2 + \left(\frac{(M^2 - m^2)\,(B_0(p^2, m^2, M^2) - B_0(0, m^2, M^2))}{2\, p^2} - \frac{1}{2} B_0(p^2, m^2, M^2)\right)(m^2 - M^2 + p^2)\right)$$

```
B00[SP[p], m^2, m^2]
```

$$\frac{1}{3} B_0(p^2, m^2, m^2)\, m^2 + \frac{1}{18}(6\, m^2 - p^2) + \frac{1}{6}\left(B_0(0, m^2, m^2)\, m^2 + m^2 - \frac{1}{2} B_0(p^2, m^2, m^2)\, p^2\right)$$

```
B00[SP[p], m^2, M^2, BReduce → False]
```

$B_{00}(p^2,\ m^2,\ M^2)$

```
B00[0, m^2, m^2]
```

$$\frac{1}{3} B_0(0, m^2, m^2)\, m^2 + \frac{m^2}{3} + \frac{1}{6}(B_0(0, m^2, m^2)\, m^2 + m^2)$$

```
B00[SmallVariable[M^2], m^2, m^2]
```

$$\frac{1}{3} B_0(M^2, m^2, m^2)\, m^2 + \frac{m^2}{3} + \frac{1}{6} (B_0(0, m^2, m^2)\, m^2 + m^2)$$

■ B0

■ Description

B0[p^2, m_1^2, m_2^2] is the Passarino-Veltman two-point integral B_0. All arguments are scalars and have dimension mass^2. If the option BReduce is set to True certain B0's are reduced to A0's.
Setting the option B0Unique to True simplifies B0[a,0,a] and B0[0,0,a].

```
Options[B0]
```

{BReduce→ False, B0Unique→ False, B0Real→ False}

■ See also: B1, B00, B11, PaVe.

■ Examples

```
B0[SP[p, p], m^2, m^2]
```

$B_0(p^2, m^2, m^2)$

```
SetOptions[B0, B0Unique → True, B0Real → True];
```

```
B0[0, 0, m^2]
```

$B_0(0, m^2, m^2) + 1$

```
B0[m^2, 0, m^2]
```

$B_0(0, m^2, m^2) + 2$

```
B0[0, m^2, m^2]
```

$B_0(0, m^2, m^2)$

■ B11

■ Description

B11[pp, m_1^2, m_2^2] is the Passarino-Veltman B_{11}-function, i.e., the coefficient function of p^μ p^ν. All arguments are scalars and have dimension mass^2.

```
Options[B11]
```

{BReduce→ True}

■ See also: B0, B00, B1, PaVe.

■ Examples

Remember that SP[p] is a shorthand input for ScalarProduct[p,p], i.e. p^2.

```
SP[p]
```

p^2

```
B11[SP[p], m^2, M^2]
```

$$\frac{1}{3\,p^2}\left(-B_0(p^2,\,m^2,\,M^2)\,m^2+M^2+M^2\,B_0(0,\,M^2,\,M^2)+\frac{1}{6}(-3\,m^2-3\,M^2+p^2)-\right.$$
$$\left.2\left(\frac{(M^2-m^2)\,(B_0(p^2,\,m^2,\,M^2)-B_0(0,\,m^2,\,M^2))}{2\,p^2}-\frac{1}{2}\,B_0(p^2,\,m^2,\,M^2)\right)(m^2-M^2+p^2)\right)$$

```
B11[SP[p], m^2, m^2]
```

$$\frac{B_0(0,\,m^2,\,m^2)\,m^2+m^2+\frac{1}{6}\,(p^2-6\,m^2)+B_0(p^2,\,m^2,\,m^2)\,(p^2-m^2)}{3\,p^2}$$

```
SetOptions[B11, BReduce → False]
```

{BReduce→ False}

```
B11[SP[p], m^2, M^2]
```

$B_{11}(p^2,\;m^2,\;M^2)$

```
SetOptions[B11, BReduce → True]
```

{BReduce→ True}

```
B11[0, m^2, m^2]
```

$$\frac{1}{3}\,B_0(0,\,m^2,\,m^2)$$

```
B11[SmallVariable[M^2], m^2, m^2]
```

$$\frac{1}{3}\,B_0(M^2,\,m^2,\,m^2)$$

■ B1

■ Description

B1 [p^2, m_1^2, m_2^2] is the Passarino-Veltman B_1 – function. All arguments are scalars and have dimension mass^2.

```
Options[B1]
```

{BReduce → True}

■ See also: B0, B00, B11, PaVe, PaVeReduce.

■ Examples

```
B1[SP[p], m^2, M^2]
```

$$\frac{(M^2 - m^2)\,(B_0(p^2, m^2, M^2) - B_0(0, m^2, M^2))}{2\,p^2} - \frac{1}{2}\,B_0(p^2, m^2, M^2)$$

```
B1[SP[p], m^2, M^2, BReduce → False]
```

$B_1(p^2, m^2, M^2)$

```
SetOptions[B1, BReduce → True];
```

```
B1[SP[p], m^2, m^2]
```

$$-\frac{1}{2}\,B_0(p^2, m^2, m^2)$$

```
B1[m^2, m^2, 0]
```

$$\frac{1}{2}\,(-B_0(0, m^2, m^2) - 2) - \frac{1}{2}$$

```
B1[0, 0, m^2]
```

$$\frac{1}{2}\,(-B_0(0, m^2, m^2) - 1) + \frac{1}{4}$$

```
B1[pp, SmallVariable[m_e^2], m_2^2]
```

$$\frac{(B_0(\text{pp}, m_e^2, m_2^2) - B_0(0, m_e^2, m_2^2))\,m_2^2}{2\,\text{pp}} - \frac{1}{2}\,B_0(\text{pp}, m_e^2, m_2^2)$$

```
B1[SmallVariable[m_e^2], SmallVariable[m_e^2], 0]
```

$$\frac{1}{2}\,(-B_0(0, m_e^2, m_e^2) - 2) - \frac{1}{2}$$

■ BackgroundGluonVertex

■ Description

BackgroundGluonVertex[{p,μ,a},{q,ν,b}, {k,λ,c}] or
BackgroundGluonVertex[p,μ,a, q,ν,b, k,λ,c] yields the 3–gluon vertex in the background field gauge, where the first set of arguments corresponds to the external background field.
BackgroundGluonVertex[{p,μ,a},{q,ν,b}, {k,λ,c}, {s,σ,d}] or
BackgroundGluonVertex[{μ,a},{ν,b}, {λ,c}, {σ,d}] or
BackgroundGluonVertex[p,μ,a q,ν,b, k,λ,c, s,σ,d] or
BackgroundGluonVertex[μ,a, ν,b, λ,c, σ,d] yields the 4–gluon vertex, with {p,μ,a} and {k,λ,c} denoting the external background fields.
The gauge, dimension and the name of the coupling constant are determined by the options Gauge, Dimension and CouplingConstant.
The Feynman rules are taken from L. Abbot **NPB** 185 (1981), 189–203; except that all momenta are incoming. Note that Abbots coupling constant convention is consistent with the defaults setting of GluonVertex.

```
Options[BackgroundGluonVertex]
```

{Dimension → D, CouplingConstant → g_s, Gauge → 1}

■ See also: GluonVertex.

■ Examples

```
BackgroundGluonVertex[{p, μ, a}, {q, ν, b}, {k, λ, c}]
```

$$g_s\,(((-k+p-q)^{\lambda})\,g^{\mu\nu}+g^{\lambda\nu}\,((q-k)^{\mu})+g^{\lambda\mu}\,((k-p+q)^{\nu}))\,f_{abc}$$

```
BackgroundGluonVertex[{p, μ, a}, {q, ν, b}, {k, λ, c}, {s, σ, d}]
```

$$-i\,g_s^2\,((g^{\lambda\sigma}\,g^{\mu\nu}-g^{\lambda\nu}\,g^{\mu\sigma}-g^{\lambda\mu}\,g^{\nu\sigma})\,f_{adu6}\,f_{bcu6}+(g^{\lambda\sigma}\,g^{\mu\nu}-g^{\lambda\nu}\,g^{\mu\sigma})\,f_{acu6}\,f_{bdu6}+(g^{\lambda\sigma}\,g^{\mu\nu}-g^{\lambda\nu}\,g^{\mu\sigma}+g^{\lambda\mu}\,g^{\nu\sigma})\,f_{abu6}\,f_{cdu6})$$

```
BackgroundGluonVertex[{p, μ, a}, {q, ν, b}, {k, λ, c}, Gauge → α]
```

$$g_s\left(\left(\left((p-q)-\frac{k}{\alpha}\right)^{\lambda}\right)g^{\mu\nu}+g^{\lambda\nu}\,((q-k)^{\mu})+g^{\lambda\mu}\left(\left((k-p)+\frac{q}{\alpha}\right)^{\nu}\right)\right)f_{abc}$$

```
BackgroundGluonVertex[{p, μ, a}, {q, ν, b}, {k, λ, c}, {s, σ, d}, Gauge → α]
```

$$-i\,g_s^2\left(\left(g^{\lambda\sigma}\,g^{\mu\nu}-\frac{g^{\lambda\nu}\,g^{\mu\sigma}}{\alpha}-g^{\lambda\mu}\,g^{\nu\sigma}\right)f_{adu7}\,f_{bcu7}+(g^{\lambda\sigma}\,g^{\mu\nu}-g^{\lambda\nu}\,g^{\mu\sigma})\,f_{acu7}\,f_{bdu7}+\left(\frac{g^{\lambda\sigma}\,g^{\mu\nu}}{\alpha}-g^{\lambda\nu}\,g^{\mu\sigma}+g^{\lambda\mu}\,g^{\nu\sigma}\right)f_{abu7}\,f_{cdu7}\right)$$

■ C0

■ Description

C0[p10, p12, p20, m1^2, m2^2, m3^2] is the scalar Passarino-Veltman C_0 function. The convention for the arguments is that if the denominator of the integrand has the form ([q^2 - m1^2] [(q+p1)^2 - m2^2] [(q+p2)^2 - m3^2]), the first three arguments of C0 are the scalar products p10 = p1^2, p12 = (p1 - p2).(p1 - p2)/2, p20 = p2^2.

■ See also: B0, D0, PaVe, PaVeOrder.

■ Examples

```
C0[a, b, c, m12, m22, m32]
```

$C_0(a, b, c, m12, m22, m32)$

```
C0[b, a, c, m32, m22, m12] // PaVeOrder
```

$C_0(a, b, c, m12, m22, m32)$

```
PaVeOrder[C0[b, a, c, m32, m22, m12], PaVeOrderList -> {c, a}]
```

$C_0(b, c, a, m22, m32, m12)$

■ Calc

■ Description

Calc[exp] performs several basic simplifications.
Calc[exp] is the same as
DotSimplify[DiracSimplify[Contract[DiracSimplify[Explicit[
SUNSimplify[Trick[exp], Explicit -> False]]]]]

■ See also: DiracSimplify, DiracTrick, Trick.

■ Examples

This calculates $\gamma^\mu \gamma_\mu$ in 4 dimensions and g^ν_ν in D dimensions.

```
Calc[{GA[μ, μ], MTD[ν, ν]}]
```

{4, *D*}

This simplifies $f_{abc} f_{abe}$.

```
Calc[SUNF[a, b, c] SUNF[a, b, e]]
```

$C_A \delta_{ce}$

```
FV[p + r, μ] MT[μ, ν] FV[q - p, ν]
```

$((q-p)^{\nu})\,((p+r)^{\mu})\,g^{\mu\nu}$

```
Calc[%]
```

$-p^2 + p\cdot q - p\cdot r + q\cdot r$

```
GluonVertex[p, 1, q, 2, -p - q, 3]
```

$g_s\, V^{\text{li1 li2 li3}}(p,\ q,\ -p-q)\, f_{\text{ci1ci2ci3}}$

```
Calc[% FVD[p, li1] FVD[q, li2] FVD[-p - q, li3]]
```

0

■ CA

■ Description

CA is one of the Casimir operators of SU(N) (CA = N).

■ See also: CF, SUNSimplify.

■ Examples

```
CA
```

C_A

```
SUNSimplify[CA, SUNNToCACF → False]
```

N

```
SUNN
```

N

■ Cases2

■ Description

Cases2[expr,f] returns a list of all objects in expr with head f.
Cases2[expr,f] is equivalent to Cases2[{expr},f[___],Infinity]//Union.
Cases2[expr,f, g, ...] or Cases2[expr,{f,g, ...}] is equivalent to
Cases[{expr},f[___]| g[___] ...] .

■ Examples

```
Cases2[f[a] + f[b]^2 + f[c, d], f]
```

$\{f(a),\ f(b),\ f(c, d)\}$

```
Cases2[Sin[x] Sin[y - z] + g[y], Sin, g]
```

$\{g(y),\ \sin(x),\ \sin(y - z)\}$

```
Cases2[Sin[x] Sin[y - z] + g[x] + g[a, b, c], {Sin, g}]
```

$\{g(x),\ g(a, b, c),\ \sin(x),\ \sin(y - z)\}$

```
Cases2[DiracSlash[p].DiracSlash[q] + ScalarProduct[p, p], Dot]
```

$\{\gamma \cdot p \gamma \cdot q\}$

■ CF

■ Description

CF is one of the Casimir operators of SU(N) (CF = (N^2–1)/(2N)).

■ See also: CA, SUNSimplify.

■ Examples

```
CF
```

C_F

```
SUNSimplify[CF, SUNNToCACF -> False]
```

$$\frac{N^2 - 1}{2\,N}$$

```
SUNN
```

N

```
SUNSimplify[CF, SUNNToCACF -> True]
```

C_F

■ ChangeDimension

■ Description

ChangeDimension[exp, dim] changes all LorentzIndex and Momentum in exp to dimension dim (and also Levi–Civita–tensors, Dirac slashes and Dirac matrices).

■ See also: LorentzIndex, Momentum, DiracGamma, Eps.

■ Examples

Remember that LorentzIndex[μ, 4] is simplified to LorentzIndex[μ] and Momentum[p, 4] to Momentum[p]. Thus the following objects are defined in four dimensions.

```
{LorentzIndex[μ], Momentum[p]}
```

$\{\mu, p\}$

```
ChangeDimension[%, D]
```

$\{\mu, p\}$

```
% // InputForm
```

```
{LorentzIndex[μ, D], Momentum[p, D]}
```

This changes all non–4–dimension objects to 4–dimensional ones.

```
ChangeDimension[%%, 4] // InputForm
```

```
{LorentzIndex[μ], Momentum[p]}
```

Consider the following list of 4– and D–dimensional object.

```
{GA[μ, ν] MT[μ, ν], GAD[μ, ν] MTD[μ, ν] f[D]}
```

$\{(\gamma^\mu \gamma^\nu)\, g^{\mu\nu}, (\gamma^\mu \gamma^\nu)\, f(D)\, g^{\mu\nu}\}$

```
DiracTrick[Contract[%]]
```

$\{4, D\, f(D)\}$

```
DiracTrick[Contract[ChangeDimension[%%, n]]]
```

$\{n, n\, f(D)\}$

Any explicit occurence of D (like in f(D)) is not replaced by ChangeDimension.

```
SetOptions[LeviCivita, Dimension → D];
```

The option Dimension of Eps must be changed too, since with the default setting Dimension->4 the arguments of Eps are automatically changed to 4 dimensions.

```
SetOptions[Eps, Dimension → D];

a1 = LeviCivita[μ, ν, ρ, σ]
```

$\epsilon^{\mu\nu\rho\sigma}$

```
a2 = ChangeDimension[a1, 4]
```

$\epsilon^{\mu\nu\rho\sigma}$

```
Contract[a1^2]  // Factor2
```

$(1-D)\,(2-D)\,(3-D)\,D$

```
Contract[a2^2]
```

-24

```
SetOptions[Eps, Dimension → 4];

Clear[a1, a2];
```

■ ChargeConjugationMatrixInv

■ Description

ChargeConjugationMatrixInv is the inverse of ChargeConjugationMatrix. It is substituted immediately by –ChargeConjugationMatrix.

■ See also: ChargeConjugationMatrix.

■ Examples

```
ChargeConjugationMatrix.DiracMatrix[μ].ChargeConjugationMatrixInv
```

$C\gamma^{\mu}\,(-C)$

```
Calc[%]
```

$-\gamma_\mu^T$

```
ChargeConjugationMatrix.DiracGamma[5].ChargeConjugationMatrixInv
```

$C\gamma^5\,(-C)$

```
Calc[%]
```

DiracGamma(5)

```
ChargeConjugationMatrix.ChargeConjugationMatrixInv
```

$C\,(-C)$

```
% // Calc
```

1

```
ChargeConjugationMatrixInv^2
```

-1

■ ChargeConjugationMatrix

■ Description

ChargeConjugationMatrix denotes the charge conjugation matrix C.

■ See also: ChargeConjugationMatrixInv.

■ Examples

```
ChargeConjugationMatrix.DiracMatrix[μ].ChargeConjugationMatrixInv
```

$C\gamma^\mu\,(-C)$

```
Calc[%]
```

$-\gamma_\mu^T$

```
ChargeConjugationMatrix.DiracGamma[5].ChargeConjugationMatrixInv
```

$C\gamma^5\,(-C)$

```
Calc[%]
```

DiracGamma(5)

```
ChargeConjugationMatrix^2
```

-1

■ ChiralityProjector

■ Description

ChiralityProjector[+1] denotes $1/2\,(1+\gamma^5)$.
ChiralityProjector[–1] denotes $1/2\,(1+\gamma^5)$.

■ See also: DiracGamma, DiracMatrix, FCI.

■ Examples

```
{ChiralityProjector[+1], ChiralityProjector[-1]}
```

$\{\gamma^6, \gamma^7\}$

```
FCI[%]
```

$\{\gamma^6, \gamma^7\}$

```
DiracSimplify[%, DiracSubstitute67 -> True]
```

$\left\{\frac{\gamma^5}{2}+\frac{1}{2}, \frac{1}{2}-\frac{\gamma^5}{2}\right\}$

■ Chisholm

■ Description

Chisholm[x] substitutes products of three Dirac matrices or slashes by the Chisholm identity.

■ See also: EpsChisholm.

■ Examples

```
Chisholm[GA[α, β, μ, ν]]
```

$-i\,(\gamma^{\$MU\$505}\,\gamma^{\nu}\,\gamma^5)\,\epsilon^{\alpha\,\beta\,\mu\,\$MU\$505}+(\gamma^{\mu}\,\gamma^{\nu})\,g^{\alpha\,\beta}-(\gamma^{\beta}\,\gamma^{\nu})\,g^{\alpha\,\mu}+(\gamma^{\alpha}\,\gamma^{\nu})\,g^{\beta\,\mu}$

```
t1 = DiracMatrix[μ, ν, ρ]
```

$\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{\rho}$

```
t2 = Chisholm[t1]
```

$i\,(\gamma^{\$MU\$506}\gamma^5)\,\epsilon^{\mu\nu\rho\$MU\$506}+\gamma^{\rho}\,g^{\mu\nu}-\gamma^{\nu}\,g^{\mu\rho}+\gamma^{\mu}\,g^{\nu\rho}$

The MU variables are unique indices.

```
Calc[t1.t1]
```

16

```
t3 = Chisholm[t1]
```

$i\,(\gamma^{\$MU\$507}\gamma^5)\,\epsilon^{\mu\nu\rho\$MU\$507}+\gamma^{\rho}\,g^{\mu\nu}-\gamma^{\nu}\,g^{\mu\rho}+\gamma^{\mu}\,g^{\nu\rho}$

```
Calc[t2.t3]
```

16

```
t4 = DiracSlash[a, b, c]
```

$\gamma\cdot a\gamma\cdot b\gamma\cdot c$

```
Chisholm[t4]
```

$-i\,(\gamma^{\$MU\$508}\gamma^5)\,\epsilon^{\$MU\$508abc}+\gamma\cdot c\,a\cdot b-\gamma\cdot b\,a\cdot c+\gamma\cdot a\,b\cdot c$

```
a1 = GA[μ, ν, ρ, σ, τ, κ]
```

$\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{\rho}\,\gamma^{\sigma}\,\gamma^{\tau}\,\gamma^{\kappa}$

```
a2 = Chisholm[a1]
```

$$\begin{aligned}&-i\,(\gamma^{\tau}\,\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\mu\nu\rho\sigma}+i\,(\gamma^{\sigma}\,\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\mu\nu\rho\tau}-i\,(\gamma^{\$MU\$511}\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\rho\sigma\tau\$MU\$511}\,g^{\mu\nu}+i\,(\gamma^{\$MU\$512}\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\nu\sigma\tau\$MU\$512}\,g^{\mu\rho}-\\&\quad i\,(\gamma^{\$MU\$513}\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\mu\sigma\tau\$MU\$513}\,g^{\nu\rho}+(\gamma^{\tau}\,\gamma^{\kappa})\,g^{\mu\sigma}\,g^{\nu\rho}-(\gamma^{\sigma}\,\gamma^{\kappa})\,g^{\mu\tau}\,g^{\nu\rho}-(\gamma^{\tau}\,\gamma^{\kappa})\,g^{\mu\rho}\,g^{\nu\sigma}+(\gamma^{\rho}\,\gamma^{\kappa})\,g^{\mu\tau}\,g^{\nu\sigma}+\\&\quad(\gamma^{\sigma}\,\gamma^{\kappa})\,g^{\mu\rho}\,g^{\nu\tau}-(\gamma^{\rho}\,\gamma^{\kappa})\,g^{\mu\sigma}\,g^{\nu\tau}+(\gamma^{\tau}\,\gamma^{\kappa})\,g^{\mu\nu}\,g^{\rho\sigma}-(\gamma^{\nu}\,\gamma^{\kappa})\,g^{\mu\tau}\,g^{\rho\sigma}+(\gamma^{\mu}\,\gamma^{\kappa})\,g^{\nu\tau}\,g^{\rho\sigma}-(\gamma^{\sigma}\,\gamma^{\kappa})\,g^{\mu\nu}\,g^{\rho\tau}+\\&\quad(\gamma^{\nu}\,\gamma^{\kappa})\,g^{\mu\sigma}\,g^{\rho\tau}-(\gamma^{\mu}\,\gamma^{\kappa})\,g^{\nu\sigma}\,g^{\rho\tau}-i\,(\gamma^{\$MU\$509}\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\mu\nu\rho\$MU\$509}\,g^{\sigma\tau}+(\gamma^{\rho}\,\gamma^{\kappa})\,g^{\mu\nu}\,g^{\sigma\tau}-(\gamma^{\nu}\,\gamma^{\kappa})\,g^{\mu\rho}\,g^{\sigma\tau}+(\gamma^{\mu}\,\gamma^{\kappa})\,g^{\nu\rho}\,g^{\sigma\tau}\end{aligned}$$

```
a3 = Chisholm[a1]
```

$$\begin{aligned}&-i\,(\gamma^{\tau}\,\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\mu\nu\rho\sigma}+i\,(\gamma^{\sigma}\,\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\mu\nu\rho\tau}-i\,(\gamma^{\$MU\$516}\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\rho\sigma\tau\$MU\$516}\,g^{\mu\nu}+i\,(\gamma^{\$MU\$517}\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\nu\sigma\tau\$MU\$517}\,g^{\mu\rho}-\\&\quad i\,(\gamma^{\$MU\$518}\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\mu\sigma\tau\$MU\$518}\,g^{\nu\rho}+(\gamma^{\tau}\,\gamma^{\kappa})\,g^{\mu\sigma}\,g^{\nu\rho}-(\gamma^{\sigma}\,\gamma^{\kappa})\,g^{\mu\tau}\,g^{\nu\rho}-(\gamma^{\tau}\,\gamma^{\kappa})\,g^{\mu\rho}\,g^{\nu\sigma}+(\gamma^{\rho}\,\gamma^{\kappa})\,g^{\mu\tau}\,g^{\nu\sigma}+\\&\quad(\gamma^{\sigma}\,\gamma^{\kappa})\,g^{\mu\rho}\,g^{\nu\tau}-(\gamma^{\rho}\,\gamma^{\kappa})\,g^{\mu\sigma}\,g^{\nu\tau}+(\gamma^{\tau}\,\gamma^{\kappa})\,g^{\mu\nu}\,g^{\rho\sigma}-(\gamma^{\nu}\,\gamma^{\kappa})\,g^{\mu\tau}\,g^{\rho\sigma}+(\gamma^{\mu}\,\gamma^{\kappa})\,g^{\nu\tau}\,g^{\rho\sigma}-(\gamma^{\sigma}\,\gamma^{\kappa})\,g^{\mu\nu}\,g^{\rho\tau}+\\&\quad(\gamma^{\nu}\,\gamma^{\kappa})\,g^{\mu\sigma}\,g^{\rho\tau}-(\gamma^{\mu}\,\gamma^{\kappa})\,g^{\nu\sigma}\,g^{\rho\tau}-i\,(\gamma^{\$MU\$514}\gamma^{\kappa}\,\gamma^5)\,\epsilon^{\mu\nu\rho\$MU\$514}\,g^{\sigma\tau}+(\gamma^{\rho}\,\gamma^{\kappa})\,g^{\mu\nu}\,g^{\sigma\tau}-(\gamma^{\nu}\,\gamma^{\kappa})\,g^{\mu\rho}\,g^{\sigma\tau}+(\gamma^{\mu}\,\gamma^{\kappa})\,g^{\nu\rho}\,g^{\sigma\tau}\end{aligned}$$

Check that both a1.a1 and a2.a3 give the same.

```
Calc[a1.a1]
```

−2048

```
Calc[a2.a3]
```

−2048

```
Clear[t1, t2, t3, t4, a1, a2, a3]
```

■ Collect2

■ Description

Collect2[expr,x] collects together terms which are not free of any occurrence of x.
Collect2[expr,{x1, x2, ...}] (or also Collect2[expr,x1, x2, ...]) collects together terms which are not free of any occurrence of x1, x2, ... The coefficients are put over a common denominator. If expr is expanded before collecting, depends on the option Factoring, which may be set to Factor, Factor2, or any other function, which is applied to the coefficients. If expr is already expanded with respect to x (x1,x2, ...), the option Expanding can be set to False.

```
Options[Collect2]
```

{Denominator→ False, Dot→ False, Expanding→ True, Factoring→ Factor2, IsolateNames→ False}

■ See also: Isolate.

■ Examples

```
Collect2[ t1 = a + r a + k^2 f[a] - k f[a] + x / 2 - y / w, a]
```

$$a(r+1)+\frac{wx-2y}{2w}-(1-k)\,k\,f(a)$$

```
Collect2[t1, a, Factoring -> False]
```

$$a(r+1)+\frac{x}{2}-\frac{y}{w}+(k^2-k)\,f(a)$$

```
Collect2[t1, a, Factoring -> Factor]
```

$$a(r+1)+\frac{wx-2y}{2w}+(k-1)\,k\,f(a)$$

```
Collect2[2 a (b - a) (h - 1) - b^2 (e a - c) + b^2, {a, b}]
```

$$2(1-h)\,a^2-b^2\,e\,a-2\,b(1-h)\,a+b^2\,(c+1)$$

```
Collect2[Expand[(a - b - c - d)^5], a, IsolateNames → L]
```

$$a^5-5\,L(1)\,a^4+10\,(L(1))^2\,a^3-10\,(L(1))^3\,a^2+5\,(L(1))^4\,a-(L(1))^5$$

```
ReleaseHold[%]
```

$$a^5 - 5\,(b+c+d)\,a^4 + 10\,(b+c+d)^2\,a^3 - 10\,(b+c+d)^3\,a^2 + 5\,(b+c+d)^4\,a - (b+c+d)^5$$

```
Clear[t1, L]
```

```
Collect2[Expand[(a - b - c)^3], a, Factoring -> fun]
```

$$\text{fun}(1)\,a^3 + \text{fun}(-3\,b - 3\,c)\,a^2 + \text{fun}(3\,b^2 + 6\,c\,b + 3\,c^2)\,a + \text{fun}(-b^3 - 3\,c\,b^2 - 3\,c^2\,b - c^3)$$

```
% /. fun -> FactorTerms
```

$$a^3 - 3\,(b+c)\,a^2 + 3\,(b^2 + 2\,c\,b + c^2)\,a - b^3 - c^3 - 3\,b\,c^2 - 3\,b^2\,c$$

■ Combine

■ Description

Combine[expr] puts terms in a sum over a common denominator, and cancels factors in the result. Combine is similar to Together, but accepts the option Expanding and works usually better than Together for polynomials involving rationals with sums in the denominator.

```
Options[Combine]
```

{Expanding → False}

■ See also: Factor2.

■ Examples

```
Combine[ (a - b) (c - d) / e + g ]
```

$$\frac{(a-b)\,(c-d) + e\,g}{e}$$

Here the result from Together, where the numerator is automatically expanded.

```
Together[ (a - b) (c - d) / e + g ]
```

$$\frac{a\,c - b\,c - a\,d + b\,d + e\,g}{e}$$

If the option Expanding is set to True, the result of Combine is the same as Together, but uses a slightly different algorithm.

```
Combine[((a - b) (c - d)) / e + g, Expanding → True]
```

$$\frac{ac - bc - ad + bd + eg}{e}$$

■ CommutatorExplicit

■ Description

CommutatorExplicit[exp] substitutes any Commutator and AntiCommutator in exp by their definitions.

■ See also: Calc, DotSimplify.

■ Examples

```
DeclareNonCommutative[a, b, c, d]
```

```
Commutator[a, b]
```

$[a, b]$

```
CommutatorExplicit[%]
```

$ab - ba$

```
AntiCommutator[a - c, b - d]
```

$\{a - c,\ b - d\}$

```
CommutatorExplicit[%]
```

$(a - c)(b - d) + (b - d)(a - c)$

```
CommutatorExplicit[%%] // DotSimplify
```

$ab - ad + ba - bc - cb + cd - da + dc$

```
UnDeclareNonCommutative[a, b, c, d]
```

■ Commutator

■ Description

Commutator[x, y] = c defines the commutator between the non−commuting objects x and y.

■ **See also: AntiCommutator, CommutatorExplicit, DeclareNonCommutative, DotSimplify.**

■ Examples

```
DeclareNonCommutative[a, b, c, d]
```

```
Commutator[a, b]
```

$[a, b]$

```
CommutatorExplicit[%]
```

$a b - b a$

```
DotSimplify[Commutator[a + b, c + d]]
```

$a c + a d + b c + b d - c a - c b - d a - d b$

```
UnDeclareNonCommutative[a, b, c, d]
```

Verify the Jacobi identity.

```
χ = Commutator;  DeclareNonCommutative[x, y, z];
```

```
χ[x, χ[y, z]] + χ[y, χ[z, x]] + χ[z, χ[x, y]]
```

$[x, [y, z]] + [y, [z, x]] + [z, [x, y]]$

```
DotSimplify[%]
```

0

```
Clear[χ]
```

```
UnDeclareNonCommutative[x, y, z]
```

■ ComplexConjugate

■ Description

ComplexConjugate[expr] complex conjugates expr.
It operates on fermion lines, i.e., products of Spinor[..].DiracMatrix[..].Spinor[..], and changes all
occuring LorentzIndex[mu] into LorentzIndex[ComplexIndex[mu]].
For taking the spin sum (i.e. constructing the traces) use
FermionSpinSum.
WARNING: In expr should be NO explicit I's in denominators!

■ **See also: ComplexIndex, FermionSpinSum, LorentzIndex.**

■ Examples

```
ComplexConjugate[MetricTensor[μ,ν]]
```

$g^{\mu^* \nu^*}$

```
InputForm[%]
```

```
Pair[LorentzIndex[ComplexIndex[μ]],
  LorentzIndex[ComplexIndex[ν]]]
```

```
GA[μ,ν,5]
```

$\gamma^\mu \gamma^\nu \gamma^5$

```
ComplexConjugate[%]
```

$-(\gamma^5 \gamma^{\nu^*} \gamma^{\mu^*})$

```
SUNTrace[SUNT[a,b,c]]
```

$\text{tr}(T_a T_b T_c)$

```
ComplexConjugate[%]
```

$\text{tr}(T_a T_b T_c)$

```
ComplexConjugate[SUNF[a,b,c]]
```

f_{abc}

```
ComplexConjugate[MetricTensor[μ,ν]]//FCE//InputForm
```

```
MT[ComplexIndex[μ], ComplexIndex[ν]]
```

```
SpinorUBar[k1, m].GA[λ].SpinorU[p1, m]
```

$\bar{u}(\text{k1}, m) \gamma^\lambda u(\text{p1}, m)$

```
ComplexConjugate[%]
```

$\varphi(\text{p1}, m) \gamma^{\lambda^*} \varphi(\text{k1}, m)$

Notice that SpinorUBar and SpinorU are only input functions Internally they are converted to Spinor objects (which display as φ).

■ ComplexIndex

■ Description

ComplexIndex is the head of a "complex conjugate index", which is understood as just another index.

■ See also: LorentzIndex.

■ Examples

```
LorentzIndex[μ]
```

μ

```
ComplexConjugate[%]
```

μ^*

```
% // InputForm
```

```
LorentzIndex[ComplexIndex[μ]]
```

```
FourVector[p, μ]
```

p^{μ}

```
ComplexConjugate[%]
```

p^{μ^*}

```
% // InputForm
```

```
Pair[LorentzIndex[ComplexIndex[μ]], Momentum[p]]
```

```
ComplexConjugate[%]
```

p^{μ}

■ Contract

■ Description

Contract[expr] contracts pairs of Lorentz indices of metric tensors, four–vectors and (depending on the option EpsContract) of Levi–Civita tensors in expr. For the contraction of Dirac matrices with each other use DiracSimplify.Contract[exp1,exp2] contracts (exp1*exp2), where exp1 and exp2 may be larger products of sums of metric tensors and 4–vectors.

```
Options[Contract]
```

```
{Collecting → True, EpsContract → True, Expanding → True, Factoring → False,
  FeynCalcInternal → False, MomentumCombine → False, Rename → False, Schouten → 0}
```

■ See also: Pair, DiracSimplify, MomentumCombine.

■ Examples

```
MetricTensor[μ, ν] FourVector[p, μ]
```

$g^{\mu\nu}\ p^{\mu}$

```
Contract[%]
```

p^{ν}

```
FourVector[p, μ] DiracMatrix[μ]
```

$\gamma^{\mu}\ p^{\mu}$

```
Contract[%]
```

$\gamma \cdot p$

```
MetricTensor[μ, μ]
```

$g^{\mu\mu}$

The default dimension for MetricTensor is 4.

```
Contract[%]
```

4

A shortway to enter D–dimensional metric tensors is given by MTD. The "." as multiplication operator is not necessary but just convenient for typesetting.

```
MTD[μ, ν] . MTD[μ, ν]
```

$g^{\mu\nu}\, g^{\mu\nu}$

```
Contract[%]
```

D

```
MTD[μ, ν] . MTD[μ, ν]
```

$g^{\mu\nu}\, g^{\mu\nu}$

```
FourVector[p, μ] FourVector[q, μ]
```

$p^{\mu}\, q^{\mu}$

```
Contract[% ]
```

$p\cdot q$

```
FourVector[p - q, μ] FourVector[a - b, μ]
```

$((a-b)^{\mu})\,((p-q)^{\mu})$

```
Contract[%]
```

$a\cdot p - a\cdot q - b\cdot p + b\cdot q$

```
LeviCivita[μ, ν, α, σ] FourVector[p, σ]
```

$\epsilon^{\mu\nu\alpha\sigma}\, p^{\sigma}$

```
Contract[%]
```

$\epsilon^{\alpha\mu\nu p}$

```
LeviCivita[μ, ν, α, β] LeviCivita[μ, ν, α, σ]
```

$\epsilon^{\mu\nu\alpha\beta}\, \epsilon^{\mu\nu\alpha\sigma}$

```
Contract[%]
```

$-6\, g^{\beta\sigma}$

```
SetOptions[Eps, Dimension → D]; LCD[μ, ν, α, β] LCD[μ, ν, α, σ]
```

$\epsilon^{\mu\nu\alpha\beta}\, \epsilon^{\mu\nu\alpha\sigma}$

```
Contract[%] // Factor2
```

$(1-D)\,(2-D)\,(3-D)\, g^{\beta\sigma}$

```
SetOptions[Eps, Dimension → 4];
```

■ Convolute

■ Description

Convolute[fg, x] convolutes f(x) and g(x), i.e.,
$\int_0^1 dx_1 \int_0^1 dx_2\, \delta(x - x_1 x_2)\, f(x_1)\, g(x_2)$.
Convolute[fg] is equivalent to Convolute[fg, x].
Convolute[exp, {x1, x2}] assumes that exp is polynomial in x1 and x2.
Convolute uses table–look–up and does not do any integral calculations, only linear algebra.

```
Options[Convolute]
```

{Bracket→ {ε}, FinalSubstitutions→ {PlusDistribution→ Identity}}

■ See also: PlusDistribution.

■ Examples

```
Convolute[1, 1]
```

$-\log(x)$

```
Convolute[x, x]
```

$-x \log(x)$

```
Convolute[1, x]
```

$1 - x$

```
Convolute[1, 1 / (1 - x)]
```

$\log(1 - x) - \log(x)$

```
Convolute[1, PlusDistribution[1 / (1 - x)]]
```

$\log(1 - x) - \log(x)$

```
Convolute[1 / (1 - x), x]
```

$\log(1 - x)\, x - \log(x)\, x - x + 1$

```
Convolute[1 / (1 - x), 1 / (1 - x)]
```

$$-\zeta(2)\, \delta(1 - x) + \frac{2 \log(1 - x)}{1 - x} - \frac{\log(x)}{1 - x}$$

```
Convolute[1, Log[1 - x]]
```

$-\log(1 - x) \log(x) - \mathrm{Li}_2(1 - x)$

```
Convolute[1, x Log[1 - x]]
```

$$x+(1-x)\log(1-x)-1$$

```
Convolute[1 / (1 - x), Log[1 - x]]
```

$$\log^2(1-x)-\log(x)\log(1-x)-\zeta(2)$$

```
Convolute[1 / (1 - x), x Log[1 - x]]
```

$$x\log^2(1-x)+(1-x)\log(1-x)-x\log(x)\log(1-x)+x-x\zeta(2)-1$$

```
Convolute[Log[1 - x] / (1 - x), x]
```

$$\frac{1}{2}x\log^2(1-x)+(1-x)\log(1-x)-x\log(x)\log(1-x)+x\log(x)-x\mathrm{Li}_2(1-x)$$

```
Convolute[1, x Log[x]]
```

$$-\log(x)\,x+x-1$$

```
Convolute[Log[1 - x], x]
```

$$(1-x)\log(1-x)+x\log(x)$$

```
Convolute[1 / (1 - x), Log[x] / (1 - x)]
```

$$\frac{\log(1-x)\log(x)}{1-x}-\frac{\log^2(x)}{2(1-x)}$$

```
Convolute[1, Log[x]]
```

$$-\frac{1}{2}\log^2(x)$$

```
Convolute[x, x Log[x]]
```

$$-\frac{1}{2}x\log^2(x)$$

```
Convolute[1 / (1 - x), Log[x]]
```

$$-\frac{1}{2}\log^2(x)+\log(1-x)\log(x)+\mathrm{Li}_2(1-x)$$

```
Convolute[1, Log[x] / (1 - x)]
```

$$-\frac{1}{2}\log^2(x)-\mathrm{Li}_2(1-x)$$

```
Convolute[1 / (1 - x), x Log[x]]
```

$$-\frac{1}{2}x\log^2(x)-x\log(x)+x\log(1-x)\log(x)+x+x\mathrm{Li}_2(1-x)-1$$

```
Convolute[Log[x] / (1 - x), x]
```

$$-\frac{1}{2} x \log^2(x) + \log(x) - x - x \text{Li}_2(1-x) + 1$$

```
Convolute[1, x Log[x]]
```

$$-\log(x)\, x + x - 1$$

```
Convolute[Log[x], x]
```

$$-x + \log(x) + 1$$

```
Convolute[1 / (1 - x), Log[1 - x] / (1 - x)]
```

$$\frac{3 \log^2(1-x)}{2(1-x)} - \frac{\log(x)\log(1-x)}{1-x} - \frac{\zeta(2)}{1-x} + \delta(1-x)\,\zeta(3)$$

■ CovariantD

■ Description

CovariantD[μ, a, b] is the covariant derivative for a bosonic field.
CovariantD[μ] is the covariant derivative for a fermionic field.
CovariantD[OPEDelta, a, b] is a short form for
CovariantD[mu,a,b]*FourVector[OPEDelta, mu].
CovariantD[{OPEDelta, a, b}, {n}] yields the product of n operators, where n is an integer.
CovariantD[OPEDelta, a, b, {m, n}] gives the expanded form of
CovariantD[OPEDelta, a, b]^m up to order
g^n for the gluon, where n is an integer and g the coupling constant indicated by the setting of the option CouplingConstant.
CovariantD[OPEDelta, {m, n}] gives the expanded form of
CovariantD[OPEDelta]^m up to order g^n of the fermionic field.

```
Options[CovariantD]
```

{CouplingConstant → g_s, DummyIndex → Automatic, Explicit → False, PartialD → RightPartialD, QuantumField → A}

Possible settings of PartialD are:
LeftPartialD, LeftRigthPartialD, RightPartialD.
The default setting of QuantumField is GaugeField.

■ See also: LeftPartialD, LeftRightPartialD, RightPartialD.

■ Examples

```
CovariantD[μ]
```

D_μ

```
CovariantD[μ, a, b]
```

$$D_{\mu}^{ab}$$

```
CovariantD[μ, Explicit -> True]
```

$$\vec{\partial}_{\mu} - i\, g_s\, (T_{c_1}\; A_{\mu}^{c_1})$$

The first argument of CovariantD is intepreted as type LorentzIndex, except for OPEDelta, which is type Momentum.

```
CovariantD[OPEDelta]
```

$$D_{\Delta}$$

```
CovariantD[OPEDelta, a, b]
```

$$D_{\Delta}^{ab}$$

```
CovariantD[OPEDelta, a, b, Explicit -> True]
```

$$\vec{\partial}_{\Delta}\; \delta_{ab} - g_s\, A_{\Delta}^{c_2}\; f_{abc_2}$$

```
CovariantD[OPEDelta, Explicit -> True]
```

$$\vec{\partial}_{\Delta} - i\, g_s\, (T_{c_3}\; A_{\Delta}^{c_3})$$

```
CovariantD[OPEDelta, a, b, {2}]
```

$$\left(\vec{\partial}_{\Delta}\; \delta_{ac8} - g_s\, A_{\Delta}^{e1}\; f_{ac8e1}\right)\left(\vec{\partial}_{\Delta}\; \delta_{bc8} - g_s\, A_{\Delta}^{e2}\; f_{c8be2}\right)$$

This gives m times $\vec{\partial}_{\Delta}$, the partial derivative $\vec{\partial}_{\mu}$ contracted with Δ^{μ}.

```
CovariantD[OPEDelta, a, b, {OPEm, 0}]
```

$$(\vec{\partial}_{\Delta})^{m}\, \delta_{ab}$$

The expansion up to first order in the coupling constant g_s (the sum is the FeynCalc OPESum).

```
CovariantD[OPEDelta, a, b, {OPEm, 1}]
```

$$(\vec{\partial}_{\Delta})^{m}\, \delta_{ab} - g_s \sum_{i=0}^{m-1} \left((\vec{\partial}_{\Delta})^{i}\, A_{\Delta}^{c_1}\, (\vec{\partial}_{\Delta})^{-i+m-1}\right) f_{abc_1}$$

The expansion up to second order in the g_s.

```
CovariantD[OPEDelta, a, b, {OPEm, 2}]
```

$$\delta_{ab}\, (\vec{\partial}_{\Delta})^{m} - g_s^2 \sum_{j=0}^{m-2} \sum_{i=0}^{j} \left((\vec{\partial}_{\Delta})^{i}\, A_{\Delta}^{c_1}\, (\vec{\partial}_{\Delta})^{j-i}\, A_{\Delta}^{c_2}\, (\vec{\partial}_{\Delta})^{-j+m-2}\right) f_{ac_1 e_1}\; f_{bc_2 e_1} - g_s \sum_{i=0}^{m-1} \left((\vec{\partial}_{\Delta})^{i}\, A_{\Delta}^{c_1}\, (\vec{\partial}_{\Delta})^{-i+m-1}\right) f_{abc_1}$$

```
CovariantD[OPEDelta, a, b]^OPEm
```

$$(D_{\Delta}^{ab})^{m}$$

```
CovariantD[OPEDelta, {OPEm, 2}]
```

$$(\vec{\partial}_\Delta)^m - i g_s \sum_{i=0}^{m-1} T_{c_1} (\vec{\partial}_\Delta)^i A_\Delta^{c_1} (\vec{\partial}_\Delta)^{-i+m-1} - g_s^2 \sum_{j=0}^{m-2} \sum_{i=0}^{j} T_{c_1} T_{c_2} (\vec{\partial}_\Delta)^i A_\Delta^{c_1} (\vec{\partial}_\Delta)^{j-i} A_\Delta^{c_2} (\vec{\partial}_\Delta)^{-j+m-2}$$

```
CovariantD[OPEDelta, Explicit -> True] // InputForm
```

```
-I*Gstrong*SUNT[SUNIndex[Subscripted[c[4]]]] .
    QuantumField[GaugeField, Momentum[OPEDelta],
     SUNIndex[Subscripted[c[4]]]] +
  RightPartialD[Momentum[OPEDelta]]
```

```
CovariantD[μ, a, b, Explicit -> True] // InputForm
```

```
RightPartialD[LorentzIndex[μ]]*SUNDelta[a, b] -
  Gstrong*QuantumField[GaugeField, LorentzIndex[μ],
    SUNIndex[Subscripted[c[5]]]]*
   SUNF[a, b, Subscripted[c[5]]]
```

■ D0

■ Description

D0[p10,p12,p23,p30,p20,p13, m1^2,m2^2,m3^2,m4^2] is the Passarino–Veltman D_0 function. The convention for the arguments is that if the denominator of the integrand has the form ([q^2–m1^2][(q+p1)^2–m2^2][(q+p2)^2–m3^2][(q+p3)^2–m4^2]), the first six arguments of D0 are the scalar products p10 = p1^2, p12 = (p1–p2)^2, p23 = (p2–p3)^2, p30 = p3^2, p20 = p2^2, p13 = (p1–p3)^2.

■ See also: B0, C0, PaVe, PaVeOrder.

■ Examples

```
D0[p10, p12, p23, p30, p20, p13, m1^2, m2^2, m3^2, m4^2]
```

$D_0(p10, p12, p23, p30, p20, p13, m1^2, m2^2, m3^2, m4^2)$

```
PaVeOrder[D0[p10, p12, p23, p30, p20, p13, m1^2, m2^2, m3^2, m4^2],
 PaVeOrderList -> {p13, p20}]
```

$D_0(p10, p30, p23, p12, p13, p20, m2^2, m1^2, m4^2, m3^2)$

```
PaVeOrder[%]
```

$D_0(p10, p12, p23, p30, p20, p13, m1^2, m2^2, m3^2, m4^2)$

■ DataType

■ Description

DataType[expr,type] = True defines the object expr to have data–type type. DataType[expr1,expr2,..., type] defines the objects expr1, expr2,... to have data–type type. The default setting is DataType[__,] := False. To assign a certain data–type do, e.g., DataType[x,PositiveInteger] = True.

■ See also: DeclareNonCommutative.

■ Examples

NonCommutative is just a data–type.

```
DataType[f, g, NonCommutative] = True;
```

```
t = f.g - g.(2 a).f
```

$$f\,g - g\,(2\,a)\,f$$

Since "f " and "g" have DataType NonCommutative the function DotSimplify extracts only "a" out of the noncommutative product.

```
DotSimplify[t]
```

$$f\,g - 2\,a\,(g\,f)$$

```
DataType[m, odd] = DataType[a, even] = True;
```

```
ptest1[x_] := x /. (-1)^n_ /; DataType[n, odd] :> -1;
```

```
ptest2[x_] := x /. (-1)^n_ /; DataType[n, even] :> 1;
```

```
t = (-1)^m + (-1)^a + (-1)^z
```

$$(-1)^a + (-1)^m + (-1)^z$$

```
ptest1[t]
```

$$-1 + (-1)^a + (-1)^z$$

```
ptest2[%]
```

$$(-1)^z$$

```
Clear[ptest1, ptest2, t, a, m];
```

```
DataType[m, ganzeZahl] = True;
```

```
f[x_] := x /. {(-1)^p_ /; DataType[p, ganzeZahl] :> 1};
```

```
test = (-1)^m + (-1)^n x
```

$(-1)^n x+(-1)^m$

```
f[test]
```

$(-1)^n x+1$

```
Clear[f, test];
```

```
DataType[f, g, NonCommutative] = False;
```

```
DataType[m, odd] = DataType[a, even] = False;
```

CertainFeynCalcobjectshaveDataTypePositiveIntegersettoTrue.

```
DataType[OPEm, PositiveInteger]
```

True

PowerSimplifyusestheDataTypeinformation.

```
PowerSimplify[ (-1)^(2 OPEm)]
```

1

```
PowerSimplify[ (- SO[q])^OPEm]
```

$(-1)^m (\Delta \cdot q)^m$

■ DeclareNonCommutative

■ Description

DeclareNonCommutative[a,b,...] declaresa,b,... to be non−commutative,
i.e.,DataType[a,b,.., NonCommutative]is set to True.

■ Seealso: DataType, UnDeclareNonCommutative.

■ Examples

```
DeclareNonCommutative[x]
```

As a sideeffectof DeclareNonCommutativexis declaredto beof datatypeNonCommutative.

```
DataType[x, NonCommutative]
```

True

```
DeclareNonCommutative[y, z]
```

```
DataType[a, x, y, z, NonCommutative]
```

{False, True, True, True}

```
UnDeclareNonCommutative[x, y, z]
```

```
DataType[a, x, y, z, NonCommutative]
```

{False, False, False, False}

■ DeltaFunction

■ Description

DeltaFunction[x] is the Dirac delta–function $\delta(x)$.

■ See also: Convolute, DeltaFunctionPrime, Integrate2, SimplifyDeltaFunction.

■ Examples

```
DeltaFunction[1 - x]
```

$\delta(1 - x)$

```
Integrate2[DeltaFunction[1 - x] f[x], {x, 0, 1}]
```

$f(1)$

```
Integrate2[DeltaFunction[x] f[x], {x, 0, 1}]
```

$f(0)$

```
Integrate2[DeltaFunction[1 - x] f[x], {x, 0, 1}]
```

$f(1)$

```
Convolute[DeltaFunction[1 - x], x]
```

x

■ DeltaFunctionPrime

■ Description

DeltaFunctionPrime[1 - x] is the derivative of the Dirac δ–function.

■ See also: Convolute, DeltaFunction, Integrate2.

■ Examples

```
DeltaFunctionPrime[1 - x]
```

$\delta'(1-x)$

```
Integrate2[DeltaFunctionPrime[1 - x] f[x], {x, 0, 1}]
```

$f'(1)$

```
Integrate2[DeltaFunctionPrime[1 - x] x^2, {x, 0, 1}]
```

2

■ Dimension

■ Description

Dimension is an option of several functions and denotes the space–time dimension. Possible settings are: 4, n, d, D, ... (the variable does not matter, but it should have head Symbol).

■ DiracGammaCombine

■ Description

DiracGammaCombine[exp] (nearly) the inverse operation to DiracGammaExpand.

■ See also: DiracGamma, DiracGammaExpand, DiracMatrix, DiracSimplify, DiracSlash, DiracTrick.

■ Examples

```
DiracGammaCombine[GS[p] + GS[q]]
```

$\gamma\cdot(p+q)$

```
InputForm[%]
```

```
DiracGamma[Momentum[p + q]]
```

```
DiracGammaCombine[2 GS[p] - 2 GS[q]]
```

$\gamma \cdot (2\,p - 2\,q)$

```
InputForm[%]
```

```
DiracGamma[Momentum[2*p - 2*q]]
```

```
DiracGammaExpand[%%]
```

$2\,\gamma \cdot p - 2\,\gamma \cdot q$

■ DiracGammaExpand

■ Description

DiracGammaExpand[exp] expands all DiracGamma[Momentum[a+b+...]] in exp into (DiracGamma[Momentum[a]] + DiracGamma[Momentum[b]] + ...).

■ See also: DiracGamma, DiracGammaCombine, DiracMatrix, DiracSimplify, DiracSlash, DiracTrick.

■ Examples

```
t = DiracGamma[Momentum[q]] . DiracGamma[Momentum[p - q]]
```

$\gamma \cdot q\,\gamma \cdot (p - q)$

Momentum is the head of p-q, i.e., it is treated as one four-momentum.

```
InputForm[t]
```

```
DiracGamma[Momentum[q]] . DiracGamma[Momentum[p - q]]
```

With DiracGammaExpand the Momentum[p-q] gets expanded.

```
DiracGammaExpand[t]
```

$\gamma \cdot q\,(\gamma \cdot p - \gamma \cdot q)$

The inverse operation is DiracGammaCombine.

```
InputForm[DiracGammaCombine[DiracGammaExpand[t]]]
```

```
DiracGamma[Momentum[q]] . DiracGamma[Momentum[p - q]]
```

```
InputForm[DiracGammaExpand[t]]
```

```
DiracGamma[Momentum[q]] .
  (DiracGamma[Momentum[p]] - DiracGamma[Momentum[q]])
```

In order to do non-commutative expansion use DiracSimplify.

```
DiracSimplify[t]
```

$\gamma \cdot q \gamma \cdot p - q^2$

```
Clear[t]
```

■ DiracGamma

■ Description

DiracGamma[x, dim] is the head of all Dirac matrices and slashes (in the internal representation). Use DiracMatrix (or GA, GAD) and DiracSlash (or GS, GSD) for manual (short) input.
DiracGamma[x^4] simplifies to DiracGamma[x].
DiracGamma[5] is γ^5.
DiracGamma[6] is (1 + γ^5) / 2.
DiracGamma[7] is (1 – γ^5) / 2.

■ See also: DiracGammaExpand, DiracMatrix, DiracSimplify, DiracSlash, DiracTrick.

■ Examples

```
DiracGamma[5]
```

γ^5

```
DiracGamma[LorentzIndex[α]]
```

γ^α

$\gamma \cdot q$ denotes q (= γ^μ q_μ).

```
DiracGamma[Momentum[q]]
```

$\gamma \cdot q$

```
DiracGamma[Momentum[q]] . DiracGamma[Momentum[p - q]]
```

$\gamma \cdot q \gamma \cdot (p - q)$

```
DiracGamma[Momentum[q, D], D]
```

$\gamma \cdot q$

```
a1 = GS[p - q].GS[p]
```

$\gamma \cdot (p - q) \gamma \cdot p$

```
a2 = DiracGammaExpand[a1]
```

$(\gamma \cdot p - \gamma \cdot q) \gamma \cdot p$

```
a3 = GAD[μ].GSD[p - q].GSD[q].GAD[μ]
```

$\gamma^\mu \gamma\cdot(p-q)\gamma\cdot q\gamma^\mu$

```
a4 = DiracTrick[a3]
```

$(D-4)(\gamma\cdot(p-q)\gamma\cdot q)+4(p-q)\cdot q$

```
a5 = DiracSimplify[a4]
```

$D(\gamma\cdot p\gamma\cdot q)-4(\gamma\cdot p\gamma\cdot q)+4\,p\cdot q-Dq^2$

```
Clear[a1, a2, a3, a4, a5]
```

■ DiracGammaT

■ Description

DiracGammaT[x] denotes the transpose of DiracGamma[x].
Transpose[DiracGammaT[x]] gives DiracGamma[x]. Note that x must have Head LorentzIndex or Momentum.

■ See also: DiracGamma.

■ Examples

```
DiracGammaT[LorentzIndex[μ]]
```

γ_μ^T

```
Transpose[%]
```

γ^μ

```
GS[p] // FCI // Transpose
```

$(\gamma\cdot p)^T$

■ DiracMatrix

■ Description

DiracMatrix[μ] denotes a Dirac γ matrix with Lorentz index μ.
DiracMatrix[μ,ν,...] is a product of γ matrices with Lorentz indices μ, ν, etc.
DiracMatrix[5] is γ^5.
DiracMatrix[6] is $1/2+\gamma^5/2$.
DiracMatrix[7] is $1/2-\gamma^5/2$.

■ **See also: DiracGammaExpand, DiracGamma, DiracSimplify, DiracSlash, DiracTrick, GA, GAD, GS, GSD.**

■ Examples

```
DiracMatrix[μ]
```

γ^{μ}

```
DiracMatrix[μ].DiracMatrix[ν]
```

$\gamma^{\mu}\ \gamma^{\nu}$

```
DiracMatrix[α] // InputForm
```

```
DiracGamma[LorentzIndex[α]]
```

```
DiracMatrix[μ] // FCE
```

γ^{μ}

```
% // InputForm
```

```
GA[μ]
```

```
GAD[μ]
```

γ^{μ}

```
% // FCI // InputForm
```

```
DiracGamma[LorentzIndex[μ, D], D]
```

```
GA[μ, ν, ρ]
```

$\gamma^{\mu}\ \gamma^{\nu}\ \gamma^{\rho}$

■ DiracOrder

■ Description

DiracOrder[expr] orders the Dirac matrices in expr alphabetically.
DiracOrder[expr, orderlist] orders the Dirac matrices in expr according to orderlist.

■ **See also: DiracSimplify, DiracTrick.**

■ Examples

```
t1 = GA[β, α]
```

$\gamma^{\beta}\,\gamma^{\alpha}$

```
DiracOrder[t1]
```

$2\,g^{\alpha\beta} - \gamma^{\alpha}\,\gamma^{\beta}$

This is a string of Dirac matrices in D dimensions.

```
t2 = GAD[μ, ν, μ]
```

$\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{\mu}$

```
DiracOrder[t1]
```

$2\,g^{\alpha\beta} - \gamma^{\alpha}\,\gamma^{\beta}$

```
t3 = GA[5, μ, ν]
```

$\gamma^{5}\,\gamma^{\mu}\,\gamma^{\nu}$

By default γ^5 is moved to the right

```
DiracOrder[t3]
```

$\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{5}$

```
t4 = GA[6, μ, 7]
```

$\gamma^{6}\,\gamma^{\mu}\,\gamma^{7}$

```
DiracOrder[t4]
```

$\gamma^{\mu}\,\gamma^{7}$

```
t5 = GA[α, β, δ]
```

$\gamma^{\alpha}\,\gamma^{\beta}\,\gamma^{\delta}$

This orders the γ^α γ^β γ^δ in reverse order.

```
DiracOrder[t5, {δ, β, α}]
```

$-(\gamma^{\delta}\,\gamma^{\beta}\,\gamma^{\alpha}) + 2\,\gamma^{\delta}\,g^{\alpha\beta} - 2\,\gamma^{\beta}\,g^{\alpha\delta} + 2\,\gamma^{\alpha}\,g^{\beta\delta}$

```
DiracOrder[%]
```

$\gamma^{\alpha}\,\gamma^{\beta}\,\gamma^{\delta}$

```
Clear[t1, t2, t3, t4, t5]
```

■ DiracReduce

■ Description

DiracReduce[expr] reduces all four–dimensional Dirac matrices in expr to the standard basis (S,P,V,A,T) using the Chisholm identity (see Chisholm). In the result the basic Dirac structures are wrapped with a head DiracBasis. I.e., S corresponds to DiracBasis[1], P : DiracBasis[DiracMatrix[5]], V: DiracBasis[DiracMatrix[mu]], A: DiracBasis[DiracMatrix[mu, 5]], T: DiracBasis[DiracSigma[DiracMatrix[mu]]]. By default DiracBasis is substituted to Identity. Notice that the result of DiracReduce is given in the FeynCalcExternal way, i.e., evtl. you may have to use FeynCalcInternal on the result.

```
Options[DiracReduce]
```

{Factoring → False, FinalSubstitutions → {DiracBasis → Identity}}

■ See also: DiracSimplify.

■ Examples

```
t1 = GA[μ, ν]
```

$\gamma^{\mu}\,\gamma^{\nu}$

```
DiracReduce[t1]
```

$g^{\mu\nu} - i\,\sigma^{\mu\nu}$

```
t2 = DiracMatrix[μ, ν, ρ]
```

$\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{\rho}$

```
DiracReduce[t2]
```

$i\,(\gamma^{\$MU(1)}\,\gamma^{5})\,\epsilon^{\mu\nu\rho\,\$MU(1)} + \gamma^{\rho}\,g^{\mu\nu} - \gamma^{\nu}\,g^{\mu\rho} + \gamma^{\mu}\,g^{\nu\rho}$

```
t3 = DiracMatrix[μ, ν, ρ, σ]
```

$\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{\rho}\,\gamma^{\sigma}$

```
t4 = DiracReduce[t3]
```

$-i\,\gamma^{5}\,\epsilon^{\mu\nu\rho\sigma} - i\,\sigma^{\rho\sigma}\,g^{\mu\nu} + i\,\sigma^{\nu\sigma}\,g^{\mu\rho} - i\,\sigma^{\nu\rho}\,g^{\mu\sigma} - i\,\sigma^{\mu\sigma}\,g^{\nu\rho} + g^{\mu\sigma}\,g^{\nu\rho} + i\,\sigma^{\mu\rho}\,g^{\nu\sigma} - g^{\mu\rho}\,g^{\nu\sigma} - i\,\sigma^{\mu\nu}\,g^{\rho\sigma} + g^{\mu\nu}\,g^{\rho\sigma}$

```
t5 = Calc[DiracSimplify[DiracSigmaExplicit[t4.t4]]]
```

-128

```
Calc[t4.t4]
```

-128

```
Clear[t1, t2, t3, t4]
```

■ DiracSigmaExplicit

■ Description

DiracSigmaExplicit[exp] inserts in exp for all DiracSigma its definition.
DiracSigmaExplicit is also an option of DiracSimplify.

■ See also: DiracSigma.

■ Examples

```
DiracSigma[GA[α], GA[β]]
```

$\sigma^{\alpha\beta}$

```
DiracSigmaExplicit[%]
```

$\frac{1}{2} i (\gamma^\alpha \gamma^\beta - \gamma^\beta \gamma^\alpha)$

■ DiracSigma

■ Description

DiracSigma[a,b] stands for i/2*(a . b – b . a) in 4 dimensions. a and b must have Head DiracGamma, DiracMatrix or DiracSlashOnly antisymmetry is implemented.

■ See also: DiracSigmaExplicit.

■ Examples

```
t1 = DiracSigma[GA[α],GA[β]]
```

$\sigma^{\alpha\beta}$

```
DiracSigmaExplicit[t1]
```

$$\frac{1}{2} i (\gamma^{\alpha} \gamma^{\beta} - \gamma^{\beta} \gamma^{\alpha})$$

```
t2 = DiracSigma[GA[β],GA[α]]
```

$$-\sigma^{\alpha \beta}$$

```
t3 = DiracSigma[GS[p],GS[q]]
```

$$\sigma^{p q}$$

```
DiracSigmaExplicit[t3]
```

$$\frac{1}{2} i (\gamma \cdot p \gamma \cdot q - \gamma \cdot q \gamma \cdot p)$$

```
Clear[t1, t2, t3]
```

■ DiracSimplify

■ Description

DiracSimplify[expr] simplifies products of Dirac matrices in expr and expands non-commutative products. Double Lorentz indices and four vectors are contracted. The Dirac equation is applied. All DiracMatrix[5], DiracMatrix[6] and DiracMatrix[7] are moved to the right. The order of the other Dirac matrices is not changed.

```
Options[DiracSimplify]
```

{DiracCanonical→ False, DiracSigmaExplicit→ True, DiracSimpCombine→ False, DiracSubstitute67→ False, Expanding→ True, Factoring→ False, FeynCalcInternal→ False, InsideDiracTrace→ False}

■ See also: Calc, DiracGammaExpand, DiracSimplify.

■ Examples

This is a string of Dirac matrices in four dimensions.

```
t1 = GA[μ, ν, μ]
```

$$\gamma^{\mu} \gamma^{\nu} \gamma^{\mu}$$

```
DiracSimplify[t1]
```

$$-2 \gamma^{\nu}$$

This is a string of Dirac matrices in D dimensions.

```
t2 = GAD[μ, ν, μ]
```

$\gamma^{\mu}\ \gamma^{\nu}\ \gamma^{\mu}$

```
DiracSimplify[t1]
```

$-2\,\gamma^{\nu}$

```
t3 = GA[5, μ, ν]
```

$\gamma^{5}\ \gamma^{\mu}\ \gamma^{\nu}$

By default γ^5 is moved to the right

```
DiracSimplify[t3]
```

$\gamma^{\mu}\ \gamma^{\nu}\ \gamma^{5}$

```
t4 = GA[6, μ, 7]
```

$\gamma^{6}\ \gamma^{\mu}\ \gamma^{7}$

```
DiracSimplify[t4]
```

$\gamma^{\mu}\ \gamma^{7}$

```
t5 = GS[a + b] . GS[p].GS[p].GS[c + d]
```

$\gamma\cdot(a+b)\,\gamma\cdot p\gamma\cdot p\gamma\cdot(c+d)$

Contrary to DiracTrick, DiracSimplify does non–commutative expansion.

```
DiracSimplify[t5]
```

$(\gamma\cdot a\gamma\cdot c)\ p^2 + (\gamma\cdot a\gamma\cdot d)\ p^2 + (\gamma\cdot b\gamma\cdot c)\ p^2 + (\gamma\cdot b\gamma\cdot d)\ p^2$

```
DiracTrick[t5]
```

$(\gamma\cdot(a+b)\,\gamma\cdot(c+d))\ p^2$

```
t6 = SpinorVBar[p] . GS[p] . SpinorUBar[q]
```

$\bar{v}(p)\,\gamma\cdot p\,\bar{u}(q)$

```
DiracSimplify[t6]
```

0

```
GAD@@Join[{μ}, Table[ν_i, {i, 6}], {μ}]
```

$\gamma^{\mu}\ \gamma^{\nu_1}\ \gamma^{\nu_2}\ \gamma^{\nu_3}\ \gamma^{\nu_4}\ \gamma^{\nu_5}\ \gamma^{\nu_6}\ \gamma^{\mu}$

```
DiracSimplify[%]
```

$$D(\gamma^{\nu_1}\gamma^{\nu_2}\gamma^{\nu_3}\gamma^{\nu_4}\gamma^{\nu_5}\gamma^{\nu_6}) - 12(\gamma^{\nu_1}\gamma^{\nu_2}\gamma^{\nu_3}\gamma^{\nu_4}\gamma^{\nu_5}\gamma^{\nu_6}) + 4(\gamma^{\nu_3}\gamma^{\nu_4}\gamma^{\nu_5}\gamma^{\nu_6})g^{\nu_1\nu_2} -$$
$$4(\gamma^{\nu_2}\gamma^{\nu_4}\gamma^{\nu_5}\gamma^{\nu_6})g^{\nu_1\nu_3} + 4(\gamma^{\nu_2}\gamma^{\nu_3}\gamma^{\nu_5}\gamma^{\nu_6})g^{\nu_1\nu_4} - 4(\gamma^{\nu_2}\gamma^{\nu_3}\gamma^{\nu_4}\gamma^{\nu_6})g^{\nu_1\nu_5} +$$
$$4(\gamma^{\nu_2}\gamma^{\nu_3}\gamma^{\nu_4}\gamma^{\nu_5})g^{\nu_1\nu_6} + 4(\gamma^{\nu_1}\gamma^{\nu_4}\gamma^{\nu_5}\gamma^{\nu_6})g^{\nu_2\nu_3} - 4(\gamma^{\nu_1}\gamma^{\nu_3}\gamma^{\nu_5}\gamma^{\nu_6})g^{\nu_2\nu_4} +$$
$$4(\gamma^{\nu_1}\gamma^{\nu_3}\gamma^{\nu_4}\gamma^{\nu_6})g^{\nu_2\nu_5} - 4(\gamma^{\nu_1}\gamma^{\nu_3}\gamma^{\nu_4}\gamma^{\nu_5})g^{\nu_2\nu_6} + 4(\gamma^{\nu_1}\gamma^{\nu_2}\gamma^{\nu_5}\gamma^{\nu_6})g^{\nu_3\nu_4} - 4(\gamma^{\nu_1}\gamma^{\nu_2}\gamma^{\nu_4}\gamma^{\nu_6})g^{\nu_3\nu_5} +$$
$$4(\gamma^{\nu_1}\gamma^{\nu_2}\gamma^{\nu_4}\gamma^{\nu_5})g^{\nu_3\nu_6} + 4(\gamma^{\nu_1}\gamma^{\nu_2}\gamma^{\nu_3}\gamma^{\nu_6})g^{\nu_4\nu_5} - 4(\gamma^{\nu_1}\gamma^{\nu_2}\gamma^{\nu_3}\gamma^{\nu_5})g^{\nu_4\nu_6} + 4(\gamma^{\nu_1}\gamma^{\nu_2}\gamma^{\nu_3}\gamma^{\nu_4})g^{\nu_5\nu_6}$$

With the option DiracCanonical an alphabetic ordering is done.

```
DiracSimplify[GA[ν, μ], DiracCanonical → True]
```

$$2\,g^{\mu\nu} - \gamma^{\mu}\gamma^{\nu}$$

Setting InsideDiracTrace→True assumes that a trace is still to be taken (later).

```
DiracSimplify[GA[μ, ν, ρ, σ], InsideDiracTrace → True]
```

$$g^{\mu\sigma}g^{\nu\rho} - g^{\mu\rho}g^{\nu\sigma} + g^{\mu\nu}g^{\rho\sigma}$$

```
DiracSimplify[GA[μ, ν, ρ], InsideDiracTrace → True]
```

$$0$$

```
Clear[t1, t2, t3, t4, t5, t6]
```

■ DiracSimplify2

■ Description

DiracSimplify2[exp] simplifies the Dirac structure but leaves any γ^5 untouched.

■ See also: DiracSimplify.

■ Examples

```
GAD[μ, ν, μ, 5, α, β, α]
```

$$\gamma^{\mu}\gamma^{\nu}\gamma^{\mu}\gamma^{5}\gamma^{\alpha}\gamma^{\beta}\gamma^{\alpha}$$

```
DiracSimplify2[%]
```

$$\gamma^{\mu}\gamma^{\nu}\gamma^{\mu}\gamma^{5}\gamma^{\alpha}\gamma^{\beta}\gamma^{\alpha}$$

■ DiracSlash

■ Description

DiracSlash[p] is the contraction $p^\mu \gamma_\mu$ (FourVector[p, μ] DiracMatrix[μ]). Products of those can be entered in the form DiracSlash[p1, p2, ...].

```
Options[DiracSlash]
```

{Dimension → 4, FeynCalcInternal → True}

■ See also: DiracGammaExpand, DiracGamma, DiracMatrix, DiracSimplify, DiracTrick, GS, GSD.

■ Examples

This is q' ($= \gamma^\mu q_\mu$).

```
DiracSlash[q]
```

$\gamma \cdot q$

```
DiracSlash[p].DiracSlash[q]
```

$\gamma \cdot p \gamma \cdot q$

```
DiracSlash[p, q]
```

$\gamma \cdot p \gamma \cdot q$

```
GS[p]
```

$\gamma \cdot p$

```
DiracSlash[q] // InputForm
```

```
DiracGamma[Momentum[q]]
```

```
DiracSlash[q, Dimension -> n] // InputForm
```

```
DiracGamma[Momentum[q, n], n]
```

■ DiracTrace

■ Description

DiracTrace[expr] is the head of DiracTraces. Whether the trace is evaluated depends on the option DiracTraceEvaluate. Direct trace evaluation is performed by Tr. The argument expr may be a product of Dirac matrices or slashes separated by the Mathematica Dot ".".

```
Options[DiracTrace]
```

```
{EpsContract→ False, Factoring→ False, FeynCalcExternal→ False, Mandelstam→ {},
  PairCollect→ True, DiracTraceEvaluate→ False, Schouten→ 0, LeviCivitaSign→ -1}
```

For comments regarding gamma^5 schemes see the notes for Tr.

■ See also: Tr.

■ Examples

```
DiracTrace[GA[μ, ν]]
```

$$\text{tr}(\gamma^{\mu}\,\gamma^{\nu})$$

```
DiracTrace[GA[μ, ν, ρ, σ]]
```

$$\text{tr}(\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{\rho}\,\gamma^{\sigma})$$

```
% /. DiracTrace → Tr
```

$$4\,(g^{\mu\sigma}\,g^{\nu\rho} - g^{\mu\rho}\,g^{\nu\sigma} + g^{\mu\nu}\,g^{\rho\sigma})$$

```
DiracTrace[GA[μ, ν, ρ, σ, 5], DiracTraceEvaluate → True]
```

$$-4\,i\,\epsilon^{\mu\nu\rho\sigma}$$

```
DiracTrace[GA[μ, ν, ρ, σ, δ, τ, 5], DiracTraceEvaluate → True]
```

$$4\,(i\,\epsilon^{\nu\rho\sigma\tau}\,g^{\delta\mu} - i\,\epsilon^{\mu\rho\sigma\tau}\,g^{\delta\nu} + i\,\epsilon^{\mu\nu\sigma\tau}\,g^{\delta\rho} - i\,\epsilon^{\mu\nu\rho\tau}\,g^{\delta\sigma} - i\,\epsilon^{\mu\nu\rho\sigma}\,g^{\delta\tau} - i\,\epsilon^{\delta\rho\sigma\tau}\,g^{\mu\nu} + i\,\epsilon^{\delta\nu\sigma\tau}\,g^{\mu\rho} - i\,\epsilon^{\delta\nu\rho\tau}\,g^{\mu\sigma} + i\,\epsilon^{\delta\nu\rho\sigma}\,g^{\mu\tau} - i\,\epsilon^{\delta\mu\sigma\tau}\,g^{\nu\rho} + i\,\epsilon^{\delta\mu\rho\tau}\,g^{\nu\sigma} - i\,\epsilon^{\delta\mu\rho\sigma}\,g^{\nu\tau} - i\,\epsilon^{\delta\mu\nu\tau}\,g^{\rho\sigma} + i\,\epsilon^{\delta\mu\nu\sigma}\,g^{\rho\tau} - i\,\epsilon^{\delta\mu\nu\rho}\,g^{\sigma\tau})$$

```
DiracTrace[GS[p, q, r, s]]
```

$$\text{tr}(\gamma\cdot p\gamma\cdot q\gamma\cdot r\gamma\cdot s)$$

```
DiracTrace[GA[μ, ν], DiracTraceEvaluate → True, FCE → True]
```

$$4\,g^{\mu\nu}$$

```
% // InputForm
```

```
4*MT[μ, ν]
```

■ DiracTrick

■ Description

DiracTrick[exp] contracts γ–matrices with each other and performs several simplifications (no expansion, use Calc or DiracSimplify for this).

```
Options[DiracTrick]
```

{Expanding → False}

■ See also: Calc, DiracGammaExpand, DiracSimplify.

■ Examples

This is a string of Dirac matrices in four dimensions.

```
t1 = GA[μ, ν, μ]
```

$\gamma^\mu \gamma^\nu \gamma^\mu$

```
DiracTrick[t1]
```

$-2 \gamma^\nu$

This is a string of Dirac matrices in D dimensions.

```
t2 = GAD[μ, ν, μ]
```

$\gamma^\mu \gamma^\nu \gamma^\mu$

```
DiracTrick[t2]
```

$(2 - D) \gamma^\nu$

```
t3 = GA[5, μ, ν]
```

$\gamma^5 \gamma^\mu \gamma^\nu$

By default γ^5 is moved to the right

```
DiracTrick[t3]
```

$\gamma^\mu \gamma^\nu \gamma^5$

```
t4 = GA[6, μ, 7]
```

$$\gamma^6\ \gamma^\mu\ \gamma^7$$

```
DiracTrick[t4]
```

$$\gamma^\mu\ \gamma^7$$

```
t5 = GS[a + b] . GS[p].GS[p].GS[c + d]
```

$$\gamma\cdot(a+b)\,\gamma\cdot p\gamma\cdot p\gamma\cdot(c+d)$$

```
DiracTrick[t5]
```

$$(\gamma\cdot(a+b)\,\gamma\cdot(c+d))\ p^2$$

```
Calc[t5]
```

$$(\gamma\cdot a\gamma\cdot c)\ p^2+(\gamma\cdot a\gamma\cdot d)\ p^2+(\gamma\cdot b\gamma\cdot c)\ p^2+(\gamma\cdot b\gamma\cdot d)\ p^2$$

```
GAD@@Join[{μ}, Table[ν_i, {i, 6}], {μ}]
```

$$\gamma^\mu\ \gamma^{\nu_1}\ \gamma^{\nu_2}\ \gamma^{\nu_3}\ \gamma^{\nu_4}\ \gamma^{\nu_5}\ \gamma^{\nu_6}\ \gamma^\mu$$

```
DiracTrick[%]
```

$$\begin{aligned}&(D-12)\,(\gamma^{\nu_1}\ \gamma^{\nu_2}\ \gamma^{\nu_3}\ \gamma^{\nu_4}\ \gamma^{\nu_5}\ \gamma^{\nu_6})-\\&\quad 4\,(-(\gamma^{\nu_3}\ \gamma^{\nu_4}\ \gamma^{\nu_5}\ \gamma^{\nu_6})\,g^{\nu_1\,\nu_2}+(\gamma^{\nu_2}\ \gamma^{\nu_4}\ \gamma^{\nu_5}\ \gamma^{\nu_6})\,g^{\nu_1\,\nu_3}-(\gamma^{\nu_2}\ \gamma^{\nu_3}\ \gamma^{\nu_5}\ \gamma^{\nu_6})\,g^{\nu_1\,\nu_4}+(\gamma^{\nu_2}\ \gamma^{\nu_3}\ \gamma^{\nu_4}\ \gamma^{\nu_6})\,g^{\nu_1\,\nu_5}-\\&\qquad(\gamma^{\nu_2}\ \gamma^{\nu_3}\ \gamma^{\nu_4}\ \gamma^{\nu_5})\,g^{\nu_1\,\nu_6}-(\gamma^{\nu_1}\ \gamma^{\nu_4}\ \gamma^{\nu_5}\ \gamma^{\nu_6})\,g^{\nu_2\,\nu_3}+(\gamma^{\nu_1}\ \gamma^{\nu_3}\ \gamma^{\nu_5}\ \gamma^{\nu_6})\,g^{\nu_2\,\nu_4}-\\&\qquad(\gamma^{\nu_1}\ \gamma^{\nu_3}\ \gamma^{\nu_4}\ \gamma^{\nu_6})\,g^{\nu_2\,\nu_5}+(\gamma^{\nu_1}\ \gamma^{\nu_3}\ \gamma^{\nu_4}\ \gamma^{\nu_5})\,g^{\nu_2\,\nu_6}-(\gamma^{\nu_1}\ \gamma^{\nu_2}\ \gamma^{\nu_5}\ \gamma^{\nu_6})\,g^{\nu_3\,\nu_4}+(\gamma^{\nu_1}\ \gamma^{\nu_2}\ \gamma^{\nu_4}\ \gamma^{\nu_6})\,g^{\nu_3\,\nu_5}-\\&\qquad(\gamma^{\nu_1}\ \gamma^{\nu_2}\ \gamma^{\nu_4}\ \gamma^{\nu_5})\,g^{\nu_3\,\nu_6}-(\gamma^{\nu_1}\ \gamma^{\nu_2}\ \gamma^{\nu_3}\ \gamma^{\nu_6})\,g^{\nu_4\,\nu_5}+(\gamma^{\nu_1}\ \gamma^{\nu_2}\ \gamma^{\nu_3}\ \gamma^{\nu_5})\,g^{\nu_4\,\nu_6}-(\gamma^{\nu_1}\ \gamma^{\nu_2}\ \gamma^{\nu_3}\ \gamma^{\nu_4})\,g^{\nu_5\,\nu_6})\end{aligned}$$

■ DotSimplify

■ Description

DotSimplify[expr] expands and reorders noncommutative terms in expr.
Simplifying relations may be specified by the option DotSimplifyRelations or by Commutator and AntiCommutator definitions.
Whether expr is expanded noncommutatively depends on the option Expanding.

```
Options[DotSimplify]
```

{Expanding → True, DotSimplifyRelations → {}, DotPower → True}

■ See also: AntiCommutator, Commutator, Calc.

■ Examples

```
t1 = GA[μ].(2 GS[p] - GS[q]).GA[ν]
```

$\gamma^{\mu} (2 \gamma \cdot p - \gamma \cdot q) \gamma^{\nu}$

```
DotSimplify[t1]
```

$2 (\gamma^{\mu} \gamma \cdot p \gamma^{\nu}) - \gamma^{\mu} \gamma \cdot q \gamma^{\nu}$

```
DeclareNonCommutative[a, b, c]
```

```
t2 = a.(b - z c).a
```

$a (b - c z) a$

```
DotSimplify[t2]
```

$a b a - z (a c a)$

```
Commutator[a, c] = 1
```

1

```
DotSimplify[t2]
```

$a b a - z (a + c a^2)$

```
Commutator[a, c] = .
```

```
DotSimplify[t2]
```

$a b a - z (a c a)$

```
AntiCommutator[b, a] = c
```

c

```
DotSimplify[t2]
```

$a c - a^2 b - z (a c a)$

```
AntiCommutator[b, a] = .
```

```
DotSimplify[t2, DotSimplifyRelations → {a.c → 1 / z}]
```

$a b a - a$

```
DeclareNonCommutative[x]
```

```
DotSimplify[x.x.x]
```

x^3

```
DotSimplify[x.x.x, DotPower → False]
```

$x\,x\,x$

```
UnDeclareNonCommutative[a, b, c, x]
```

■ EpsChisholm

■ Description

EpsChisholm[expr] substitutes for a gamma matrix contracted with a Levi Civita tensor (Eps) the Chisholm identity.

■ See also: Chisholm.

■ Examples

```
Chisholm[GA[μ, ν, ρ, σ]]
```

$$-i\,(\gamma^{\$MU\$667}\,\gamma^{\sigma}\,\gamma^{5})\,\epsilon^{\mu\,\nu\,\rho\,\$MU\$667}+(\gamma^{\rho}\,\gamma^{\sigma})\,g^{\mu\,\nu}-(\gamma^{\nu}\,\gamma^{\sigma})\,g^{\mu\,\rho}+(\gamma^{\mu}\,\gamma^{\sigma})\,g^{\nu\,\rho}$$

```
EpsChisholm[%]
```

$\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{\rho}\,\gamma^{\sigma}$

■ EpsEvaluate

■ Description

EpsEvaluate[expr] applies total antisymmetry and linearity (w.r.t. Momentum's) to all Levi–Civita tensors (Eps') in expr.

■ See also: Contract, Eps, LeviCivita, Trick.

■ Examples

```
Trick[LeviCivita[μ, ν, ρ, σ] FourVector[p + q, σ]]
```

$\epsilon^{\mu\,\nu\,\rho\,p+q}$

```
EpsEvaluate[%]
```

$\epsilon^{\mu\,\nu\,\rho\,p}+\epsilon^{\mu\,\nu\,\rho\,q}$

```
InputForm[%]
```

```
Eps[LorentzIndex[μ], LorentzIndex[ν],
   LorentzIndex[ρ], Momentum[p]] +
  Eps[LorentzIndex[μ], LorentzIndex[ν],
   LorentzIndex[ρ], Momentum[q]]
```

■ Epsilon

■ Description

Epsilon is $(n-4)$, where n is the space–time dimension. Epsilon stands for a small positive number.

■ See also: Series2.

■ Examples

```
Epsilon
```

ε

Epsilon has no functional properties, but some upvalues are changed:

```
{Re[Epsilon] > -4, Re[Epsilon] > -3,
 Re[Epsilon] > -2, Re[Epsilon] > -1, Re[Epsilon] > 0}
```

{True, True, True, True, True}

■ Eps

■ Description

Eps[a,b, c, d] is the head of the totally antisymmetric (– (Levi–Civita) tensor. The a,b,... may have head LorentzIndex, Momentum or Integer. In case of integers the Levi–Civita tensor is evaluated immediately.

```
Options[Eps]
```

{Dimension→ 4}

■ See also: EpsEvaluate, LC, LCD, LeviCivita.

■ Examples

```
Eps[LorentzIndex[μ], LorentzIndex[ν], LorentzIndex[ρ], LorentzIndex[σ]]
```

$\epsilon^{\mu\nu\rho\sigma}$

```
Eps[Momentum[p], LorentzIndex[ν], LorentzIndex[ρ], LorentzIndex[σ]]
```

$$\epsilon^{p\nu\rho\sigma}$$

```
Eps[b, a, c, d] // InputForm
```

```
-Eps[a, b, c, d]
```

```
Eps[0, 1, 2, 3]
```

$$1$$

```
Eps[1, 0, 2, 3]
```

$$-1$$

```
SetOptions[Eps, Dimension → 4];
```

```
a1 = Eps[LorentzIndex[μ, D], LorentzIndex[ν, D], LorentzIndex[ρ, D], LorentzIndex[σ, D]]
```

$$\epsilon^{\mu\nu\rho\sigma}$$

```
Contract[a1 a1]
```

$$-24$$

```
SetOptions[Eps, Dimension → D];
```

```
a2 = Eps[LorentzIndex[μ, D], LorentzIndex[ν, D], LorentzIndex[ρ, D], LorentzIndex[σ, D]]
```

$$\epsilon^{\mu\nu\rho\sigma}$$

```
Contract[a2 a2] // Factor2
```

$$(1-D)\,(2-D)\,(3-D)\,D$$

```
g5 = -I/24 LCD[μ,ν,ρ,α] .  GAD[μ,ν,ρ, α]//FCI
```

$$-\frac{1}{24}\,i\,(\epsilon^{\mu\nu\rho\alpha}\,\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{\rho}\,\gamma^{\alpha})$$

```
g5p = -I/24 LCD[μ',ν',ρ',α'] .
 GAD[μ',ν', ρ',α']//FCI
```

$$-\frac{1}{24}\,i\,(\epsilon^{\mu'\nu'\rho'\alpha'}\,\gamma^{\mu'}\,\gamma^{\nu'}\,\gamma^{\rho'}\,\gamma^{\alpha'})$$

```
g52 = Calc[g5 . g5p]//Factor2
```

$$-\frac{1}{24}\,(1-D)\,(2-D)\,(3-D)\,D$$

```
g52 /. D → 4
```

$$1$$

```
Clear[a1, a2, g5, g5p]
```

■ ExpandPartialD

■ Description

ExpandPartialD[exp] expands all products of QuantumField's and partial differentiation operators in exp and applies Leibniz rule.

■ See also: ExplicitPartialD, LeftPartialD, LeftRightPartialD, RightPartialD.

■ Examples

```
RightPartialD[μ].QuantumField[A, LorentzIndex[μ]].QuantumField[A, LorentzIndex[ν]]
```

$\vec{\partial}_\mu \; A_\mu \; A_\nu$

```
ExpandPartialD[%]
```

$A_\mu \; \partial_\mu \; A_\nu + \partial_\mu \; A_\mu \; A_\nu$

```
% // InputForm
```

```
QuantumField[A, LorentzIndex[μ]] .
   QuantumField[PartialD[LorentzIndex[μ]], A,
    LorentzIndex[ν]] +
  QuantumField[PartialD[LorentzIndex[μ]], A,
    LorentzIndex[μ]] .
   QuantumField[A, LorentzIndex[ν]]
```

```
LeftRightPartialD[μ].QuantumField[A, LorentzIndex[ν]]
```

$\overleftrightarrow{\partial}_\mu \; A_\nu$

```
ExpandPartialD[%]
```

$\frac{\partial_\mu \; A_\nu}{2} - \frac{1}{2} \overleftarrow{\partial}_\mu \; A_\nu$

```
QuantumField[A, LorentzIndex[μ]].
 (LeftRightPartialD[OPEDelta]^2).QuantumField[A, LorentzIndex[ρ]]
```

$A_\mu \; \overleftrightarrow{\partial}_\Delta^2 \; A_\rho$

```
ExpandPartialD[%]
```

$\frac{1}{4} \left(A_\mu \; \partial_{\Delta^2} \; A_\rho\right) - \frac{1}{2} \left(\partial_\Delta \; A_\mu \; \partial_\Delta \; A_\rho\right) + \frac{1}{4} \left(\partial_{\Delta^2} \; A_\mu \; A_\rho\right)$

```
8 LeftRightPartialD[OPEDelta]^3
```

$8 \overleftrightarrow{\partial}_\Delta^3$

```
ExplicitPartialD[%]
```

$$(\vec{\partial}_\Delta - \overleftarrow{\partial}_\Delta)(\vec{\partial}_\Delta - \overleftarrow{\partial}_\Delta)(\vec{\partial}_\Delta - \overleftarrow{\partial}_\Delta)$$

```
ExpandPartialD[%]
```

$$-3\,\overleftarrow{\partial}_\Delta\,\vec{\partial}_\Delta^{\,2} + 3\left(\overleftarrow{\partial}_\Delta^{\,2}\,\vec{\partial}_\Delta\right) - \overleftarrow{\partial}_\Delta^{\,3} + \vec{\partial}_\Delta^{\,3}$$

```
LeviCivita[μ, ν, ρ, τ] RightPartialD[α, μ, β, ν]
```

$$\left(\vec{\partial}_\alpha\,\vec{\partial}_\mu\,\vec{\partial}_\beta\,\vec{\partial}_\nu\right)\epsilon^{\mu\nu\rho\tau}$$

```
ExpandPartialD[%]
```

0

■ ExpandScalarProduct

■ Description

ExpandScalarProduct[expr] expands scalar products of sums of momenta in expr. ExpandScalarProduct does not use Expand on expr.

■ See also: Calc.

■ Examples

```
SP[p, p - q]
```

$p\cdot(p-q)$

```
ExpandScalarProduct[%]
```

$p^2 - p\cdot q$

```
FV[p - q, μ]
```

$(p-q)^\mu$

```
ExpandScalarProduct[%]
```

$p^\mu - q^\mu$

```
SP[p - q, q - r] // FCI
```

$(p-q)\cdot(q-r)$

```
% /. Pair → ExpandScalarProduct
```

$p\cdot q - p\cdot r - q^2 + q\cdot r$

■ Explicit

■ Description

Explicit is an option for FieldStrength, GluonVertex, SUNF, and
Twist2GluonOperator.
If set to True the full form of the operator is inserted. Explicit[exp]
inserts explicit expressions of GluonVertex and Twist2GluonOperator in exp. SUNF's are replaced by SUNTrace objects.

■ See also: GluonVertex, Twist2GluonOperator.

■ Examples

```
GluonVertex[p, μ, a, q, ν, b, r, ρ, c]
```

$g_s\, V^{\mu\nu\rho}(p,\ q,\ r)\, f_{abc}$

```
Explicit[%]
```

$g_s\,(q^\mu\, g^{\nu\rho} - r^\mu\, g^{\nu\rho} - g^{\mu\rho}\, p^\nu + g^{\mu\rho}\, r^\nu + g^{\mu\nu}\, p^\rho - g^{\mu\nu}\, q^\rho)\, f_{abc}$

```
Twist2GluonOperator[p, μ, a, ν, b]
```

$\frac{1}{2}\,((-1)^m + 1)\,\delta_{ab}\,(O^{\mathrm{G2}}_{\mu\nu}(p))$

```
Explicit[%]
```

$\frac{1}{2}\,(g^{\mu\nu}\,(\Delta\cdot p)^2 - (p^\mu\,\Delta^\nu + \Delta^\mu\, p^\nu)\,\Delta\cdot p + \Delta^\mu\,\Delta^\nu\, p^2)\,((-1)^m + 1)\,(\Delta\cdot p)^{m-2}\,\delta_{ab}$

```
FieldStrength[μ, ν, a]
```

$F^a_{\mu\nu}$

```
Explicit[%]
```

$\partial_\mu\, A^a_\nu - \partial_\nu\, A^a_\mu + g_s\,(A^{\mathrm{b1}}_\mu\, A^{\mathrm{c10}}_\nu)\, f_{\mathrm{ab1c10}}$

■ ExplicitPartialD

■ Description

ExplicitPartialD[exp] inserts in exp the definition for LeftRightPartialD[z]
(and LeftRightPartialD2[z]).

■ See also: ExpandPartialD, LeftRightPartialD, LeftRightPartialD2.

■ Examples

```
ExplicitPartialD[ LeftRightPartialD[μ] ]
```

$$\frac{1}{2}\left(\vec{\partial}_\mu - \overleftarrow{\partial}_\mu\right)$$

```
ExplicitPartialD[ LeftRightPartialD2[μ] ]
```

$$\overleftarrow{\partial}_\mu + \vec{\partial}_\mu$$

```
ExplicitPartialD[ LeftRightPartialD[OPEDelta] ]
```

$$\frac{1}{2}\left(\vec{\partial}_\Delta - \overleftarrow{\partial}_\Delta\right)$$

```
16 LeftRightPartialD[OPEDelta]^4
```

$$16\,\overleftrightarrow{\partial}_\Delta^4$$

■ Factor2

■ Description

Factor2[poly] factors a polynomial in a standard way. Factor2 works
(sometimes) better than Factor on polynomials involving rationals with sums in the denominator. Factor2 uses Factor
internally and is in general slower
than Factor. There are four possible settings of the option Method (0,1,2,3).
In general Factor will work faster than Factor2.

```
Options[Factor2]
```

{FactorFull→ False, Method→ 3}

■ See also: Collect2.

■ Examples

```
t1 = (a - x) (b - x)
```

$(a-x)(b-x)$

```
t2 = {Factor2[t1], Factor[t1]}
```

$\{(a-x)(b-x), -(a-x)(x-b)\}$

```
t3 = Expand[(a - b) (a + b)]
```

$a^2 - b^2$

```
Factor[t3]
```

$(a-b)(a+b)$

```
Factor2[t3]
```

$a^2 - b^2$

```
Factor2[t3, FactorFull → True]
```

$(a-b)(a+b)$

```
Clear[t1, t2, t3]
```

■ FAD

■ Description

FAD is the FeynCalc external form of FeynAmpDenominator and denotes an inverse propagator.
FAD[q, q-p,...] is 1/(q^2(q-p)^2..).
FAD[{q1,m}, {q1-p,m},q2, ...] is 1/((q1^2-m^2)((q1-p)^2-m^2) q2^2
...).
Translation into FeynCalc internal form is performed by FeynCalcInternal.

■ **See also: FAD, FCE, FCI, FeynAmpDenominator, FeynAmpDenominatorSimplify, PropagatorDenominator.**

■ Examples

```
FAD[q, p - q]
```

$$\frac{1}{([(p-q)^2])([q^2])}$$

```
FAD[p, {p - q, m}]
```

$$\frac{1}{([p^2])([(p-q)^2-m^2])}$$

```
FAD[q, p - q] // FCI // FCE // InputForm
```

```
FAD[q, p - q]
```

```
FAD[q, p - q] // FCI // InputForm
```

```
FeynAmpDenominator[PropagatorDenominator[Momentum[q, D], 0],
  PropagatorDenominator[Momentum[p, D] - Momentum[q, D], 0]]
```

```
FAD[p] FAD[p - q] // FeynAmpDenominatorCombine // InputForm
```

```
FeynAmpDenominator[PropagatorDenominator[Momentum[p, D], 0],
  PropagatorDenominator[Momentum[p, D] - Momentum[q, D], 0]]
```

■ FCE

■ Description

FCE[exp] translates exp from the internal FeynCalc representation to a short form.

FCE is equivalent to FeynCalcExternal.

■ See also: FeynCalcExternal, FCI, FeynCalcInternal.

■ Examples

```
FCE[{DiracGamma[5], DiracGamma[Momentum[p]]}]
```

$$\{\gamma^5, \gamma \cdot p\}$$

```
% // InputForm
```

```
{GA[5], GS[p]}
```

```
{GA[μ], GAD[ρ], GS[p], SP[p, q], MT[α, β], FV[p, μ]}
```

$\{\gamma^{\mu}, \gamma^{\rho}, \gamma \cdot p, p \cdot q, g^{\alpha\beta}, p^{\mu}\}$

```
% // InputForm
```

```
{GA[μ], GAD[ρ], GS[p], SP[p, q],
  MT[α, β], FV[p, μ]}
```

```
% // FCI
```

$\{\gamma^{\mu}, \gamma^{\rho}, \gamma \cdot p, p \cdot q, g^{\alpha\beta}, p^{\mu}\}$

```
% // InputForm
```

```
{DiracGamma[LorentzIndex[μ]],
  DiracGamma[LorentzIndex[ρ, D], D],
  DiracGamma[Momentum[p]], Pair[Momentum[p], Momentum[q]],
  Pair[LorentzIndex[α], LorentzIndex[β]],
  Pair[LorentzIndex[μ], Momentum[p]]}
```

```
FCE[%] // InputForm
```

```
{GA[μ], GAD[ρ], GS[p], SP[p, q],
  MT[α, β], FV[p, μ]}
```

■ FCI

■ Description

FCI[exp] translates exp into the internal FeynCalc (datatype–) representation.

FCI is equivalent to FeynCalcInternal.

■ See also: FeynCalcExternal, FeynCalcInternal, FCE.

■ Examples

```
{GA[μ], GAD[ρ], GS[p], SP[p, q], MT[α, β], FV[p, μ]}
```

$\{\gamma^{\mu}, \gamma^{\rho}, \gamma \cdot p, p \cdot q, g^{\alpha\beta}, p^{\mu}\}$

```
% // InputForm
```

```
{GA[μ], GAD[ρ], GS[p], SP[p, q],
  MT[α, β], FV[p, μ]}
```

```
% // FCI
```

$\{\gamma^{\mu}, \gamma^{\rho}, \gamma \cdot p, p \cdot q, g^{\alpha\beta}, p^{\mu}\}$

```
% // InputForm
```

```
{DiracGamma[LorentzIndex[μ]],
  DiracGamma[LorentzIndex[ρ, D], D],
  DiracGamma[Momentum[p]], Pair[Momentum[p], Momentum[q]],
  Pair[LorentzIndex[α], LorentzIndex[β]],
  Pair[LorentzIndex[μ], Momentum[p]]}
```

```
FCE[%] // InputForm
```

```
{GA[μ], GAD[ρ], GS[p], SP[p, q],
  MT[α, β], FV[p, μ]}
```

■ FermionSpinSum

■ Description

FermionSpinSum[x] constructs the Traces out of squared amplitudes.

```
Options[FermionSpinSum]
```

{SpinPolarizationSum → Identity, SpinorCollect → False, ExtraFactor → 1}

■ See also: ComplexConjugate, DiracTrace, Tr.

■ Examples

```
t = Spinor[k1, m].DiracSlash[p].GA[5].Spinor[p1, m]
```

$\varphi(\text{k1}, m).\gamma\cdot p\gamma^5.\varphi(\text{p1}, m)$

Notice that SpinorUBar and SpinorU are only input functions. Internally they are converted to Spinor objects (which display as φ).

```
ct = ComplexConjugate[t]
```

$-(\varphi(\text{p1}, m).\gamma^5.\gamma\cdot p.\varphi(\text{k1}, m))$

```
FermionSpinSum[t ct]
```

$-\text{tr}((m+\gamma\cdot\text{k1}).\gamma\cdot p.\gamma^5.(m+\gamma\cdot\text{p1}).\gamma^5.\gamma\cdot p)$

```
% /. DiracTrace -> Tr
```

$-4\,(p^2\,m^2+\text{k1}\cdot\text{p1}\,p^2-2\,\text{k1}\cdot p\,p\cdot\text{p1})$

■ FeynAmpDenominatorCombine

■ Description

FeynAmpDenominatorCombine[expr] expands expr with respect to FeynAmpDenominator and combines products of FeynAmpDenominator in expr into one FeynAmpDenominator.

■ See also: FeynAmpDenominatorSplit.

■ Examples

```
t = FAD[q] FAD[q - p]
```

$$\frac{1}{[q^2]}\,\frac{1}{[(q-p)^2]}$$

```
FeynAmpDenominatorCombine[%] // FCE // InputForm
```

```
FAD[q, -p + q]
```

```
FeynAmpDenominatorSplit[%] // FCE // InputForm
```

```
FAD[q]*FAD[-p + q]
```

■ FeynAmpDenominator

■ Description

FeynAmpDenominator[PropagatorDenominator[...], PropagatorDenominator[...], ...] is the head of the denominators of the propagators, i.e., FeynAmpDenominator[x] the representation of 1/x.

■ See also: FAD, FeynAmpDenominatorSimplify.

■ Examples

```
FeynAmpDenominator[PropagatorDenominator[p, m]]
```

$$\frac{1}{p^2 - m^2}$$

```
FeynAmpDenominator[PropagatorDenominator[p, m],  PropagatorDenominator[p - q, m]]
```

$$\frac{1}{(p^2 - m^2)\,((p-q)^2 - m^2)}$$

```
t = FeynAmpDenominator[PropagatorDenominator[p, m]];

InputForm[t // FCI]

FeynAmpDenominator[PropagatorDenominator[Momentum[p, D], m]]

InputForm[t // FCE]

FAD[{p, m}]

Clear[t];
```

■ FeynAmpDenominatorSimplify

■ Description

FeynAmpDenominatorSimplify[expr] tries to simplify each PropagatorDenominator in a canonical way. FeynAmpDenominatorSimplify[expr, {p1, p2, ...}] simplifies all FeynAmpDenominators in expr in a canonical way, including some translation of momenta. FeynAmpDenominatorSimplify[expr, {p1, p2, ...}, q2] additionally removes integrals with no mass scale.

FDS can be used as an alias.

```
FDS
```

FeynAmpDenominatorSimplify

■ See also: OneLoopSimplify.

■ Examples

The cornerstone of dimensional regularization is that $\int d\hat{}n\,k\, f(k) / k\hat{}(2\,m) = 0$.

```
FeynAmpDenominatorSimplify[f[k] FAD[k, k], k]
```

0

This brings $1/((k-p_1)\hat{}2\,(k-p_2)\hat{}2)$ into a standard form.

```
FeynAmpDenominatorSimplify[FAD[k - p1, k - p2], k]
```

$$\frac{1}{k^2\,(k-p_1+p_2)^2}$$

```
t = FeynAmpDenominatorSimplify[FAD[k - p1, k - p2] SPD[k, k], k]
```

$$\frac{k^2}{k^2\,(k-p_1+p_2)^2}+\frac{2\,k\cdot p_2}{k^2\,(k-p_1+p_2)^2}+\frac{p_2{}^2}{k^2\,(k-p_1+p_2)^2}$$

```
r = SPC[t, k, FDS → True]
```

$$\frac{p_2{}^2}{k^2\,(k-p_1+p_2)^2}-\frac{2\,k\cdot p_2}{k^2\,(k+p_1-p_2)^2}$$

```
OneLoopSimplify[r, k]
```

$$\frac{p_1\cdot p_2}{k^2\,(k-p_1+p_2)^2}$$

```
FDS[FAD[k - p1, k - p2] SPD[k, OPEDelta]^2, k]
```

$$\frac{(k\cdot\Delta)^2}{k^2\,(k-\text{p1}+\text{p2})^2}+\frac{2\,k\cdot\Delta\,\Delta\cdot\text{p2}}{k^2\,(k-\text{p1}+\text{p2})^2}+\frac{(\Delta\cdot\text{p2})^2}{k^2\,(k-\text{p1}+\text{p2})^2}$$

```
Clear[t, r]
```

■ FeynAmpDenominatorSplit

■ Description

FeynAmpDenominatorSplit[expr] splits all FeynAmpDenominator[a,b,...] in expr into FeynAmpDenominator[a]*FeynAmpDenominator[b]...

FeynAmpDenominatorSplit[expr, q1] splits all FeynAmpDenominator in expr into a product of two, one containing q1 and other momenta, the second without q1.

■ See also: FeynAmpDenominatorCombine.

■ Examples

```
t = FAD[q1, q1 - p, q1 - q2, q2, q2 - p] // FCI
```

$$\frac{1}{q_1^2\,(q_1-p)^2\,(q_1-q_2)^2\,q_2^2\,(q_2-p)^2}$$

```
t // Head
```

FeynAmpDenominator

```
FeynAmpDenominatorSplit[t]
```

$$\frac{1}{q_1^2\,(q_1-p)^2\,(q_1-q_2)^2\,q_2^2\,(q_2-p)^2}$$

```
% // FCE // InputForm
```

```
FAD[q1]*FAD[-p + q1]*FAD[q1 - q2]*FAD[q2]*FAD[-p + q2]
```

```
FeynAmpDenominatorSplit[t, q1] // FCE // InputForm
```

```
FAD[q2, -p + q2]*FAD[q1, -p + q1, q1 - q2]
```

```
FeynAmpDenominatorCombine[%] // FCE // InputForm
```

```
FAD[q1, q2, q1 - q2, -p + q1, -p + q2]
```

```
Clear[t]
```

■ FeynAmp

■ Description

FeynAmp[q,amp] is the head of a Feynman amplitude. amp denotes the analytical expression for the amplitude and q is the integration variable. FeynAmp[q1,q2, amp] denotes a two–loop amplitude.

FeynAmp has no functional properties and serves just as a head. There are however special typesetting rules attached.

■ See also: Amplitudes.

■ Examples

```
Cell["This is a 1-loop gluon selfenergy amplitude.",
  "Text"]
```

```
FeynAmp[q, GV[p, μ, a, q - p, α, c, -q, β, e]
  GP[p - q, α, c, ρ, d] GV[-p, ν, b, p - q, ρ, d, q, σ, f] GP[q, β, e, σ, f]]
```

$$\int \frac{d^D q}{(2\pi)^D} \left(-\frac{g_s^2 V^{\nu\rho\sigma}(-p, p-q, q) V^{\mu\alpha\beta}(p, q-p, -q) g^{\alpha\rho} g^{\beta\sigma} \delta_{cd} \delta_{ef} f_{ace} f_{bdf}}{(p-q)^2 q^2} \right)$$

This is a generic 2–loop amplitude.

```
FeynAmp[q1, q2, anything goes]
```

$$\int \frac{d^D q_1}{(2\pi)^D} \int \frac{d^D q_2}{(2\pi)^D} \left(\text{anything goes} \right)$$

■ FeynCalc2FORM

■ Description

FeynCalc2FORM[expr] displays expr in FORM syntax.
FeynCalc2FORM[file, x] writes x in FORM syntax to a file.
FeynCalc2FORM[file, x==y] writes x=y to a file in FORM syntax.

■ See also: FORM2FeynCalc.

```
Options[FeynCalc2FORM]
```

{EpsDiscard→ False, FORMEpilog→ , FORMProlog→ writestatistics;,
Replace→ {α → al, β → be, γ → ga, δ → de, μ → mu, ν → nu, ρ → ro, σ → si}, TraceDimension→ 4}

■ Examples

```
MT[μ, ν] FV[p, ρ] y^2 / d
```

$$\frac{y^2\, p^\rho\, g^{\mu\nu}}{d}$$

```
FeynCalc2FORM[%];
```

(y^2*d_(mu,nu)*p(ro))/d

```
LC[α, β, δ, ρ]
```

$\epsilon^{\alpha\beta\delta\rho}$

```
FeynCalc2FORM[%];
```

-i_*e_(al,be,de,ro)

```
DiracTrace[GA[μ, ν, ρ, σ]]
```

$\mathrm{tr}(\gamma^\mu\, \gamma^\nu\, \gamma^\rho\, \gamma^\sigma)$

```
FeynCalc2FORM[%];
```

g_(1,mu)*g_(1,nu)*g_(1,ro)*g_(1,si)

```
DiracTrace[GA[μ, ν]] DiracTrace[GA[μ, ρ]]
```

$\mathrm{tr}(\gamma^\mu\, \gamma^\nu)\, \mathrm{tr}(\gamma^\mu\, \gamma^\rho)$

```
FeynCalc2FORM[%];
```

g_(1,mu)*g_(1,nu)*g_(2,mu)*g_(2,ro)

```
FeynCalc2FORM["fc2ftest.f", MT[μ, ν] FV[p, μ]];
```

```
TableForm[ReadList["fc2ftest.f",String]]
```

```
d_(mu,nu)*p(mu)
```

```
t = Tr[GA[μ, ν, ρ, σ].GS[p, q]]
```

$$4\,(q^{\mu}\,p^{\nu}\,g^{\rho\sigma} - p^{\mu}\,q^{\nu}\,g^{\rho\sigma} + g^{\mu\nu}\,p\cdot q\,g^{\rho\sigma} - q^{\mu}\,g^{\nu\sigma}\,p^{\rho} + g^{\mu\sigma}\,q^{\nu}\,p^{\rho} + p^{\mu}\,g^{\nu\sigma}\,q^{\rho} - g^{\mu\sigma}\,p^{\nu}\,q^{\rho} + q^{\mu}\,g^{\nu\rho}\,p^{\sigma} - g^{\mu\rho}\,q^{\nu}\,p^{\sigma} + g^{\mu\nu}\,q^{\rho}\,p^{\sigma} - p^{\mu}\,g^{\nu\rho}\,q^{\sigma} + g^{\mu\rho}\,p^{\nu}\,q^{\sigma} - g^{\mu\nu}\,p^{\rho}\,q^{\sigma} + g^{\mu\sigma}\,g^{\nu\rho}\,p\cdot q - g^{\mu\rho}\,g^{\nu\sigma}\,p\cdot q)$$

```
FeynCalc2FORM["fc2ftest.f", L == t];
```

```
TableForm[ReadList["fc2ftest.f", String]]
```

```
Indices mu,nu,ro,si;
Vectors p,q;

writestatistics;

Local L = (
4*(d_(mu,si)*d_(nu,ro)*q.p-d_(mu,ro)*d_(nu,si)*q.p+d_(mu,nu)*d_(ro,si)*q.p+
d_(ro,si)*p(nu)*q(mu)-d_(nu,si)*p(ro)*q(mu)+d_(nu,ro)*p(si)*q(mu)-
d_(ro,si)*p(mu)*q(nu)+d_(mu,si)*p(ro)*q(nu)-d_(mu,ro)*p(si)*q(nu)+
d_(nu,si)*p(mu)*q(ro)-d_(mu,si)*p(nu)*q(ro)+d_(mu,nu)*p(si)*q(ro)-
d_(nu,ro)*p(mu)*q(si)+d_(mu,ro)*p(nu)*q(si)-d_(mu,nu)*p(ro)*q(si)) );

Print;
.end
```

```
If[FileNames["fc2ftest.f"], DeleteFile["fc2ftest.f"]];
```

■ FeynCalcExternal

■ Description

FeynCalcExternal[exp] translates exp from the internal FeynCalc representation to a shorthand form.

■ See also: FeynCalcInternal.

■ Examples

```
FeynCalcExternal[DiracGamma[5]]
```

$$\gamma^5$$

```
% // InputForm
```

```
GA[5]
```

```
{GA[μ], GAD[ρ], GS[p], SP[p, q], MT[α, β], FV[p, μ]}
```

$\{\gamma^{\mu}, \gamma^{\rho}, \gamma \cdot p, p \cdot q, g^{\alpha\beta}, p^{\mu}\}$

```
% // InputForm
```

```
{GA[μ], GAD[ρ], GS[p], SP[p, q],
  MT[α, β], FV[p, μ]}
```

```
% // FeynCalcInternal
```

$\{\gamma^{\mu}, \gamma^{\rho}, \gamma \cdot p, p \cdot q, g^{\alpha\beta}, p^{\mu}\}$

```
% // InputForm
```

```
{DiracGamma[LorentzIndex[μ]],
  DiracGamma[LorentzIndex[ρ, D], D],
  DiracGamma[Momentum[p]], Pair[Momentum[p], Momentum[q]],
  Pair[LorentzIndex[α], LorentzIndex[β]],
  Pair[LorentzIndex[μ], Momentum[p]]}
```

```
FeynCalcExternal[%] // InputForm
```

```
{GA[μ], GAD[ρ], GS[p], SP[p, q],
  MT[α, β], FV[p, μ]}
```

■ FeynCalcInternal

■ Description

FeynCalcInternal[*exp*] translates *exp* into the internal FeynCalc (abstract data–type) representation.

■ See also: FeynCalcExternal, FCI, FCE.

■ Examples

```
{GA[μ], GAD[ρ], GS[p], SP[p, q], MT[α, β], FV[p, μ]}
```

$\{\gamma^{\mu}, \gamma^{\rho}, \gamma \cdot p, p \cdot q, g^{\alpha\beta}, p^{\mu}\}$

```
% // InputForm
```

```
{GA[μ], GAD[ρ], GS[p], SP[p, q],
  MT[α, β], FV[p, μ]}
```

```
% // FeynCalcInternal
```

$\{\gamma^{\mu}, \gamma^{\rho}, \gamma \cdot p, p \cdot q, g^{\alpha\beta}, p^{\mu}\}$

```
% // InputForm
```

```
{DiracGamma[LorentzIndex[μ]],
  DiracGamma[LorentzIndex[ρ, D], D],
  DiracGamma[Momentum[p]], Pair[Momentum[p], Momentum[q]],
  Pair[LorentzIndex[α], LorentzIndex[β]],
  Pair[LorentzIndex[μ], Momentum[p]]}
```

```
FeynCalcExternal[%] // InputForm
```

```
{GA[μ], GAD[ρ], GS[p], SP[p, q],
  MT[α, β], FV[p, μ]}
```

```
FCI[{SD[a, b], SUND[a, b, c], SUNF[a, b, c], FAD[q], LC[μ, ν, ρ, σ]}]
```

$$\left\{\delta_{ab},\ d_{abc},\ f_{abc},\ \frac{1}{q^2},\ \epsilon^{\mu\nu\rho\sigma}\right\}$$

```
% // InputForm
```

```
{SUNDelta[SUNIndex[a], SUNIndex[b]],
  SUND[SUNIndex[a], SUNIndex[b], SUNIndex[c]],
  SUNF[SUNIndex[a], SUNIndex[b], SUNIndex[c]],
  FeynAmpDenominator[PropagatorDenominator[Momentum[q, D],
    0]], Eps[LorentzIndex[μ], LorentzIndex[ν],
   LorentzIndex[ρ], LorentzIndex[σ]]}
```

■ FeynRule

■ Description

FeynRule[lag,{fields}] derives the Feynman rule corresponding to the field configuration fields of the lagrangian lag.

```
Options[FeynRule]
```

```
{Anti5 → −∞, Contract → False, Factor1 → False, FinalSubstitutions → {},
  PartialD → RightPartialD, Schouten → False, ZeroMomentumInsertion → True}
```

FeynRule does not calculate propagator Feynman rules.

The option ZeroMomentumInsertion can be used for twist–2 and higher twist operators.

■ See also: Lagrangian.

■ Examples

```
gou = Lagrangian["ogu"]
```

$$\frac{1}{2}\, i^{m-1} \left(F^a_{\alpha\Delta}\, (D^{\Delta b})^{m-2}\, F^{b}_{\alpha\Delta}\right)$$

```
gop = Lagrangian["ogp"]
```

$$\frac{1}{2}\, i^m \left(\epsilon^{\alpha\beta\gamma\Delta}\, F^a_{\beta\gamma}\, (D^{ab}_{\Delta})^{m-2}\, F^b_{\alpha\Delta}\right)$$

```
Explicit[gop]
```

$$\frac{1}{2}\, i^m \left(\epsilon^{\alpha\beta\gamma\Delta}\, (\partial_\beta A^a_\gamma - \partial_\gamma A^a_\beta + g_s\, (A^{\text{b2}}_\beta A^{\text{c11}}_\gamma)\, f_{\text{ab2c11}})\, (D^{ab}_{\Delta})^{m-2}\, (\partial_\alpha A^b_\Delta - \partial_\Delta A^b_\alpha + g_s\, (A^{\text{b3}}_\alpha A^{\text{c12}}_\Delta)\, f_{\text{bb3c12}})\right)$$

```
Cases2[%, QuantumField] // InputForm
```

```
{QuantumField[GaugeField, LorentzIndex[α],
   SUNIndex[b3]], QuantumField[GaugeField,
   LorentzIndex[β], SUNIndex[b2]],
  QuantumField[GaugeField, LorentzIndex[γ],
   SUNIndex[c11]], QuantumField[GaugeField,
   Momentum[OPEDelta], SUNIndex[c12]],
  QuantumField[PartialD[LorentzIndex[α]],
   GaugeField, Momentum[OPEDelta], SUNIndex[b]],
  QuantumField[PartialD[LorentzIndex[β]], GaugeField,
   LorentzIndex[γ], SUNIndex[a]],
  QuantumField[PartialD[LorentzIndex[γ]],
   GaugeField, LorentzIndex[β], SUNIndex[a]],
  QuantumField[PartialD[Momentum[OPEDelta]], GaugeField,
   LorentzIndex[α], SUNIndex[b]]}
```

■ 2-gluon Feynman rules (unpolarized)

```
fi = {QuantumField[GaugeField,{μ},{a}][p],
      QuantumField[GaugeField,{ν},{b}][q]}
```

$$\{A^a_\mu(p),\, A^b_\nu(q)\}$$

```
f2u = FullSimplify /@ Factor2[FeynRule[gou, fi, ZeroMomentumInsertion → False]]
```

$$\frac{1}{2\,(\Delta\cdot p)^2\,(\Delta\cdot q)^2}\,(((\Delta\cdot q)^2\,(\Delta\cdot p)^m + (\Delta\cdot q)^m\,(\Delta\cdot p)^2)\,(q^\mu\,\Delta^\nu\,\Delta\cdot p - g^{\mu\nu}\,\Delta\cdot q\,\Delta\cdot p + \Delta^\mu\,(p^\nu\,\Delta\cdot q - \Delta^\nu\,p\cdot q))\,\delta_{ab})$$

■ 2-gluon Feynman rules (polarized)

```
fi = {QuantumField[GaugeField, {μ}, {a}][p], QuantumField[GaugeField, {ν}, {b}][q]}
```

$$\{A^a_\mu(p),\, A^b_\nu(q)\}$$

```
f2p = FullSimplify/@Factor2[FeynRule[gop, fi, ZeroMomentumInsertion → False]]
```

$$-\frac{1}{(\Delta\cdot p)^2\,(\Delta\cdot q)^2}\,(i\,((\Delta\cdot p)^m\,(\epsilon^{\nu pq\Delta}\,\Delta^\mu + \epsilon^{\mu\nu q\Delta}\,\Delta\cdot p)\,(\Delta\cdot q)^2 - (\Delta\cdot p)^2\,(\Delta\cdot q)^m\,(\epsilon^{\mu pq\Delta}\,\Delta^\nu + \epsilon^{\mu\nu p\Delta}\,\Delta\cdot q))\,\delta_{ab})$$

```
Factor2[Calc[f2p /. p → -q]]
```

$$-i\,(1-(-1)^m)\,\epsilon^{\mu\nu q\Delta}\,(\Delta\cdot q)^{m-1}\,\delta_{ab}$$

Compare with the Feynman rule tabulated in Twist2GluonOperator.

```
Twist2GluonOperator[q, {μ, a}, {ν, b},     Polarization → 1, Explicit → True]
```

$$i\,\epsilon^{\mu\nu\Delta q}\,(1-(-1)^m)\,(\Delta\cdot q)^{m-1}\,\delta_{ab}$$

■ quark–quark Feynman rule (unpolarized)

```
qo = Lagrangian["oqu"]
```

$$i^m (\bar{\psi}\, \gamma\cdot\Delta\, D_\Delta{}^{m-1}\, \psi)$$

This is the list of the fermionic fields, with p and q are flowing into the vertex.

```
qf = {QuantumField[QuarkField][p],
      QuantumField[AntiQuarkField][q]}
```

$$\{\psi(p), \bar{\psi}(q)\}$$

For non–zero momentum insertion it is essential to let the partial derivative operator act on both sides. This can be achieved by setting the option PartialD to LeftRightPartialD.

```
n2 = FeynRule[qo, qf, ZeroMomentumInsertion → False, PartialD → LeftRightPartialD]
```

$$2^{1-m}\, \gamma\cdot\Delta\, (\Delta\cdot p - \Delta\cdot q)^{m-1}$$

```
PowerSimplify[%]
```

$$2^{1-m}\, \gamma\cdot\Delta\, (\Delta\cdot p - \Delta\cdot q)^{m-1}$$

For $q = -p$ this reduces to the zero–momentum insertion Feynman rule.

```
Calc[n2 /. q → -p]
```

$$\gamma\cdot\Delta\, (\Delta\cdot p)^{m-1}$$

This can also be derived directly.

```
FeynRule[qo, Reverse[qf], ZeroMomentumInsertion → True]
```

$$\gamma\cdot\Delta\, (\Delta\cdot p)^{m-1}$$

■ quark–quark-gluon Feynman rule (unpolarized)

```
qo = Lagrangian["oqu"]
```

$$i^m (\bar{\psi}\, \gamma\cdot\Delta\, D_\Delta{}^{m-1}\, \psi)$$

```
qgf = {QuantumField[QuarkField][p],
       QuantumField[AntiQuarkField][q],
       QuantumField[GaugeField, {μ}, {a}][r]}
```

$$\{\psi(p), \bar{\psi}(q), A^a_\mu(r)\}$$

```
n3 = FeynRule[qo, qgf, ZeroMomentumInsertion → False, PartialD → LeftRightPartialD]
```

$$2^{2-m}\, g_s\, \gamma\cdot\Delta \left(\sum_{i=0}^{m-2} (\Delta\cdot p - \Delta\cdot q - \Delta\cdot r)^{-i+m-2}\, (\Delta\cdot p - \Delta\cdot q + \Delta\cdot r)^i\right) \Delta^\mu\, T_a$$

```
Trick[OPESumSimplify[Trick[Calc[n3 /. r → -p - q]]]]
```

$$g_s \left(\gamma\cdot\Delta \sum_{i=0}^{m-2} (-1)^i (\Delta\cdot p)^{-i+m-2} (\Delta\cdot q)^i\right) \Delta^\mu \; T_a$$

```
Twist2QuarkOperator[{p}, {q}, {r, μ, a}, Polarization → 0]
```

$$g_s \, (\gamma\cdot\Delta \; T_a) \left(\sum_{i=0}^{m-2} (-1)^i (\Delta\cdot p)^{-i+m-2} (\Delta\cdot q)^i\right) \Delta^\mu$$

■ quark–quark–gluon Feynman rule (polarized)

```
qp = Lagrangian["oqp"]
```

$$i^m \, (\overline{\psi}\, \gamma^5 \; \gamma\cdot\Delta \; {D_\Delta}^{m-1} \; \psi)$$

```
qgf = {QuantumField[QuarkField][p],
       QuantumField[AntiQuarkField][q],
       QuantumField[GaugeField,{μ},{a}][r]}
```

$$\{\psi(p), \overline{\psi}(q), A_\mu^a(r)\}$$

```
n3 = FeynRule[qp, qgf, ZeroMomentumInsertion → False, PartialD → LeftRightPartialD]
```

$$2^{2-m} \, g_s \, (T_a \, \gamma^5 \; \gamma\cdot\Delta) \left(\sum_{i=0}^{m-2} (\Delta\cdot p - \Delta\cdot q - \Delta\cdot r)^{-i+m-2} \, (\Delta\cdot p - \Delta\cdot q + \Delta\cdot r)^i\right) \Delta^\mu$$

```
Trick[OPESumSimplify[Trick[Calc[n3 /. r → -p - q]]]]
```

$$-g_s \left(\gamma\cdot\Delta \; \gamma^5 \sum_{i=0}^{m-2} (-1)^i (\Delta\cdot p)^{-i+m-2} (\Delta\cdot q)^i\right) \Delta^\mu \; T_a$$

```
Twist2QuarkOperator[{p}, {q}, {r, μ, a}, Polarization → 1]
```

$$-g_s \, (\gamma\cdot\Delta \; \gamma^5 \; T_a) \left(\sum_{i=0}^{m-2} (-1)^i (\Delta\cdot p)^{-i+m-2} (\Delta\cdot q)^i\right) \Delta^\mu$$

■ quark–quark–gluon–ghost Feynman rule (unpolarized)

```
qo = Lagrangian["oqu"]
```

$$i^m \, (\overline{\psi}\, \gamma\cdot\Delta \; {D_\Delta}^{m-1} \; \psi)$$

```
qggf = {QuantumField[QuarkField][p],
        QuantumField[AntiQuarkField][q],
        QuantumField[GaugeField,{μ},{a}][r],
        QuantumField[GaugeField,{ ν},{b}][s]}
```

$$\{\psi(p), \overline{\psi}(q), A_\mu^a(r), A_\nu^b(s)\}$$

```
n4 = FeynRule[qo, qggf, ZeroMomentumInsertion → True]
```

$$g_s^2\,(T_b\,\gamma\cdot\Delta\,T_a)\left(\sum_{j=0}^{m-3}\sum_{i=0}^{j}(-1)^j\,(\Delta\cdot p)^{-j+m-3}\,(\Delta\cdot q)^i\,(\Delta\cdot q+\Delta\cdot s)^{j-i}\right)\Delta^\mu\,\Delta^\nu\,-$$

$$(-1)^m\,g_s^2\,(T_a\,\gamma\cdot\Delta\,T_b)\left(\sum_{j=0}^{m-3}\sum_{i=0}^{j}(-1)^j\,(\Delta\cdot p)^{i}\,(\Delta\cdot q)^{-j+m-3}\,(\Delta\cdot p+\Delta\cdot s)^{j-i}\right)\Delta^\mu\,\Delta^\nu$$

```
t4 = Twist2QuarkOperator[{p}, {q}, {r, μ, a}, {s, ν, b}, Polarization → 0]
```

$$-(-1)^m\,g_s^2$$
$$\left(\gamma\cdot\Delta\left((T_a\,T_b)\left(\sum_{i=0}^{m-3}\sum_{j=0}^{i}(-(\Delta\cdot p))^{-i+m-3}\,(\Delta\cdot q)^j\,(\Delta\cdot q+\Delta\cdot r)^{i-j}\right)+(T_b\,T_a)\left(\sum_{i=0}^{m-3}\sum_{j=0}^{i}(-(\Delta\cdot p))^{-i+m-3}\,(\Delta\cdot q)^j\,(\Delta\cdot q+\Delta\cdot s)^{i-j}\right)\right)\right)$$
$$\Delta^\mu\,\Delta^\nu$$

```
Calc[(n4 - t4) /. OPEm → 5 /. s → -p - q - r /. D → 4]
```

0

In general equality can be shown by Timing[Factor2[
FCE[Calc[ChangeDimension[FCE[OPESumExplicit[n4-t4]],4]/.s→-p-q-r]]]]
but it is a little bit slow...

■ QCD vertices

```
fii = {QuantumField[GaugeField, {μ}, {a}][p],
  QuantumField[GaugeField, {ν}, {b}][q],   QuantumField[GaugeField, {ρ}, {c}][r]}
```

$\{A_\mu^a(p),\ A_\nu^b(q),\ A_\rho^c(r)\}$

```
g3 = FeynRule[Lagrangian["QCD"], fii]
```

$g_s\,((q^\mu - r^\mu)\,g^{\nu\rho} - g^{\mu\rho}\,(p^\nu - r^\nu) + g^{\mu\nu}\,(p^\rho - q^\rho))\,f_{abc}$

```
GluonVertex[{p, μ, a}, {q, ν, b}, {r, ρ, c}, Explicit -> True]
```

$g_s\,(((q-r)^\mu)\,g^{\nu\rho} + g^{\mu\rho}\,((r-p)^\nu) + g^{\mu\nu}\,((p-q)^\rho))\,f_{abc}$

```
Calc[g3 - ChangeDimension[%, 4]]
```

0

```
fi4 = {QuantumField[GaugeField, {μ}, {a}][p],
  QuantumField[GaugeField, {ν}, {b}][q],
  QuantumField[GaugeField, {ρ}, {c}][r],          QuantumField[GaugeField, {σ}, {d}][s]}
```

$\{A_\mu^a(p),\ A_\nu^b(q),\ A_\rho^c(r),\ A_\sigma^d(s)\}$

```
g4 = FeynRule[Lagrangian["QCD"], fi4]
```

$i\,(g^{\mu\rho}\,g^{\nu\sigma} - g^{\mu\nu}\,g^{\rho\sigma})\,f_{adsi1}\,f_{bcsi1}\,g_s^2 + i\,(g^{\mu\sigma}\,g^{\nu\rho} - g^{\mu\nu}\,g^{\rho\sigma})\,f_{acsi1}\,f_{bdsi1}\,g_s^2 + i\,(g^{\mu\sigma}\,g^{\nu\rho} - g^{\mu\rho}\,g^{\nu\sigma})\,f_{absi1}\,f_{cdsi1}\,g_s^2$

```
GluonVertex[{p, μ, a}, {q, ν, b}, {r, ρ, c}, {s, σ, d}, Explicit -> True]
```

$$-i\, g_s^2\, ((g^{\mu\nu}\, g^{\rho\sigma} - g^{\mu\rho}\, g^{\nu\sigma})\, f_{adu8}\, f_{bcu8} + (g^{\mu\nu}\, g^{\rho\sigma} - g^{\mu\sigma}\, g^{\nu\rho})\, f_{acu8}\, f_{bdu8} + (g^{\mu\rho}\, g^{\nu\sigma} - g^{\mu\sigma}\, g^{\nu\rho})\, f_{abu8}\, f_{cdu8})$$

```
Calc[g4 - ChangeDimension[%, 4]]
```

0

```
Clear[f2p, f2u, f3, f32, fi, fi4, fii, g3, g4, gop,
 gou, n3, nf3, np2, npf3, p33, pf3, pn3, pqo, qf, qgf, qo, t4]
```

■ FieldStrength

■ Description

FieldStrength[μ, ν, a] is the field strength tensor
$\partial_\mu A_\nu^a - \partial_\nu A_\mu^a + g_s\, A_\mu^{b1}\, A_\nu^{c1}\, f^{\wedge}(a\,b1\,c1)$.
FieldStrength[μ, ν] is the field strength tensor $\partial_\mu A_\nu - \partial_\nu A_\mu$.
The name of the field (A) and the coupling constant (g) can be set through the options or by additional arguments.
The first two indices are interpreted as type LorentzIndex, except OPEDelta, which is converted to Momentum[OPEDelta].

```
Options[FieldStrength]
```

{CouplingConstant → g_s, Explicit → False, IndexPosition → {0, 0}, Symbol → F, QuantumField → A}

■ See also: QuantumField.

■ Examples

```
FieldStrength[μ, ν]
```

$F_{\mu\nu}$

```
FieldStrength[μ, ν, a]
```

$F_{\mu\nu}^a$

```
FieldStrength[μ, ν, Explicit → True]
```

$\partial_\mu A_\nu - \partial_\nu A_\mu$

```
FieldStrength[μ, ν, a, Explicit → True]
```

$\partial_\mu A_\nu^a - \partial_\nu A_\mu^a + g_s\, (A_\mu^{b12}\, A_\nu^{c55})\, f_{ab12c55}$

```
InputForm[FieldStrength[μ, ν, Explicit → True]]
```

```
QuantumField[PartialD[LorentzIndex[μ]], GaugeField,
   LorentzIndex[ν]] -
  QuantumField[PartialD[LorentzIndex[ν]], GaugeField,
   LorentzIndex[μ]]
```

```
InputForm[FieldStrength[μ, OPEDelta, Explicit → True]]
```

```
QuantumField[PartialD[LorentzIndex[μ]], GaugeField,
   Momentum[OPEDelta]] -
  QuantumField[PartialD[Momentum[OPEDelta]], GaugeField,
   LorentzIndex[μ]]
```

```
FieldStrength[μ, ν, a, CouplingConstant → -Gstrong, Explicit → True]
```

$$\partial_\mu A_\nu^a - \partial_\nu A_\mu^a - g_s \left(A_\mu^{b13} A_\nu^{c56}\right) f_{ab13c56}$$

■ FORM2FeynCalc

■ Description

FORM2FeynCalc[expr] translates the FORM expr into FeynCalc notation.
FORM2FeynCalc[file] translates the FORM expresssions in file into FeynCalc notation. FORM2FeynCalc[file, x1, x2, ...] reads in a file in FORM–format and translates the assignments for the variables a, b, ... into FeynCalc syntax. If the option Set is True, the variables x1, x2 are assigned to the right hand sides defined in the FORM –file.

■ See also: FeynCalc2FORM.

```
Options[FORM2FeynCalc]
```

```
{Dimension → 4, FinalSubstitutions → {}, Dot → Dot, HoldForm → True,
  LorentzIndex → {mu, nu, al, be}, Set → False, Replace → {}, Vectors → Automatic}
```

■ Examples

```
FORM2FeynCalc["p.q + 2*x m^2"]
```

$$2\,(x\,m^2) + p\cdot q$$

```
% // InputForm
```

```
2*x . m^2 + SP[p, q]
```

Functions are automatically converted right, but bracketed expressions need to be substituted explicitly.

```
FORM2FeynCalc["x +f(z)+ log(x)^2+[li2(1-x)]",
 Replace → {"[li2(1-x)]" -> "PolyLog[2,1-x]"}]
```

$$\log^2(x) + x + f(z) + \text{Li}_2(1-x)$$

```
% // InputForm
```

```
x + f[z] + Log[x]^2 + PolyLog[2, 1 - x]
```

```
FORM2FeynCalc["x + [(1)]*y "]
```

$$x + \text{Hold}(1)\,y$$

```
FORM2FeynCalc["p(mu)*q(nu)+d_(mu,nu)"]
```

$p\text{mu} q^{\text{nu}} + g^{\text{munu}}$

```
% // InputForm
```

```
p . mu . FV[q, nu] + MT[mu, nu]
```

```
FORM2FeynCalc["p(mu)*q(nu)+d_(mu,nu)", Replace -> {mu -> μ, nu -> ν}]
```

$p\mu q^{\nu} + g^{\mu\nu}$

```
FORM2FeynCalc["i_*az*bz*aM^2*D1*[(1)]*b_G1 * ( 4*eperp(mu,nu)*avec.bvec*blam \n\n)"]
```

$4 i \text{azbzaM}^2 \text{D1Hold}(1) \text{b\$G1eperp}(\text{mu}, \text{nu}) \text{avec} \cdot \text{bvecblam}$

■ FourDivergence

■ Description

FourDivergence[exp,FourVector[p,mu]] calculates the partial derivative of exp w.r.t. p(mu). FourDivergence[exp, FourVector[p,mu],FourVector[p,nu],...] gives the multiple derivative.

■ See also: RussianTrick.

■ Examples

```
t = ScalarProduct[p, q]
```

$p \cdot q$

```
FourDivergence[t, FourVector[q, μ]]
```

p^{μ}

```
t = ScalarProduct[p - k, q]
```

$(p - k) \cdot q$

```
FourDivergence[t, FourVector[k - p, μ]]
```

0

```
Clear[t]
```

■ FourLaplacian

■ Description

FourLaplacian[exp, p, q] is $\partial / \partial p_\mu \; \partial / \partial q_\mu$ exp.

```
Options[FourLaplacian]
```

{Dimension → *D*}

■ See also: FourDivergence, RussianTrick.

■ Examples

```
SP[q, q]
```

q^2

```
FourLaplacian[%, q, q]
```

$2\,D$

```
SOD[q]^OPEm FAD[q, q - p] // FCI
```

$$\frac{(\Delta \cdot q)^m}{q^2\,(q-p)^2}$$

```
FourLaplacian[%, q, q]
```

$$\frac{4\,m\Delta \cdot p\,(\Delta \cdot q)^{m-1}}{q^2\,(q-p)^2\,(q-p)^2} - \frac{2\,D\,(\Delta \cdot q)^m}{q^2\,q^2\,(q-p)^2} - \frac{4\,m\,(\Delta \cdot q)^m}{q^2\,q^2\,(q-p)^2} + \frac{12\,(\Delta \cdot q)^m}{q^2\,q^2\,(q-p)^2} - \frac{2\,D\,(\Delta \cdot q)^m}{q^2\,(q-p)^2\,(q-p)^2} - \frac{4\,m\,(\Delta \cdot q)^m}{q^2\,(q-p)^2\,(q-p)^2} + \frac{12\,(\Delta \cdot q)^m}{q^2\,(q-p)^2\,(q-p)^2} - \frac{4\,(\Delta \cdot q)^m\,p^2}{q^2\,q^2\,(q-p)^2\,(q-p)^2}$$

■ FourVector

■ Description

FourVector[p, μ] is the four–dimensional vector p with Lorentz index μ.
A vector with space–time Dimension D is obtained by supplying the option Dimension → D.

■ See also: FV, FVD, Pair.

■ Examples

```
FourVector[p, μ]
```

p^{μ}

```
FourVector[p - q, μ]
```

$(p-q)^{\mu}$

```
InputForm[FourVector[p, μ]]
```

```
Pair[LorentzIndex[μ], Momentum[p]]
```

```
InputForm[FourVector[p, μ, Dimension → D]]
```

```
Pair[LorentzIndex[μ, D], Momentum[p, D]]
```

```
InputForm[FCE[FourVector[p, μ]]]
```

```
FV[p, μ]
```

There is no special function to expand momenta in FourVector.
Since FourVector is turned into Pair internally ExpandScalarProduct may be used.

```
ExpandScalarProduct[FourVector[p - q, μ]]
```

$p^{\mu}-q^{\mu}$

■ FreeQ2

■ Description

FreeQ2[expr,{form1, form2,...}] yields True if expr does not contain any occurence of form1, form2,... and False otherwise.
FreeQ2[expr,form] is the same as FreeQ[expr,form].

■ See also: SelectFree, SelectNotFree.

■ Examples

```
FreeQ2[x + f[x] + y, {a, x}]
```

False

```
FreeQ2[x + f[x] + y, {a, b}]
```

True

```
FreeQ2[x, y]
```

True

```
FreeQ2[f[x], f]
```

False

■ FromTFi

■ Description

FromTFi[expr,q1, q2, p] translates the TFi notation from the TARCER package to the usual FeynCalc notation. See TFi for details on the conventions.

■ See also: TFi, ToTFi.

■ Examples

```
FAD[q1, q1 - p, {q2, M}, {q2 - p, m}, q1 - q2] // ToTFi
```

$$\mathbf{F}^{(D)}_{\{1,0\}\{1,M\}\{1,0\}\{1,m\}\{1,0\}}$$

```
FromTFi[TFi[D, SPD[p, p], SOD[p], {{1, 0}, {1, M}, {1, 0}, {1, m}, {1, 0}}], q1, q2, p]
```

$$\frac{1}{q_2^2\,((q_1-p)^2-M^2)\,(q_2-p)^2\,(q_1-q_2)^2\,(q_1^2-m^2)}$$

```
FromTFi[
 TFi[D, SPD[p, p], SOD[p], {0, 1}, {{1, m}, {1, M}, {1, 0}, {1, m}, {1, 0}}], q1, q2, p]
```

$$\frac{\Delta\cdot q_2}{(q_1-q_2)^2\,(q_2^2-M^2)\,((q_2-p)^2-m^2)\,(q_1^2-m^2)\,(q_1-p)^2}$$

■ FRH

■ Description

FRH[exp_]:= FixedPoint[ReleaseHold,exp], i.e., FRH removes all HoldForm and Hold in exp.

■ See also: Isolate.

■ Examples

```
Hold[1 - 1 - Hold[2 - 2]]
```

Hold(−Hold(2 − 2) + 1 − 1)

```
FRH[%]
```

0

```
Isolate[Solve[x^3 - x - 1 == 0], x, IsolateNames -> H]
```

$\{\{x \to H(3)\}, \{x \to H(6)\}, \{x \to H(7)\}\}$

```
FRH[H[3]]
```

$$\frac{1}{3}\sqrt[3]{\frac{27}{2}-\frac{3\sqrt{69}}{2}}+\frac{\sqrt[3]{\frac{1}{2}\left(9+\sqrt{69}\right)}}{3^{2/3}}$$

■ FunctionalD

■ Description

FunctionalD[exp, {QuantumField[name, μ], {a}][p], ...}] calculates the functional derivative of expr with respect to the field list (with incoming momenta p, etc.) and does the fourier transform.
FunctionalD[exp, {QuantumField[name, μ], {a}], ...}] calculates the functional derivate and does partial integration but omits the x−space delta functions.

FunctionalD is a low level function used in FeynRule.

■ See also: FeynRule, QuantumField.

■ Examples

Instead of the usual $\delta\phi(x)/\delta\phi(y) = \delta^{(D)}(x-y)$ the arguments and the δ function are omitted, i.e., for the program for simplicity $\delta\phi/\delta\phi = 1$

```
FunctionalD[QuantumField[ϕ], QuantumField[ϕ]]
```

1

```
FunctionalD[QuantumField[ϕ]^2, QuantumField[ϕ]]
```

2ϕ

Instead of the usual $(\delta\partial_\mu \phi(x))/\delta\phi(y) = \partial_\mu \delta\text{^}(D)(x-y)$ the arguments are omitted, and the ∂_μ operator is specified by default to be an integration by parts operator, i.e., the right hand side will be just $-\partial_\mu$, or, more precisely (by default) $-\vec{\partial}_\mu$.

```
FunctionalD[QuantumField[PartialD[μ], ϕ], QuantumField[ϕ]]
```

$-\vec{\partial}_\mu$

■ S@ϕD = 1 2 d^ Dx @∂μ ϕHxL∂ ^ μϕHxL − m^ 2ϕHxLϕHyLL

```
s[ϕ] = (QuantumField[PartialD[μ], ϕ].QuantumField[PartialD[μ], ϕ] -
    m^2 QuantumField[ϕ].QuantumField[ϕ]) / 2
```

$\frac{1}{2}(\partial_\mu \phi\, \partial_\mu \phi - m^2 (\phi\, \phi))$

```
FunctionalD[s[ϕ], QuantumField[ϕ]]
```

$-\phi\, m^2 - \partial_\mu \partial_\mu \phi$

■ S@AD = − d^ Dx 1 4 $F_a^{\mu\nu}$HxL$F_{\mu\nu a}$HxL

First approach

```
F1 = FieldStrength[μ, ν, a, {A, b, c}, 1, Explicit → True]
```

$\partial_\mu A_\nu^a - \partial_\nu A_\mu^a + (A_\mu^b A_\nu^c) f_{abc}$

```
F2 = FieldStrength[μ, ν, a, {A, d, e}, 1, Explicit → True]
```

$\partial_\mu A_\nu^a - \partial_\nu A_\mu^a + (A_\mu^d A_\nu^e) f_{ade}$

```
S[A] = -1 / 4 F1.F2
```

$-\frac{1}{4}((\partial_\mu A_\nu^a - \partial_\nu A_\mu^a + (A_\mu^b A_\nu^c) f_{abc})(\partial_\mu A_\nu^a - \partial_\nu A_\mu^a + (A_\mu^d A_\nu^e) f_{ade}))$

In order to derive the equation of motion the functional derivative of S with respect to A_σ^g has to be set to zero. Bearing in mind that for FeynCalc we have to be precise as to where which operators (coming from the substitution of the derivative of the δ function) act:, act with the functional derivative operator on the first field strength:

$0 = (\delta S)/(\delta A_\sigma^g(y)) = -2/4 \int d\text{^}Dx (\delta/(\delta A_\sigma^g(y)) F_{\mu\nu a}(x))\, F_a^{\mu\nu}(x)$

See what happens with just $(\delta\, S[A])/(\delta\, A_\sigma^g)$.

```
Ag = QuantumField[A, {σ}, {g}]
```

A_σ^g

```
t1 = FunctionalD[F1, Ag]
```

$-g^{\nu\sigma}\, \vec{\partial}_\mu\, \delta_{ag} + g^{\mu\sigma}\, \vec{\partial}_\nu\, \delta_{ag} + g^{\nu\sigma}\, A_\mu^b\, f_{abg} - g^{\mu\sigma}\, A_\nu^c\, f_{acg}$

In order to minimize the number of dummy indices, replace b → c.

```
t1 = t1 /. b → c
```

$$-g^{\nu\sigma}\,\vec{\partial}_\mu\,\delta_{ag}+g^{\mu\sigma}\,\vec{\partial}_\nu\,\delta_{ag}+g^{\nu\sigma}\,A^c_\mu\,f_{acg}-g^{\mu\sigma}\,A^c_\nu\,f_{acg}$$

Instead of inserting the definition for the second $F^{\mu\nu}_a$, introduce a QuantumField object with antisymmetry built into the Lorentz indices:

```
F /: QuantumField[pard___, F, β_, α_, s_] :=
 -QuantumField[pard, F, α, β, s] /; ! OrderedQ[{β, α}]

QuantumField[F, {μ, ν}, {a}]
```

$$F^a_{\mu\nu}$$

```
% /. {μ :→ ν, ν :→ μ}
```

$$-F^a_{\mu\nu}$$

```
t2 = Contract[ExpandPartialD[-1 / 2
     t1.QuantumField[F, LorentzIndex[μ], LorentzIndex[ν], SUNIndex[a]]]] /. Dot → Times
```

$$\frac{1}{2}\,\partial_\mu\,F^g_{\mu\sigma}+\frac{1}{2}\,\partial_\nu\,F^g_{\nu\sigma}-\frac{1}{2}\,A^c_\mu\,F^a_{\mu\sigma}\,f_{acg}-\frac{1}{2}\,A^c_\nu\,F^a_{\nu\sigma}\,f_{acg}$$

```
t3 = t2 /. ν → μ
```

$$\partial_\mu\,F^g_{\mu\sigma}-A^c_\mu\,F^a_{\mu\sigma}\,f_{acg}$$

```
t4 = FCE[t3] /. SUNF[a, c, g] → -SUNF[g, c, a]
```

$$\partial_\mu\,F^g_{\mu\sigma}+A^c_\mu\,F^a_{\mu\sigma}\,f_{gca}$$

Since the variational derivative vanishes t4 implies that $0 = D_\mu\,F^{\mu\sigma}_g$.

Second approach

It is of course also possible to do the functional deriviate on the S[A] with both field strength tensors inserted.

```
S[A]
```

$$-\frac{1}{4}\left((\partial_\mu\,A^a_\nu-\partial_\nu\,A^a_\mu+(A^b_\mu\,A^c_\nu)\,f_{abc})\,(\partial_\mu\,A^a_\nu-\partial_\nu\,A^a_\mu+(A^d_\mu\,A^e_\nu)\,f_{ade})\right)$$

```
r1 =
  FunctionalD[S[A], Ag]
```

$$\begin{aligned}&\frac{1}{2}\,(\partial_\mu\,\partial_\mu\,A^g_\sigma)-\frac{1}{2}\,(\partial_\mu\,\partial_\sigma\,A^g_\mu)+\frac{1}{2}\,(\partial_\nu\,\partial_\nu\,A^g_\sigma)-\frac{1}{2}\,(\partial_\nu\,\partial_\sigma\,A^g_\nu)-\frac{1}{4}\,(A^b_\mu\,\partial_\mu\,A^a_\sigma)\,f_{abg}+\frac{1}{4}\,(A^b_\mu\,\partial_\sigma\,A^a_\mu)\,f_{abg}-\\&\frac{1}{4}\,(A^c_\nu\,\partial_\nu\,A^a_\sigma)\,f_{acg}+\frac{1}{4}\,(A^c_\nu\,\partial_\sigma\,A^a_\nu)\,f_{acg}-\frac{1}{4}\,(A^b_\mu\,A^d_\mu\,A^e_\sigma)\,f_{abg}\,f_{ade}+\frac{1}{4}\,(A^c_\nu\,A^d_\sigma\,A^e_\nu)\,f_{acg}\,f_{ade}-\\&\frac{1}{4}\,(\partial_\mu\,A^a_\sigma\,A^d_\mu)\,f_{adg}+\frac{1}{4}\,(\partial_\sigma\,A^a_\mu\,A^d_\mu)\,f_{adg}-\frac{1}{4}\,(A^b_\mu\,A^c_\sigma\,A^d_\mu)\,f_{abc}\,f_{adg}-\frac{1}{4}\,(\partial_\nu\,A^a_\sigma\,A^e_\nu)\,f_{aeg}+\\&\frac{1}{4}\,(\partial_\sigma\,A^a_\nu\,A^e_\nu)\,f_{aeg}+\frac{1}{4}\,(A^b_\sigma\,A^c_\nu\,A^e_\nu)\,f_{abc}\,f_{aeg}+\frac{1}{4}\,(A^b_\mu\,\partial_\mu\,A^c_\sigma)\,f_{bcg}-\frac{1}{4}\,(A^b_\sigma\,\partial_\nu\,A^c_\nu)\,f_{bcg}+\frac{1}{4}\,(\partial_\mu\,A^b_\mu\,A^c_\sigma)\,f_{bcg}-\\&\frac{1}{4}\,(\partial_\nu\,A^b_\sigma\,A^c_\nu)\,f_{bcg}+\frac{1}{4}\,(A^d_\mu\,\partial_\mu\,A^e_\sigma)\,f_{deg}-\frac{1}{4}\,(A^d_\sigma\,\partial_\nu\,A^e_\nu)\,f_{deg}+\frac{1}{4}\,(\partial_\mu\,A^d_\mu\,A^e_\sigma)\,f_{deg}-\frac{1}{4}\,(\partial_\nu\,A^d_\sigma\,A^e_\nu)\,f_{deg}\end{aligned}$$

This is just funcional derivatves and partial integration and simple contraction of indices. No attempt is made to rename dummy indices (since this is difficult in general...).

With a general replacement rule only valid for commuting fields the color indices can be canonicalized a bit more. The idea is to use the commutative properties of the vector fields, and canonicalize the color indices by a trick. This function will work on any commuting product of fields.

```
Clear[symfun];

symfun[z_, fieldname_Symbol] :=
 Expand[SUNSimplify[FixedPoint[Collect2[DotSimplify[#1 /. Times → Dot] /.
        (((qi___).QuantumField[par1___, fieldname, li1_, sui1_].
             QuantumField[par2___, fieldname, li2_, sui2_].qf___) any_) :→
         ((qi.QuantumField[par1, A, li1, sui2].QuantumField[par2, fieldname, li2, sui1].
              qf) (any /. {sui1 :→ sui2, sui2 :→ sui1})) /; (! (FreeQ2[any, {sui1, su2}])) &&
           ! (OrderedQ[{sui1, sui2}]) /. Dot → Times, QuantumField] &, z, 42]]]

r2 = symfun[r1, A]
```

$$\frac{1}{2}(\partial_\mu \partial_\mu A^g_\sigma) - \frac{1}{2}(\partial_\mu \partial_\sigma A^g_\mu) + \frac{1}{2}(\partial_\nu \partial_\nu A^g_\sigma) - \frac{1}{2}(\partial_\nu \partial_\sigma A^g_\nu) - \frac{1}{2} A^a_\sigma \partial_\mu A^b_\mu f_{abg} + A^a_\mu \partial_\mu A^b_\sigma f_{abg} - \frac{1}{2} A^a_\sigma \partial_\nu A^b_\nu f_{abg} +$$
$$A^a_\nu \partial_\nu A^b_\sigma f_{abg} - \frac{1}{2} A^a_\mu \partial_\sigma A^b_\mu f_{abg} - \frac{1}{2} A^a_\nu \partial_\sigma A^b_\nu f_{abg} - \frac{1}{4} A^b_\mu A^c_\mu A^e_\sigma f_{abg} f_{ace} - \frac{1}{2} A^b_\nu A^c_\nu A^e_\sigma f_{abg} f_{ace} - \frac{1}{4} A^b_\mu A^c_\mu A^e_\sigma f_{abe} f_{acg}$$

```
r3 = r2 /. ν → μ
```

$$\partial_\mu \partial_\mu A^g_\sigma - \partial_\mu \partial_\sigma A^g_\mu - A^a_\sigma \partial_\mu A^b_\mu f_{abg} + 2 A^a_\mu \partial_\mu A^b_\sigma f_{abg} - A^a_\mu \partial_\sigma A^b_\mu f_{abg} - \frac{3}{4} A^b_\mu A^c_\mu A^e_\sigma f_{abg} f_{ace} - \frac{1}{4} A^b_\mu A^c_\mu A^e_\sigma f_{abe} f_{acg}$$

Inspection reveals that still terms are the same. Gather the terms with two f's:

```
twof = Select[r3, Count[#, SUNF[__]] === 2 &]
```

$$-\frac{3}{4} A^b_\mu A^c_\mu A^e_\sigma f_{abg} f_{ace} - \frac{1}{4} A^b_\mu A^c_\mu A^e_\sigma f_{abe} f_{acg}$$

```
twofnew = (twof[[1]] + (twof[[2]] /. {b :→ c, c :→ b})) /. {a :→ c, c :→ a}
```

$$-A^a_\mu A^b_\mu A^e_\sigma f_{ace} f_{bcg}$$

```
r4 = r3 - twof + twofnew
```

$$\partial_\mu \partial_\mu A^g_\sigma - \partial_\mu \partial_\sigma A^g_\mu - A^a_\sigma \partial_\mu A^b_\mu f_{abg} + 2 A^a_\mu \partial_\mu A^b_\sigma f_{abg} - A^a_\mu \partial_\sigma A^b_\mu f_{abg} - A^a_\mu A^b_\mu A^e_\sigma f_{ace} f_{bcg}$$

Check that this is now indeed the same as the t4 result from the first attempt.

```
t4
```

$$\partial_\mu F^g_{\mu\sigma} + A^c_\mu F^a_{\mu\sigma} f_{gca}$$

```
w0 = RightPartialD[μ].FieldStrength[μ, σ, g, {A, a, b}, 1] +
  QuantumField[A, LorentzIndex[μ], SUNIndex[c]]
   FieldStrength[μ, σ, a, {A, b, d}, 1] SUNF[g, c, a]
```

$$\vec{\partial}_\mu F^{g\{A,a,b\}1}_{\mu\sigma} + F^{a\{A,b,d\}1}_{\mu\sigma} A^c_\mu f_{gca}$$

```
w1 = Explicit[w0]
```

$$\vec{\partial}_\mu \left(\partial_\mu A^g_\sigma - \partial_\sigma A^g_\mu + (A^a_\mu A^b_\sigma) f_{gab}\right) + A^c_\mu \left(\partial_\mu A^a_\sigma - \partial_\sigma A^a_\mu + (A^b_\mu A^d_\sigma) f_{abd}\right) f_{gca}$$

```
w2 = ExpandPartialD[w1] /. Dot → Times
```

$$\partial_\mu \partial_\mu A^g_\sigma - \partial_\mu \partial_\sigma A^g_\mu + A^b_\sigma \partial_\mu A^a_\mu f_{abg} + A^a_\mu \partial_\mu A^b_\sigma f_{abg} - A^c_\mu \partial_\mu A^a_\sigma f_{acg} + A^c_\mu \partial_\sigma A^a_\mu f_{acg} - A^b_\mu A^c_\mu A^d_\sigma f_{abd} f_{acg}$$

```
dif1 = w2 - r4
```

$$A^b_\sigma \partial_\mu A^a_\mu f_{abg} + A^a_\sigma \partial_\mu A^b_\mu f_{abg} - A^a_\mu \partial_\mu A^b_\sigma f_{abg} + A^a_\mu \partial_\sigma A^b_\mu f_{abg} - A^c_\mu \partial_\mu A^a_\sigma f_{acg} + A^c_\mu \partial_\sigma A^a_\mu f_{acg} - A^b_\mu A^c_\mu A^d_\sigma f_{abd} f_{acg} + A^a_\mu A^b_\mu A^e_\sigma f_{ace} f_{bcg}$$

```
dif2 = symfun[dif1, A]
```

0

quoderatdemonstrandum

```
Unset[s[ϕ]]; Unset[S[A]];

Clear[Ag, F1, F2, t1, t2, t3, t4, F, r1, r2,
 r3, symfun, twof, twofnew, w0, w1, w2, dif1, dif2, dif3]
```

■ Examples of funcional differentiations as used in FeynRule

This is a part of the QCD Lagrangian.

```
(Gstrong * QuantumField[GaugeField, LorentzIndex[li1],     SUNIndex[si2]][] .
    QuantumField[GaugeField, LorentzIndex[li2],     SUNIndex[si4]][] .
    QuantumField[PartialD[LorentzIndex[li1]], GaugeField,     LorentzIndex[li2],
      SUNIndex[si1]][] *   SUNF[SUNIndex[si1], SUNIndex[si2], SUNIndex[si4]]) / 4
```

$$\frac{1}{4} g_s \left(A^{si2}_{li1}() A^{si4}_{li2}() \left(\partial_{li1} A^{si1}_{li2}\right)()\right) f_{si1si2si4}$$

```
FunctionalD[%, {QuantumField[GaugeField, {μ}, {a}][p],
  QuantumField[GaugeField, {ν}, {b}][q],   QuantumField[GaugeField, {ρ}, {c}][r]}]
```

$$\frac{1}{4} g_s \left(\left(g^{li1\,\mu} \delta_{asi2}\right)\left(g^{li2\,\nu} \delta_{bsi4}\right)\left(-i\, r^{li1}\, g^{li2\,\rho} \delta_{csi1}\right) + \left(g^{li1\,\mu} \delta_{asi2}\right)\left(g^{li2\,\rho} \delta_{csi4}\right)\left(-i\, q^{li1}\, g^{li2\,\nu} \delta_{bsi1}\right) + \left(g^{li1\,\nu} \delta_{bsi2}\right)\left(g^{li2\,\mu} \delta_{asi4}\right)\left(-i\, r^{li1}\, g^{li2\,\rho} \delta_{csi1}\right) + \left(g^{li1\,\nu} \delta_{bsi2}\right)\left(g^{li2\,\rho} \delta_{csi4}\right)\left(-i\, p^{li1}\, g^{li2\,\mu} \delta_{asi1}\right) + \left(g^{li1\,\rho} \delta_{csi2}\right)\left(g^{li2\,\mu} \delta_{asi4}\right)\left(-i\, q^{li1}\, g^{li2\,\nu} \delta_{bsi1}\right) + \left(g^{li1\,\rho} \delta_{csi2}\right)\left(g^{li2\,\nu} \delta_{bsi4}\right)\left(-i\, p^{li1}\, g^{li2\,\mu} \delta_{asi1}\right)\right) f_{si1si2si4}$$

```
Calc[% // Calc] // Factor
```

$$\frac{1}{4} i\, g_s \left(q^\mu g^{\nu\rho} - r^\mu g^{\nu\rho} - g^{\mu\rho} p^\nu + g^{\mu\rho} r^\nu + g^{\mu\nu} p^\rho - g^{\mu\nu} q^\rho\right) f_{abc}$$

■ FVD

■ Description

FVD[p, μ] is the D–dimensional vector p with Lorentz index μ.

■ See also: FCE, FCI, FV, FourVector, Pair.

■ Examples

```
FVD[p, μ]
```

p^{μ}

```
FVD[p - q, μ]
```

$(p-q)^{\mu}$

```
FVD[p, μ] // InputForm
```

```
FVD[p, μ]
```

```
FCI[FVD[p, μ]] // InputForm
```

```
Pair[LorentzIndex[μ, D], Momentum[p, D]]
```

There is no special function to expand momenta in FVD.

```
ExpandScalarProduct[FVD[p - q, μ]]
```

$p^{\mu} - q^{\mu}$

```
InputForm[%]
```

```
Pair[LorentzIndex[μ, D], Momentum[p, D]] -
  Pair[LorentzIndex[μ, D], Momentum[q, D]]
```

■ FV

■ Description

FV[p, μ] is the four–dimensional vector p^μ.

■ See also: FCE,FCI, FVD, FourVector, Pair.

■ Examples

```
FV[p, μ]
```

p^{μ}

```
FV[p - q, μ]
```

$(p-q)^{\mu}$

```
FV[p, μ] // InputForm
```

```
FV[p, μ]
```

```
FCI[FV[p, μ]] // InputForm
```

```
Pair[LorentzIndex[μ], Momentum[p]]
```

There is no special function to expand momenta in FV; ExpandScalarProduct does the job.

```
ExpandScalarProduct[FV[p - q, μ]]
```

$p^{\mu} - q^{\mu}$

```
InputForm[%]
```

```
Pair[LorentzIndex[μ], Momentum[p]] -
  Pair[LorentzIndex[μ], Momentum[q]]
```

■ GAD

■ Description

GAD[μ] can be used as input for a D–dimensional γ_{μ} and is transformed into DiracGamma[LorentzIndex[μ,D],D] by FeynCalcInternal (=FCI). GAD[μ,ν,...] is a short form for GAD[μ].GAD[ν]. ... .

■ See also: DiracMatrix, GA, GS.

■ Examples

```
GAD[μ]
```

γ^{μ}

```
GAD[μ, ν] - GAD[ν, μ]
```

$\gamma^{\mu}\,\gamma^{\nu} - \gamma^{\nu}\,\gamma^{\mu}$

```
InputForm[FCI[GAD[μ]]]
```

```
DiracGamma[LorentzIndex[μ, D], D]
```

```
GAD[μ, ν, ρ, σ]
```

$\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{\rho}\,\gamma^{\sigma}$

```
InputForm[GAD[μ, ν, ρ, σ]]
```

```
GAD[μ] . GAD[ν] . GAD[ρ] . GAD[σ]
```

```
GAD[α].(GSD[p] + m).GAD[β]
```

$\gamma^{\alpha}\,(m+\gamma\cdot p)\,\gamma^{\beta}$

■ GammaEpsilon

■ Description

GammaEpsilon[ex] gives a series expansion of Gamma[ex] in Epsilon up to order 6 (where EulerGamma is neglected).

■ See also: GammaExpand, Series2.

■ Examples

If the argument is of the form (1+a Epsilon) the result is not calculated but tabulated.

```
GammaEpsilon[1 + a Epsilon]
```

$$\text{C\$1145}\,\varepsilon^6+\left(-\frac{1}{5}\,\zeta(5)\,a^5-\frac{1}{36}\,\pi^2\,\zeta(3)\,a^5\right)\varepsilon^5+\frac{1}{160}\,a^4\,\pi^4\,\varepsilon^4-\frac{1}{3}\,a^3\,\zeta(3)\,\varepsilon^3+\frac{1}{12}\,a^2\,\pi^2\,\varepsilon^2+1$$

```
GammaEpsilon[1 - Epsilon / 2]
```

$$\text{C\$1146}\,\varepsilon^6+\left(\frac{\pi^2\,\zeta(3)}{1152}+\frac{\zeta(5)}{160}\right)\varepsilon^5+\frac{\pi^4\,\varepsilon^4}{2560}+\frac{1}{24}\,\zeta(3)\,\varepsilon^3+\frac{\pi^2\,\varepsilon^2}{48}+1$$

For other arguments the expansion is calculated.

```
GammaEpsilon[Epsilon]
```

$$\text{C\$1150}\,\varepsilon^6+\left(\frac{1}{36}\,\pi^2\,\psi^{(2)}(1)-\frac{\frac{1}{36}\,\pi^3\,\psi^{(2)}(1)+\frac{1}{120}\,\pi\left(-\frac{5}{3}\,\pi^2\,\psi^{(2)}(1)+24\,\zeta(5)\right)}{\pi}\right)\varepsilon^4+\frac{\pi^4\,\varepsilon^3}{160}+\frac{1}{6}\,\psi^{(2)}(1)\,\varepsilon^2+\frac{\pi^2\,\varepsilon}{12}+\frac{1}{\varepsilon}$$

```
GammaEpsilon[x]
```

$\Gamma(x)$

■ GammaExpand

■ Description

GammaExpand[expr] rewrites Gamma[n + m] in expr (where n has Head Integer).

■ See also: GammaEpsilon.

■ Examples

```
GammaExpand[Gamma[2 + Epsilon]]
```

$(\varepsilon + 1)\,\Gamma(\varepsilon + 1)$

```
GammaExpand[Gamma[-3 + Epsilon]]
```

$$\frac{\Gamma(\varepsilon + 1)}{(\varepsilon - 3)\,(\varepsilon - 2)\,(\varepsilon - 1)\,\varepsilon}$$

```
GammaExpand[Gamma[1 + Epsilon]]
```

$\Gamma(\varepsilon + 1)$

■ GA

■ Description

GA[μ] can be used as input for a 4–dimensional γ_μ and is transformed into DiracGamma[LorentzIndex[μ]] by FeynCalcInternal (=FCI). GA[μ,ν, ...] is a short form for GA[μ].GA[ν]. ... .

■ See also: DiracMatrix, GAD, GS.

■ Examples

```
GA[μ]
```

γ^μ

```
GA[μ, ν] - GA[ν, μ]
```

$\gamma^\mu\,\gamma^\nu - \gamma^\nu\,\gamma^\mu$

```
InputForm[FCI[GA[μ]]]
```

```
DiracGamma[LorentzIndex[μ]]
```

```
GA[μ, ν, ρ, σ]
```

$$\gamma^{\mu}\,\gamma^{\nu}\,\gamma^{\rho}\,\gamma^{\sigma}$$

```
InputForm[GA[μ, ν, ρ, σ]]
```

```
GA[μ] . GA[ν] . GA[ρ] . GA[σ]
```

```
GA[α].(GS[p] + m).GA[β]
```

$$\gamma^{\alpha}\,(m+\gamma\cdot p)\,\gamma^{\beta}$$

■ GhostPropagator

■ Description

GhostPropagator[p, a, b] gives the ghost propagator where "a" and "b" are the color indices. GhostPropagator[p] omits the δ_{ab}.

■ See also: GHP, GluonPropagator, GluonGhostVertex.

■ Examples

```
GhostPropagator[p, a, b]
```

$$\frac{i\,\delta_{ab}}{p^2}$$

```
GhostPropagator[p]
```

$$\frac{i}{p^2}$$

```
GhostPropagator[p, 1, 2]
```

$$\frac{i\,\delta_{\text{ci1 ci2}}}{p^2}$$

```
GhostPropagator[-k, 3, 4] // FCE // InputForm
```

```
I*FAD[-k]*SD[ci3, ci4]
```

■ GHP

■ Description

GHP[p,a, b] gives theghostpropagatorwhere"a" and"b" arethe colorindices.GHP[p] omitsthe δ_{ab}.

■ Seealso: GhostPropagator, GluonPropagator, GluonGhostVertex.

■ Examples

```
GHP[p, a, b]
```

$$\frac{i\,\delta_{ab}}{p^2}$$

```
GHP[p]
```

$$\frac{i}{p^2}$$

```
GHP[p, 1, 2]
```

$$\frac{i\,\delta_{\text{ci1 ci2}}}{p^2}$$

```
InputForm[FCE[GHP[-k, 3, 4]]]
```

```
I*FAD[-k]*SD[ci3, ci4]
```

■ GaugeField

■ Description

GaugeFields a justa name.No functionalpropertiesare associatedwithit.
GaugeFields usedas defaultsettingfor theoptionQuantumFieldof FieldStrength.

■ Seealso: FieldStrength, QuantumField.

■ Examples

```
GaugeField
```

$$A$$

```
QuantumField[GaugeField, LorentzIndex[μ], SUNIndex[a]]
```

A_μ^a

■ GluonGhostVertex

■ Description

GluonGhostVertex[{p,mu,a},{q,nu,b},{k,rho,c}] or
GluonGhostVertex[p,mu,a, q,nu,b, k,rho,c] yields the Gluon–Ghost–vertex.
The first argument represents the gluon and the third argument the outgoing ghost field (but incoming four–momentum).
The dimension and the name of the coupling constant are determined by the options Dimension and CouplingConstant.

■ See also: GluonVertex.

```
Options[GluonGhostVertex]
```

{Dimension → D, CouplingConstant → g_s}

■ Examples

```
GluonGhostVertex[{p, μ, a}, {q, ν, b}, {k, ρ, c}]
```

$g_s\, k^\mu\, f_{abc}$

```
% // InputForm
```

```
Gstrong*Pair[LorentzIndex[μ, D], Momentum[k, D]]*
  SUNF[SUNIndex[a], SUNIndex[b], SUNIndex[c]]
```

```
GluonGhostVertex[p, 1, q, 2, k, 3]
```

$g_s\, k^{\text{li1}}\, f_{\text{ci1ci2ci3}}$

■ GluonPropagator

■ Description

GluonPropagator[{p,mu,a},{nu,b}] or
GluonPropagator[p,mu,a, nu, b] yields the gluon propagator.
GluonPropagator[{p,mu},{nu}] or GluonPropagator[p,mu,nu] omits the SUNDelta. The gauge and the dimension is determined by the option Gauge and Dimension.
The following settings of Gauge are possible:
1 : the Feynman gauge
α : the general covariant gauge
{Momentum[n],1} : axial gauge.

GP can be used as an abbreviation of GluonPropagator.

```
GP
```

GluonPropagator

```
Options[GluonPropagator]
```

{CounterTerm → False, CouplingConstant → g_s, Dimension → D, Gauge → 1, Ϙ → False}

■ See also: GluonVertex, GhostPropagator, GluonGhostVertex.

■ Examples

```
GluonPropagator[p, μ, a, ν, b]
```

$$-\frac{i\, g^{\mu\nu}\, \delta_{ab}}{p^2}$$

```
GluonPropagator[p, μ, a, ν, b, Gauge → α]
```

$$\frac{i\left(\frac{(1-\alpha)\, p^{\mu}\, p^{\nu}}{p^2} - g^{\mu\nu}\right)\delta_{ab}}{p^2}$$

```
GluonPropagator[p, μ, a, ν, b, Gauge → {Momentum[n], 1}]
```

$$\frac{i\left(-g^{\mu\nu} - p^{\mu}\, p^{\nu}\, n^2 + \frac{n^{\mu}\, n^{\nu}\, p^2}{(n\cdot p)^2} + \frac{n^{\nu}\, p^{\mu} + n^{\mu}\, p^{\nu}}{n\cdot p}\right)\delta_{ab}}{p^2}$$

```
GluonPropagator[p, μ, ν]
```

$$-\frac{i\, g^{\mu\nu}}{p^2}$$

This is a convenient way to enter amplitudes by hand (GP is an abbreviation of GluonPropagator).

```
GP[p, 1, 2]
```

$$-\frac{i\, g^{\text{li1 li2}}\, \delta_{\text{ci1 ci2}}}{p^2}$$

```
GP[-k, 3, 4] // FCE // InputForm
```

```
-I*FAD[-k]*MTD[li3, li4]*SD[ci3, ci4]
```

```
GluonPropagator[p, μ, a, ν, b, CounterTerm → 1]
```

$$-\frac{i\, C_A\, g_s^2\, S_n\left(\frac{11\, p^{\mu}\, p^{\nu}}{3} - \frac{19}{6}\, g^{\mu\nu}\, p^2\right)\delta_{ab}}{\varepsilon}$$

```
GluonPropagator[p, μ, a, ν, b, CounterTerm → 2]
```

$$-\frac{i\, C_A\, g_s^2\, S_n \left(-\frac{1}{3}\, p^\mu\, p^\nu - \frac{1}{6}\, g^{\mu\nu}\, p^2\right) \delta_{ab}}{\varepsilon}$$

```
GluonPropagator[p, μ, a, ν, b, CounterTerm → 3]
```

$$-\frac{2\, i\, g_s^2\, S_n\, T_f \left(\frac{4}{3}\, g^{\mu\nu}\, p^2 - \frac{4\, p^\mu\, p^\nu}{3}\right) \delta_{ab}}{\varepsilon}$$

```
GluonPropagator[p, μ, a, ν, b, CounterTerm → 4]
```

$$-\frac{i\, C_A\, g_s^2\, S_n \left(\frac{10\, p^\mu\, p^\nu}{3} - \frac{10}{3}\, g^{\mu\nu}\, p^2\right) \delta_{ab}}{\varepsilon}$$

```
GluonPropagator[p, μ, a, ν, b, CounterTerm → 5]
```

$$\frac{i\, C_A\, S_n \left(\frac{10\, p^\mu\, p^\nu}{3} - \frac{10}{3}\, g^{\mu\nu}\, p^2\right) \delta_{ab}\, g_s^2}{\varepsilon} + \frac{i\, S_n\, T_f \left(\frac{4}{3}\, g^{\mu\nu}\, p^2 - \frac{4\, p^\mu\, p^\nu}{3}\right) \delta_{ab}\, g_s^2}{\varepsilon}$$

■ GluonVertex

■ Description

Note:All momentaareflowingintothevertex.

GluonVertex[{p,mu,a},{q,nu,b},{k,la,c}] orGluonVertex[p,mu,a,q,nu,b, k,la,c] yieldsthe3–gluonvertex.
GluonVertex[{p,mu},{q,nu}, {k,la}] yieldsthe3–gluonvertexwithoutcolorstructureandthe couplingconstant.
GluonVertex[{p,mu,a},{q,nu,b},{k,la,c}, {s,si,d}] or
GluonVertex[{mu,a},{nu,b}, {la,c}, {si,d}] or
GluonVertex[p,mu,a,q,nu,b, k,la,c, s,si,d]or
GluonVertex[mu,a, nu,b, la,c, si,d] yieldsthe 4–gluonvertex.
Thedimensionandthenameof thecouplingconstantaredeterminedby theoptionsDimensionandCouplingConstant.

GV canbeusedas anabbreviationof GluonVertex.

```
GV
```

GluonVertex

```
Options[GluonVertex]
```

{CouplingConstant→ g_s, Dimension→ D, Explicit→ False, Ϙ → False}

■ **See also: GluonGhostVertex, GluonPropagator.**

■ Examples

```
GluonVertex[{p, μ, a}, {q, ν, b}, {r, ρ, c}]
```

$g_s\, V^{\mu\nu\rho}(p,\ q,\ r)\, f_{abc}$

```
GluonVertex[{p, μ, a}, {q, ν, b}, {r, ρ, c}] // Explicit
```

$g_s\, (q^\mu\, g^{\nu\rho} - r^\mu\, g^{\nu\rho} - g^{\mu\rho}\, p^\nu + g^{\mu\rho}\, r^\nu + g^{\mu\nu}\, p^\rho - g^{\mu\nu}\, q^\rho)\, f_{abc}$

```
GluonVertex[{p, μ}, {q, ν}, {r, ρ}]
```

$V^{\mu\nu\rho}(p,\ q,\ r)$

```
GluonVertex[{p, μ}, {q, ν}, {r, ρ}] // Explicit
```

$q^\mu\, g^{\nu\rho} - r^\mu\, g^{\nu\rho} - g^{\mu\rho}\, p^\nu + g^{\mu\rho}\, r^\nu + g^{\mu\nu}\, p^\rho - g^{\mu\nu}\, q^\rho$

```
GluonVertex[{p, μ, a}, {q, ν, b}, {r, ρ, c}, {s, σ, d}]
```

$-i\, g_s^2\, ((g^{\mu\nu}\, g^{\rho\sigma} - g^{\mu\rho}\, g^{\nu\sigma})\, f_{adu9}\, f_{bcu9} + (g^{\mu\nu}\, g^{\rho\sigma} - g^{\mu\sigma}\, g^{\nu\rho})\, f_{acu9}\, f_{bdu9} + (g^{\mu\rho}\, g^{\nu\sigma} - g^{\mu\sigma}\, g^{\nu\rho})\, f_{abu9}\, f_{cdu9})$

A very convenient way to enter diagrams by hand is to label each line hitting a vertex by a number and put this number after the inflowing momentum.

```
GV[p, 1, q, 2, r, 3]
```

$g_s\, V^{li1\,li2\,li3}(p,\ q,\ r)\, f_{ci1ci2ci3}$

```
GV[p, 1, q, 2, r, 3, s, 4]
```

$-i\, g_s^2\, ((g^{li1\,li2}\, g^{li3\,li4} - g^{li1\,li3}\, g^{li2\,li4})\, f_{ci1ci4u10}\, f_{ci2ci3u10} +$
$(g^{li1\,li2}\, g^{li3\,li4} - g^{li1\,li4}\, g^{li2\,li3})\, f_{ci1ci3u10}\, f_{ci2ci4u10} + (g^{li1\,li3}\, g^{li2\,li4} - g^{li1\,li4}\, g^{li2\,li3})\, f_{ci1ci2u10}\, f_{ci3ci4u10})$

■ GSD

■ Description

GSD[p] can be used as input for a D-dimensional $\not{p}$ (= $\gamma.p$ = $\gamma_\mu\ p$^μ) and is transformed into DiracGamma[Momentum[p,D],D] by FeynCalcInternal (=FCI).
GSD[p,q,...] is a short form for GSD[p].GSD[q].... .

■ **See also: DiracGamma, DiracSlash, GA, GAD.**

■ Examples

```
GSD[p]
```

$\gamma \cdot p$

```
GSD[p] // FCI // InputForm
```

```
DiracGamma[Momentum[p, D], D]
```

```
GSD[p, q, r, s]
```

$\gamma \cdot p \gamma \cdot q \gamma \cdot r \gamma \cdot s$

```
GSD[p, q, r, s] // InputForm
```

```
GSD[p] . GSD[q] . GSD[r] . GSD[s]
```

```
GSD[q].(GSD[p] + m).GSD[q]
```

$\gamma \cdot q (m + \gamma \cdot p) \gamma \cdot q$

■ GS

■ Description

GS[p] can be used as input for a 4–dimensional $\not{p}$ (= $\gamma.p$ = γ_μ p^μ) and is transformed into DiracGamma[Momentum[p
by FeynCalcInternal (=FCI).
GS[p,q,...] is a short form for GS[p].GS[q].... .

■ **See also: DiracGamma, DiracSlash, GA, GAD.**

■ Examples

```
GS[p]
```

$\gamma \cdot p$

```
GS[p] // FCI // InputForm
```

```
DiracGamma[Momentum[p]]
```

```
GS[p, q, r, s]
```

$\gamma \cdot p \gamma \cdot q \gamma \cdot r \gamma \cdot s$

```
GS[p, q, r, s] // InputForm
```

```
GS[p] . GS[q] . GS[r] . GS[s]
```

```
GS[q].(GS[p] + m).GS[q]
```

$\gamma \cdot q (m + \gamma \cdot p) \gamma \cdot q$

■ Hill

■ Description

Hill[x,y] gives the Hill identity with arguments x and y. The returned object is 0.

■ See also: SimplifyPolyLog.

■ Examples

```
Hill[a, b]
```

$$\log(a)\,(\log(1-a)-\log(1-b))+\log\left(\frac{1-a}{1-b}\right)\left(-\log(a)+\log(1-b)-\log\left(\frac{a-b}{a}\right)+\log\left(\frac{a-b}{1-b}\right)\right)-$$
$$\left(\log(1-b)-\log\left(\frac{a-b}{a}\right)+\log\left(\frac{a-b}{a(1-b)}\right)\right)\log\left(\frac{(1-a)\,b}{a(1-b)}\right)+\text{Li}_2(a)+\text{Li}_2\left(\frac{1-a}{1-b}\right)-\text{Li}_2(b)+\text{Li}_2\left(\frac{b}{a}\right)-\text{Li}_2\left(\frac{(1-a)\,b}{a(1-b)}\right)-\frac{\pi^2}{6}$$

```
% /. a :> .123 /. b :> .656 // Chop
```

0

```
Hill[x, x y] // PowerExpand // SimplifyPolyLog // Expand
```

$$\zeta(2)-\log(1-y)\log(y)-\log(x)\log(1-xy)-\text{Li}_2(1-x)-\text{Li}_2(1-y)-\text{Li}_2(xy)+\text{Li}_2\left(\frac{1-x}{1-xy}\right)-\text{Li}_2\left(\frac{(1-x)\,y}{1-xy}\right)$$

```
% /. x :> .34 /. y -> .6 // N // Chop
```

0

■ HypergeometricAC

■ Description

HypergeometricAC[n][exp] analytically continues Hypergeometric2F1 functions in exp. The second argument n refers to the equation number (n) in chapter
2.10 of "Higher Transcendental Functions" by Ergelyi, Magnus,
Oberhettinger, Tricomi. In case of eq. (6) (p.109) the last line is returned for HypergeometricAC[6][exp], while the first equality is given by
HypergeometricAC[61][exp] ((2.10.1) is identical to eq. (9.5.7) of "Special Functions & their Applications" by N.N. Lebedev).

```
Options[HypergeometricAC]
```

{Collect2→ True}

■ See also: HypExplicit, HypergeometricIR, HypergeometricSE, ToHypergeometric.

■ Examples

These are all transformation rules currently built in.

```
HypergeometricAC[1][Hypergeometric2F1[α, β, γ, z]]
```

$$\frac{\Gamma(\alpha+\beta-\gamma)\,\Gamma(\gamma)\;{}_2F_1(\gamma-\alpha,\gamma-\beta;-\alpha-\beta+\gamma+1;1-z)\,(1-z)^{-\alpha-\beta+\gamma}}{\Gamma(\alpha)\,\Gamma(\beta)} + \frac{\Gamma(\gamma)\,\Gamma(-\alpha-\beta+\gamma)\;{}_2F_1(\alpha,\beta;\alpha+\beta-\gamma+1;1-z)}{\Gamma(\gamma-\alpha)\,\Gamma(\gamma-\beta)}$$

```
HypergeometricAC[2][Hypergeometric2F1[α, β, γ, z]]
```

$$\frac{\Gamma(\beta-\alpha)\,\Gamma(\gamma)\;{}_2F_1\left(\alpha,\alpha-\gamma+1;\alpha-\beta+1;\frac{1}{z}\right)(-z)^{-\alpha}}{\Gamma(\beta)\,\Gamma(\gamma-\alpha)} + \frac{\Gamma(\alpha-\beta)\,\Gamma(\gamma)\;{}_2F_1\left(\beta,\beta-\gamma+1;-\alpha+\beta+1;\frac{1}{z}\right)(-z)^{-\beta}}{\Gamma(\alpha)\,\Gamma(\gamma-\beta)}$$

```
HypergeometricAC[3][Hypergeometric2F1[α, β, γ, z]]
```

$$\frac{\Gamma(\beta-\alpha)\,\Gamma(\gamma)\;{}_2F_1\left(\alpha,\gamma-\beta;\alpha-\beta+1;\frac{1}{1-z}\right)(1-z)^{-\alpha}}{\Gamma(\beta)\,\Gamma(\gamma-\alpha)} + \frac{\Gamma(\alpha-\beta)\,\Gamma(\gamma)\;{}_2F_1\left(\beta,\gamma-\alpha;-\alpha+\beta+1;\frac{1}{1-z}\right)(1-z)^{-\beta}}{\Gamma(\alpha)\,\Gamma(\gamma-\beta)}$$

```
HypergeometricAC[4][Hypergeometric2F1[α, β, γ, z]]
```

$$\frac{\Gamma(\gamma)\,\Gamma(-\alpha-\beta+\gamma)\;{}_2F_1\left(\alpha,\alpha-\gamma+1;\alpha+\beta-\gamma+1;-\frac{1-z}{z}\right)z^{-\alpha}}{\Gamma(\gamma-\alpha)\,\Gamma(\gamma-\beta)} + \frac{(1-z)^{-\alpha-\beta+\gamma}\,\Gamma(\alpha+\beta-\gamma)\,\Gamma(\gamma)\;{}_2F_1\left(\gamma-\alpha,1-\alpha;-\alpha-\beta+\gamma+1;-\frac{1-z}{z}\right)z^{\alpha-\gamma}}{\Gamma(\alpha)\,\Gamma(\beta)}$$

```
HypergeometricAC[6][Hypergeometric2F1[α, β, γ, z]]
```

$$(1-z)^{-\beta}\ {}_2F_1\left(\beta, \gamma-\alpha; \gamma; -\frac{z}{1-z}\right)$$

```
HypergeometricAC[61][Hypergeometric2F1[α, β, γ, z]]
```

$$(1-z)^{-\alpha}\ {}_2F_1\left(\alpha, \gamma-\beta; \gamma; -\frac{z}{1-z}\right)$$

■ HypergeometricIR

■ Description

HypergeometricIR[exp], substitutes for all Hypergeometric2F1[a,b,c,z] exp by its Euler integral reprentation. The factor Integrated x[t, 0, 1] can be omitted by setting the option Integrated x -> False.

■ See also: HypergeometricAC, HypergeometricSE, ToHypergeometric.

```
Options[HypergeometricIR]
```

{Integrated x → False}

■ Examples

```
HypergeometricIR[Hypergeometric2F1[a, b, c, z], t]
```

$$\frac{(1-t)^{-b+c-1}\ t^{b-1}\ (1-t z)^{-a}\ \Gamma(c)}{\Gamma(b)\ \Gamma(c-b)}$$

```
ToHypergeometric[t^b (1 - t)^c (1 + t z)^a, t]
```

$$\frac{\Gamma(b+1)\ \Gamma(c+1)\ {}_2F_1\ (-a, b+1; b+c+2; -z)}{\Gamma(b+c+2)}$$

```
HypergeometricIR[%, t]
```

$$(1-t)^c\ t^b\ (t z+1)^a$$

■ HypergeometricSE

■ Description

HypergeometricSE[exp] expresses Hypergeometric functions by their series expansion in terms of a sum (the Sum is omitted and ν, running from 0 to ∞, is the summation index).

```
Options[HypergeometricSE]
```

{Simplify→ FullSimplify}

■ See also: HypergeometricIR.

■ Examples

```
HypergeometricSE[Hypergeometric2F1[a, b, c, z], ν]
```

$$\frac{z^{\nu}\,\Gamma(c)\,\Gamma(a+\nu)\,\Gamma(b+\nu)}{\Gamma(a)\,\Gamma(b)\,\Gamma(\nu+1)\,\Gamma(c+\nu)}$$

```
HypergeometricSE[HypergeometricPFQ[{a, b, c}, {d, e}, z], ν]
```

$$\frac{z^{\nu}\,\Gamma(d)\,\Gamma(e)\,\Gamma(a+\nu)\,\Gamma(b+\nu)\,\Gamma(c+\nu)}{\Gamma(a)\,\Gamma(b)\,\Gamma(c)\,\Gamma(\nu+1)\,\Gamma(d+\nu)\,\Gamma(e+\nu)}$$

■ HypExplicit

■ Description

HypExplicit[exp, ν] expresses Hypergeometric functions in exp by their definition in terms of a sum (the Sum is omitted and ν is the summation index).

■ See also: HypergeometricIR.

■ Examples

```
Hypergeometric2F1[a, b, c, z]
```

$${}_2F_1(a, b, c; z)$$

```
HypExplicit[%, ν]
```

$$\frac{z^{\nu}\,\Gamma(c)\,\Gamma(a+\nu)\,\Gamma(b+\nu)}{\Gamma(a)\,\Gamma(b)\,\Gamma(\nu+1)\,\Gamma(c+\nu)}$$

```
HypergeometricPFQ[{a, b, c}, {d, e}, z]
```

$${}_3F_2(a, b, c; d, e; z)$$

```
HypExplicit[%, ν]
```

$$\frac{z^{\nu}\,\Gamma(d)\,\Gamma(e)\,\Gamma(a+\nu)\,\Gamma(b+\nu)\,\Gamma(c+\nu)}{\Gamma(a)\,\Gamma(b)\,\Gamma(c)\,\Gamma(\nu+1)\,\Gamma(d+\nu)\,\Gamma(e+\nu)}$$

■ Integrate2

■ Description

Integrate2 is like Integrate, but:
Integrate2[a_Plus, b__] := Map[Integrate2[#, b] &, a] (effectively, more linear algebra is done)
Integrate2[f[x] DeltaFunction[x], x] ⟶ f[0]
Integrate2[f[x] DeltaFunction[x0 − x], x] ⟶ f[x0].
Integrate2[f[x] DeltaFunction[a + b x], x] ⟶ Integrate[f[x] (1/abs[b]) DeltaFunction[a/b + x], x], where abs[b] ⟶ b, if b is a Symbol, and if b = −c, then abs[−c] ⟶ c, i.e., the variable contained in b is supposed to be positive.
π^2 is replaced by 6 Zeta2.
Integrate2[1/(1−y), {y,x,1}] is intepreted as distribution, i.e. as Integrate2[−1/(1−y), {y,0,x}] ⟶ Log[1−y].
Integrate2[1/(1−x), {x,0,1}] ⟶ 0.

■ See also: DeltaFunction, Integrate3, SumS, SumT.

■ Examples

```
Integrate2[Log[1 + x] Log[x] / (1 - x), {x, 0, 1}] // Timing
```

$$\{0.26\,\text{Second}, \zeta(3) - \frac{3}{2}\zeta(2)\log(2)\}$$

Since Integrate2 uses table–look–up methods it is much faster than *Mathematica*'s Integrate.

```
Integrate2[PolyLog[2, x^2], {x, 0, 1}]
```

$$\zeta(2) + 4\log(2) - 4$$

```
Integrate2[PolyLog[3, -x], {x, 0, 1}]
```

$$\frac{\zeta(2)}{2} - \frac{3\zeta(3)}{4} - 2\log(2) + 1$$

```
Integrate2[PolyLog[3, 1 / (1 + x)], {x, 0, 1}]
```

$$-\log(2)\zeta(2) + \frac{3\zeta(3)}{4} + \frac{\log^3(2)}{3} - \log^2(2) + 2\log(2)$$

```
Integrate2[DeltaFunction[1 - x] f[x], {x, 0, 1}]
```

$$f(1)$$

Integrate2 does integration in a Hadamard sense, i.e., $\int_0^1 f(x)\,dx$ means actually expanding the result of $\int_\delta^{1-\delta} f(x)\,dx$ up do $O(\delta)$ and neglecting all δ–dependent terms.
E.g. $\int_\delta^{1-\delta} 1/(1-x)\,dx = -\log(1-x)\,|_\delta^{1-\delta} = -\log(\delta) + \log(1) \to 0$.

```
Integrate2[1 / (1 - x), {x, 0, 1}]
```

0

In the physics literature sometimes the "+" notation is used. In FeynCalc the $(\frac{1}{1-x})_+$ is represented by PlusDistribution[1/(1 − x)] or just 1/(1 − x).

```
Integrate2[PlusDistribution[1 / (1 - x)], {x, 0, 1}]
```

0

```
Integrate2[PolyLog[2, 1 - x] / (1 - x) ^ 2, {x, 0, 1}]
```

$2-\zeta(2)$

```
Integrate2[(Log[x] Log[1 + x]) / (1 + x), {x, 0, 1}]
```

$$-\frac{\zeta(3)}{8}$$

```
Integrate2[Log[x] ^ 2 / (1 - x), {x, 0, 1}]
```

$2\zeta(3)$

```
Integrate2[PolyLog[2, -x] / (1 + x), {x, 0, 1}]
```

$$\frac{\zeta(3)}{4}-\frac{1}{2}\zeta(2)\log(2)$$

```
Integrate2[Log[x] PolyLog[2, x], {x, 0, 1}]
```

$3-2\zeta(2)$

```
Integrate2[x PolyLog[3, x], {x, 0, 1}]
```

$$-\frac{\zeta(2)}{4}+\frac{\zeta(3)}{2}+\frac{3}{16}$$

```
Integrate2[(Log[x] ^ 2 Log[1 - x]) / (1 + x), {x, 0, 1}]
```

$$\log^2(2)\zeta(2)-4\,\mathrm{Li}_4\left(\frac{1}{2}\right)-\frac{\log^4(2)}{6}+\frac{\pi^4}{90}$$

```
Integrate2[PolyLog[2, ((x (1 - z) + z) (1 - x + x z)) / z] / (1 - x + x z), {x, 0, 1}]
```

$$\frac{\log^3(z)}{6(1-z)}-\frac{\log(1-z)\log^2(z)}{1-z}-\frac{\log(z+1)\log^2(z)}{1-z}-\frac{i\pi\log^2(z)}{2(1-z)}-\frac{\zeta(2)\log(z)}{1-z}+\frac{4\log(1-z)\log(z+1)\log(z)}{1-z}+$$
$$\frac{2i\pi\log(z+1)\log(z)}{1-z}-\frac{2\,\mathrm{Li}_2(1-z)\log(z)}{1-z}-\frac{2\,\mathrm{Li}_2(-z)\log(z)}{1-z}-\frac{2\log^3(z+1)}{3(1-z)}-\frac{2i\pi\log^2(1-z)}{1-z}-$$
$$\frac{2i\pi\log^2(z+1)}{1-z}+\frac{i\pi\zeta(2)}{1-z}-\frac{10\zeta(2)\log(1-z)}{1-z}+\frac{14\zeta(2)\log(z+1)}{1-z}+\frac{4i\pi\log(1-z)\log(z+1)}{1-z}-\frac{2\,S_{12}(1-z)}{1-z}+$$
$$\frac{4\log(1-z)\,\mathrm{Li}_2(-z)}{1-z}+\frac{2i\pi\,\mathrm{Li}_2(-z)}{1-z}+\frac{4\,\mathrm{Li}_3(1-z)}{1-z}+\frac{2\,\mathrm{Li}_3(-z)}{1-z}+\frac{4\,\mathrm{Li}_3\left(\frac{1}{z+1}\right)}{1-z}+\frac{4\,\mathrm{Li}_3\left(-\frac{z+1}{1-z}\right)}{1-z}-\frac{4\,\mathrm{Li}_3\left(\frac{z+1}{1-z}\right)}{1-z}-\frac{2\zeta(3)}{1-z}$$

```
Apart[Integrate2[x^(OPEm - 1) PolyLog[3, 1 - x], {x, 0, 1}], OPEm]
```

$$-\frac{\zeta(2)}{m-1}-\frac{\zeta(2)}{m^2}+\frac{-S_1(m-2)\,\zeta(2)+\zeta(2)+S_{12}(m)+\zeta(3)}{m}$$

```
DataType[OPEm, PositiveInteger]
```

True

```
Integrate2[x^(OPEm - 1) DeltaFunction[1 - x], {x, 0, 1}]
```

1

This is the polarized non–singlet spin splitting function whose first moment vanishes.

```
tt = SplittingFunction[PQQNS]
```

$$\left(-4\,(x+1)\log^2(x)-8\left(2\,x+\frac{3}{1-x}\right)\log(x)-\frac{16\,(x^2+1)\log(1-x)\log(x)}{1-x}-40\,(1-x)+\delta(1-x)\,(-24\,\zeta(2)+48\,\zeta(3)+3)\right)C_F^2+$$
$$N_f\left(\frac{88\,x}{9}+\left(-\frac{16\,\zeta(2)}{3}-\frac{2}{3}\right)\delta(1-x)-\frac{8\,(x^2+1)\log(x)}{3\,(1-x)}-\frac{80}{9}\left(\frac{1}{1-x}\right)_+-\frac{8}{9}\right)C_F-$$
$$8\left(C_F-\frac{C_A}{2}\right)\left(4\,(1-x)+2\,(x+1)\log(x)+\frac{(x^2+1)\,(\log^2(x)-4\log(x+1)\log(x)-2\,\zeta(2)-4\,\mathrm{Li}_2(-x))}{x+1}\right)C_F+$$
$$C_A\left(\frac{4\,(x^2+1)\log^2(x)}{1-x}-\frac{4}{3}\left(5\,x-\frac{22}{1-x}+5\right)\log(x)+\frac{4}{9}\,(53-187\,x)+\right.$$
$$\left.8\,(x+1)\,\zeta(2)+\left(\frac{536}{9}-16\,\zeta(2)\right)\left(\frac{1}{1-x}\right)_+\!+\delta(1-x)\left(\frac{88\,\zeta(2)}{3}-24\,\zeta(3)+\frac{17}{3}\right)\right)C_F$$

```
Integrate2[tt, {x, 0, 1}] // Timing
```

{1.78 Second, 0}

```
Integrate2[DeltaFunction[1-x] f[x],{x,0,1}]
```

$f(1)$

■ Integrate3

■ Description

Integrate3 contains the integral table used by Integrate2. Integration is performed in a distributional sense.

■ See also: Integrate2.

■ Examples

```
Integrate3[x^OPEm Log[x], {x, 0, 1}]
```

$$-\frac{1}{(m+1)^2}$$

```
Integrate3[(x^OPEm Log[x] Log[1 - x]) / (1 - x), {x, 0, 1}]
```

$$\zeta(2)\, S_1(m) - S_{12}(m) - S_{21}(m) + \zeta(3)$$

```
Integrate3[DeltaFunctionPrime[1 - x], {x, 0, 1}]
```

0

```
Integrate3[f[x] DeltaFunctionPrime[1 - x], {x, 0, 1}]
```

$f'(1)$

```
Integrate3[1 / (1 - x), {x, 0, 1}]
```

0

■ IntegrateByParts

■ Description

IntegrateByParts[(1–t)^(a Epsilon–1)g[t], deriv,t] does an integration by parts of the definite integral over t from 0 to 1.

```
Options[IntegrateByParts]
```

{Hold→ False}

■ Examples

```
IntegrateByParts[(1 - t)^(a Epsilon - 1) g[t], (1 - t)^(a Epsilon - 1), t]
```

$$\frac{g'(t)\,(1-t)^{a\varepsilon}}{a\varepsilon} + \frac{g(0)}{a\varepsilon}$$

■ InverseMellin

■ Description

InverseMellin[exp,y,] performs the inverse Mellin transform of polynomials in OPEm.
The inverse transforms are not calculated but a table–look up is done.
WARNING: do not "trust" the results for the inverse Mellin transform involving SumT's; there is an unresolved inconsistency here (related to (–1)^*m*).

■ See also: DeltaFunction, Integrate2, OPEm, SumS, SumT.

■ Examples

```
InverseMellin[1 / OPEm, y]
```

y^{m-1}

```
InverseMellin[1 / (OPEm + 3), y]
```

y^{m+2}

```
InverseMellin[1, y]
```

$y^{m-1}\ \delta(1-y)$

```
InverseMellin[1 / OPEm^4, y]
```

$-\frac{1}{6}\ y^{m-1}\ \log^3(y)$

```
InverseMellin[1 / OPEm + 1, y]
```

$\delta(1-y)\ y^{m-1} + y^{m-1}$

```
InverseMellin[1 / i + 1, y, i]
```

$\delta(1-y)\ y^{i-1} + y^{i-1}$

The inverse operation to InverseMellin is done by Integrate2.

```
Integrate2[InverseMellin[1 / OPEm, y], {y, 0, 1}]
```

$\frac{1}{m}$

Below is a list of all built–in basic inverse Mellin transforms

```
list={1,
1/(OPEm + n),
1/(-opem + n),
PolyGamma[0,OPEm],
SumS[1,-1+OPEm],
SumS[1,-1+OPEm]/(OPEm-1),
SumS[1,-1+OPEm]/(1-OPEm),
SumS[1,-1+OPEm]/(OPEm+1),
SumS[1,-1+OPEm]/OPEm^2,
SumS[1,-1+OPEm]/OPEm,
SumS[1,-1+OPEm]^2/OPEm,
SumS[2,-1+OPEm],
SumS[2,-1+OPEm]/OPEm,
SumS[3,-1+OPEm],
SumS[1,1,-1+OPEm],
SumS[1,OPEm-1]^2,
SumS[1,2,-1+OPEm],
SumS[2,1,-1+OPEm],
SumS[1, -1 + OPEm]^3,
SumS[1, -1 + OPEm]*SumS[2, -1 + OPEm],
SumS[1,1,1,-1+OPEm]
};

SetAttributes[im, HoldAll];
    im[z_] := SequenceForm[HoldForm[z], " ≐ ",          InverseMellin[z, y]];

im[OPEm^(-3)]
```

$$\frac{1}{m^3} \doteq \frac{1}{2}\, y^{m-1} \log^2(y)$$

```
im[OPEm^(-2)]
```

$$\frac{1}{m^2} \doteq -y^{m-1} \log(y)$$

```
im[PolyGamma[0, OPEm]]
```

$$\psi^{(0)}(m) \doteq -\gamma\, \delta(1-y)\, y^{m-1} - \left(\frac{1}{1-y}\right)_+ y^{m-1}$$

```
im[SumS[1, OPEm - 1]]
```

$$S_1(m-1) \doteq -y^{m-1} \left(\frac{1}{1-y}\right)_+$$

```
im[SumS[1, OPEm - 1] / (OPEm - 1)]
```

$$\frac{S_1(m-1)}{m-1} \simeq -y^{m-2}\log(1-y)$$

```
im[SumS[1, OPEm - 1] / (OPEm + 1)]
```

$$\frac{S_1(m-1)}{m+1} \simeq -y^{m-1} - \log(1-y)\,y^m + \log(y)\,y^m + y^m$$

```
im[SumS[1, -1 + OPEm] / OPEm^2]
```

$$\frac{S_1(m-1)}{m^2} \simeq y^{m-1}\left(-\frac{1}{2}\log^2(y) + \zeta(2) - \mathrm{Li}_2(y)\right)$$

```
im[SumS[1, -1 + OPEm] / OPEm]
```

$$\frac{S_1(m-1)}{m} \simeq y^{m-1}\left(\log(y) - \log(1-y)\right)$$

```
im[SumS[1, -1 + OPEm]^2 / OPEm]
```

$$\frac{S_1^2(m-1)}{m} \simeq y^{m-1}\left(\log^2(1-y) + \frac{\log^2(y)}{2} - 3\,\zeta(2) + \mathrm{Li}_2(1-y) + 2\,\mathrm{Li}_2(y)\right)$$

```
im[SumS[2, OPEm - 1]]
```

$$S_2(m-1) \simeq y^{m-1}\left(\zeta(2)\,\delta(1-y) + \frac{\log(y)}{1-y}\right)$$

```
im[SumS[2, OPEm - 1] / OPEm]
```

$$\frac{S_2(m-1)}{m} \simeq y^{m-1}\left(-\frac{1}{2}\log^2(y) + \zeta(2) - \mathrm{Li}_2(1-y)\right)$$

```
im[SumS[3, OPEm - 1]]
```

$$S_3(m-1) \simeq y^{m-1}\left(\delta(1-y)\,\zeta(3) - \frac{\log^2(y)}{2\,(1-y)}\right)$$

```
im[SumS[1, 1, OPEm - 1]]
```

$$S_{11}(m-1) \simeq y^{m-1}\left(\frac{\log(1-y)}{1-y}\right)_+$$

```
im[SumS[1, 2, OPEm - 1]]
```

$$S_{12}(m-1) \simeq y^{m-1}\left(-\zeta(3)\,\delta(1-y) - \zeta(2)\left(\frac{1}{1-y}\right)_+ + \frac{\mathrm{Li}_2(1-y)}{1-y}\right)$$

```
im[SumS[2, 1, OPEm - 1]]
```

$$S_{21}(m-1) \simeq y^{m-1}\left(2\,\zeta(3)\,\delta(1-y) - \zeta(2)\left(\frac{1}{1-y}\right)_+ + \frac{\mathrm{Li}_2(y)}{1-y}\right)$$

```
im[SumS[1, 1, 1, OPEm - 1]]
```

$$S_{111}(m-1) = -\frac{1}{2}\, y^{m-1} \left(\frac{\log^2(1-y)}{1-y}\right)_+$$

```
Clear[im];
```

■ Isolate

■ Description

Isolate[expr] substitutes abbreviations KK[i] for all Plus[...] (sub-sums) in expr. The inserted KK[i] have head HoldForm.
Isolate[expr, varlist] substitutes KK[i] for all subsums in expr which are free of any occurence of a member of the list varlist. Instead of KK any other head or a list of names of the abbreviations may be specified with the option IsolateNames.

```
Options[Isolate]
```

{IsolateNames → KK, IsolatePrint → False, IsolateSplit → ∞}

■ Examples

```
Isolate[a + b]
```

KK(3)

```
t1 = Isolate[(a + b) f + (c + d) f + e, f]
```

$e + f\,\mathrm{KK}(3) + f\,\mathrm{KK}(4)$

```
InputForm[t1]
```

```
e + f*HoldForm[KK[3]] + f*HoldForm[KK[4]]
```

```
{KK[1], t1, ReleaseHold[t1]}
```

$\{p_1^2 - 2\, p_1 \cdot p_2 + p_2^2,\ e + f\,\mathrm{KK}(3) + f\,\mathrm{KK}(4),\ e + (a+b)\, f + (c+d)\, f\}$

```
Isolate[a[z] (b + c (y + z)) + d[z] (y + z), {a, d}, IsolateNames → F]
```

$d(z)\, F(3) + a(z)\, F(4)$

```
?? F
```

Global`F

$F[3] = y + z$

$F[4] = b + c * \mathrm{HoldForm}[F[3]]$

```
Isolate[a - b - c - d - e, IsolateNames → L, IsolateSplit → 15]
```

$L(3)$

```
{L[2], L[1]}
```

$\{a - b, L(1)\}$

```
Clear[t1, L]
```

■ Kummer

■ Description

Kummer[i][exp] applies Kummer relation number i (i = 1, ... 24, 94,95,96) to all Hypergeometric2F1 in exp.
i = 94 corresponds to eq. 9.131.2,
i = 95 to eq. 9.132.1 and
i = 96 to eq. 9.132.2 in Gradsteyn & Ryzhik.

■ See also: HypergeometricAC.

■ Examples

```
Hypergeometric2F1[a, b, c, z] == Kummer[2][Hypergeometric2F1[a, b, c, z]]
```

$${}_2F_1(a, b, c, z) == (1-z)^{-a-b+c}\ {}_2F_1(c-a, c-b, c, z)$$

```
Hypergeometric2F1[a, b, c, z] == Kummer[3][Hypergeometric2F1[a, b, c, z]]
```

$${}_2F_1(a, b, c, z) == (1-z)^{-a}\ {}_2F_1\left(a, c-b, c, -\frac{z}{1-z}\right)$$

```
Hypergeometric2F1[a, b, c, z] == Kummer[4][Hypergeometric2F1[a, b, c, z]]
```

$${}_2F_1(a, b, c, z) == (1-z)^{-b}\ {}_2F_1\left(c-a, b, c, -\frac{z}{1-z}\right)$$

```
Hypergeometric2F1[a, b, c, 1 - z] == Kummer[6][Hypergeometric2F1[a, b, c, 1 - z]]
```

$${}_2F_1(a, b, c, 1-z) == z^{-a-b+c}\ {}_2F_1(c-b, c-a, c, 1-z)$$

```
Hypergeometric2F1[a, b, a + b + 1 - c, 1 - z] ==
 Kummer[6][Hypergeometric2F1[a, b, a + b + 1 - c, 1 - z]]
```

$${}_2F_1(a, b, a+b-c+1, 1-z) == z^{1-c}\ {}_2F_1(a-c+1, b-c+1, a+b-c+1, 1-z)$$

```
Hypergeometric2F1[a, b, c, 1 - z] == Kummer[7][Hypergeometric2F1[a, b, c, 1 - z]]
```

$${}_2F_1(a, b, c, 1-z) == z^{-a}\ {}_2F_1\left(a, c-b, c, -\frac{1-z}{z}\right)$$

```
Hypergeometric2F1[a, b, a + b + 1 - c, 1 - z] ==
 Kummer[7][Hypergeometric2F1[a, b, a + b + 1 - c, 1 - z]]
```

$${}_2F_1(a, b, a+b-c+1; 1-z) == z^{-a}\ {}_2F_1\left(a, a-c+1; a+b-c+1; -\frac{1-z}{z}\right)$$

```
Hypergeometric2F1[a, b, c, 1 - z] == Kummer[8][Hypergeometric2F1[a, b, c, 1 - z]]
```

$${}_2F_1(a, b, c; 1-z) == z^{-b}\ {}_2F_1\left(c-a, b, c; -\frac{1-z}{z}\right)$$

```
Hypergeometric2F1[a, b, a + b + 1 - c, 1 - z] ==
 Kummer[8][Hypergeometric2F1[a, b, a + b + 1 - c, 1 - z]]
```

$${}_2F_1(a, b, a+b-c+1; 1-z) == z^{-b}\ {}_2F_1\left(b-c+1, b, a+b-c+1; -\frac{1-z}{z}\right)$$

```
Hypergeometric2F1[a, b, c, z^(-1)] == Kummer[10][Hypergeometric2F1[a, b, c, z^(-1)]]
```

$${}_2F_1\left(a, b, c; \frac{1}{z}\right) == (1-z)^{-a-b+c}\,(-z)^{a+b-c}\ {}_2F_1\left(c-a, c-b, c; \frac{1}{z}\right)$$

```
Hypergeometric2F1[a, a + 1 - c, a + 1 - b, z^(-1)] ==
 Kummer[10][Hypergeometric2F1[a, a + 1 - c, a + 1 - b, z^(-1)]]
```

$${}_2F_1\left(a, a-c+1; a-b+1; \frac{1}{z}\right) == (1-z)^{-a-b+c}\,(-z)^{a+b-c}\ {}_2F_1\left(1-b, c-b, a-b+1; \frac{1}{z}\right)$$

```
Hypergeometric2F1[a, b, c, z^(-1)] == Kummer[11][Hypergeometric2F1[a, b, c, z^(-1)]]
```

$${}_2F_1\left(a, b, c; \frac{1}{z}\right) == \left(-\frac{1-z}{z}\right)^{-a}\,(-z)^{a}\ {}_2F_1\left(a, c-b, c; \frac{1}{1-z}\right)$$

```
Hypergeometric2F1[a, a + 1 - c, a + 1 - b, z^(-1)] ==
 Kummer[11][Hypergeometric2F1[a, a + 1 - c, a + 1 - b, z^(-1)]]
```

$${}_2F_1\left(a, a-c+1; a-b+1; \frac{1}{z}\right) == \left(-\frac{1-z}{z}\right)^{-a}\,(-z)^{a}\ {}_2F_1\left(a, c-b, a-b+1; \frac{1}{1-z}\right)$$

```
Hypergeometric2F1[a, b, c, z^(-1)] == Kummer[12][Hypergeometric2F1[a, b, c, z^(-1)]]
```

$${}_2F_1\left(a, b, c; \frac{1}{z}\right) == (1-z)^{-b}\,(-z)^{b-a}\ {}_2F_1\left(b, c-a, c; \frac{1}{1-z}\right)$$

```
Hypergeometric2F1[a, a + 1 - c, a + 1 - b, z^(-1)] ==
 Kummer[12][Hypergeometric2F1[a, a + 1 - c, a + 1 - b, z^(-1)]]
```

$${}_2F_1\left(a, a-c+1; a-b+1; \frac{1}{z}\right) == (1-z)^{-a+c-1}\,(-z)^{1-c}\ {}_2F_1\left(a-c+1, 1-b, a-b+1; \frac{1}{1-z}\right)$$

```
Hypergeometric2F1[a, b, c, z^(-1)] == Kummer[14][Hypergeometric2F1[a, b, c, z^(-1)]]
```

$${}_2F_1\left(a, b, c; \frac{1}{z}\right) == (1-z)^{-a-b+c}\,(-z)^{a+b-c}\ {}_2F_1\left(c-b, c-a, c; \frac{1}{z}\right)$$

```
Hypergeometric2F1[b + 1 - c, b, b + 1 - a, z^(-1)] ==
 Kummer[14][Hypergeometric2F1[b + 1 - c, b, b + 1 - a, z^(-1)]]
```

$${}_2F_1\left(b-c+1, b, -a+b+1; \frac{1}{z}\right) == (1-z)^{-a-b+c}\,(-z)^{a+b-c}\ {}_2F_1\left(1-a, c-a, -a+b+1; \frac{1}{z}\right)$$

```
Hypergeometric2F1[a, b, c, z^(-1)] == Kummer[15][Hypergeometric2F1[a, b, c, z^(-1)]]
```

$$ {}_2F_1\left(a, b, c, \frac{1}{z}\right) == (1-z)^{-b}\,(-z)^{b}\,{}_2F_1\left(b, c-a, c, \frac{1}{1-z}\right) $$

```
Hypergeometric2F1[b + 1 - c, b, b + 1 - a, z^(-1)] ==
 Kummer[15][Hypergeometric2F1[b + 1 - c, b, b + 1 - a, z^(-1)]]
```

$$ {}_2F_1\left(b-c+1, b, -a+b+1, \frac{1}{z}\right) == (1-z)^{-b}\,(-z)^{b}\,{}_2F_1\left(b, c-a, -a+b+1, \frac{1}{1-z}\right) $$

```
Hypergeometric2F1[a, b, c, z^(-1)] == Kummer[16][Hypergeometric2F1[a, b, c, z^(-1)]]
```

$$ {}_2F_1\left(a, b, c, \frac{1}{z}\right) == (1-z)^{-a}\,(-z)^{a}\,{}_2F_1\left(a, c-b, c, \frac{1}{1-z}\right) $$

```
Hypergeometric2F1[b + 1 - c, b, b + 1 - a, z^(-1)] ==
 Kummer[16][Hypergeometric2F1[b + 1 - c, b, b + 1 - a, z^(-1)]]
```

$$ {}_2F_1\left(b-c+1, b, -a+b+1, \frac{1}{z}\right) == (1-z)^{-b+c-1}\,(-z)^{b-c+1}\,{}_2F_1\left(b-c+1, 1-a, -a+b+1, \frac{1}{1-z}\right) $$

```
Hypergeometric2F1[a + 1 - c, b + 1 - c, 2 - c, z] ==
 Kummer[18][Hypergeometric2F1[a + 1 - c, b + 1 - c, 2 - c, z]]
```

$$ {}_2F_1(a-c+1, b-c+1; 2-c; z) == (1-z)^{-a-b+c}\,{}_2F_1(1-a, 1-b; 2-c; z) $$

```
Hypergeometric2F1[a, b, c, z] == Kummer[18][Hypergeometric2F1[a, b, c, z]]
```

$$ {}_2F_1(a, b, c; z) == (1-z)^{-a-b+c}\,{}_2F_1(c-a, c-b, c; z) $$

```
Hypergeometric2F1[a + 1 - c, b + 1 - c, 2 - c, z] ==
 Kummer[19][Hypergeometric2F1[a + 1 - c, b + 1 - c, 2 - c, z]]
```

$$ {}_2F_1(a-c+1, b-c+1; 2-c; z) == (1-z)^{-a+c-1}\,{}_2F_1\left(a-c+1, 1-b, 2-c, \frac{z}{z-1}\right) $$

```
Hypergeometric2F1[a, b, c, z] == Kummer[19][Hypergeometric2F1[a, b, c, z]]
```

$$ {}_2F_1(a, b, c; z) == (1-z)^{-a}\,{}_2F_1\left(a, c-b, c, \frac{z}{z-1}\right) $$

```
Hypergeometric2F1[a + 1 - c, b + 1 - c, 2 - c, z] ==
 Kummer[20][Hypergeometric2F1[a + 1 - c, b + 1 - c, 2 - c, z]]
```

$$ {}_2F_1(a-c+1, b-c+1; 2-c; z) == (1-z)^{-b+c-1}\,{}_2F_1\left(b-c+1, 1-a, 2-c, \frac{z}{z-1}\right) $$

```
Hypergeometric2F1[a, b, c, z] == Kummer[20][Hypergeometric2F1[a, b, c, z]]
```

$$ {}_2F_1(a, b, c; z) == (1-z)^{-b}\,{}_2F_1\left(b, c-a, c, \frac{z}{z-1}\right) $$

```
Hypergeometric2F1[c - a, c - b, c + 1 - a - b, 1 - z] ==
 Kummer[22][Hypergeometric2F1[c - a, c - b, c + 1 - a - b, 1 - z]]
```

$$ {}_2F_1(c-a, c-b, -a-b+c+1; 1-z) == z^{1-c}\,{}_2F_1(1-a, 1-b, -a-b+c+1; 1-z) $$

```
Hypergeometric2F1[a, b, c, 1 - z] == Kummer[22][Hypergeometric2F1[a, b, c, 1 - z]]
```

$$ {}_2F_1(a, b, c; 1-z) == z^{-a-b+c}\,{}_2F_1(c-b, c-a, c; 1-z) $$

```
Hypergeometric2F1[c - a, c - b, c + 1 - a - b, 1 - z] ==
 Kummer[23][Hypergeometric2F1[c - a, c - b, c + 1 - a - b, 1 - z]]
```

$${}_2F_1(c-a, c-b, -a-b+c+1; 1-z) == (1-z)^{a-c}\ {}_2F_1\left(c-a, 1-a, -a-b+c+1; 1-\frac{1}{1-z}\right)$$

```
Hypergeometric2F1[a, b, c, 1 - z] == Kummer[23][Hypergeometric2F1[a, b, c, 1 - z]]
```

$${}_2F_1(a, b, c; 1-z) == (1-z)^{-a}\ {}_2F_1\left(a, c-b, c; 1-\frac{1}{1-z}\right)$$

```
Hypergeometric2F1[c - a, c - b, c + 1 - a - b, 1 - z] ==
 Kummer[24][Hypergeometric2F1[c - a, c - b, c + 1 - a - b, 1 - z]]
```

$${}_2F_1(c-a, c-b, -a-b+c+1; 1-z) == z^{b-c}\ {}_2F_1\left(c-b, 1-b, -a-b+c+1; -\frac{1-z}{z}\right)$$

```
Hypergeometric2F1[a, b, c, 1 - z] == Kummer[24][Hypergeometric2F1[a, b, c, 1 - z]]
```

$${}_2F_1(a, b, c; 1-z) == z^{-b}\ {}_2F_1\left(b, c-a, c; -\frac{1-z}{z}\right)$$

```
Hypergeometric2F1[a, b, c, z] == Kummer[94][Hypergeometric2F1[a, b, c, z]]
```

$${}_2F_1(a, b, c; z) == \frac{\Gamma(a+b-c)\,\Gamma(c)\ {}_2F_1(c-a, c-b, -a-b+c+1; 1-z)\,(1-z)^{-a-b+c}}{\Gamma(a)\,\Gamma(b)} + \frac{\Gamma(c)\,\Gamma(-a-b+c)\ {}_2F_1(a, b, a+b-c+1; 1-z)}{\Gamma(c-a)\,\Gamma(c-b)}$$

```
Hypergeometric2F1[a, b, c, z] == Kummer[95][Hypergeometric2F1[a, b, c, z]]
```

$${}_2F_1(a, b, c; z) == \frac{\Gamma(b-a)\,\Gamma(c)\ {}_2F_1\left(a, c-b, a-b+1; \frac{1}{1-z}\right)(1-z)^{-a}}{\Gamma(b)\,\Gamma(c-a)} + \frac{\Gamma(a-b)\,\Gamma(c)\ {}_2F_1\left(b, c-a, -a+b+1; \frac{1}{1-z}\right)(1-z)^{-b}}{\Gamma(a)\,\Gamma(c-b)}$$

```
Hypergeometric2F1[a, b, c, z] == Kummer[96][Hypergeometric2F1[a, b, c, z]]
```

$${}_2F_1(a, b, c; z) == \frac{(-1)^a\,\Gamma(b-a)\,\Gamma(c)\ {}_2F_1\left(a, a-c+1; a-b+1; \frac{1}{z}\right)z^{-a}}{\Gamma(b)\,\Gamma(c-a)} + \frac{(-1)^b\,\Gamma(a-b)\,\Gamma(c)\ {}_2F_1\left(b, b-c+1; -a+b+1; \frac{1}{z}\right)z^{-b}}{\Gamma(a)\,\Gamma(c-b)}$$

■ Lagrangian

■ Description

Lagrangian[oqu]gives the unpolarized quark operator. Lagrangian[oqp]gives the polarized quark operator. Lagrangian["QCD"]gives part of the QCD Lagrangian.

■ See also: FeynRule.

■ Examples

```
Lagrangian["QCD"]
```

$$-\frac{1}{4}\left(F^a_{\alpha\beta}\,F^a_{\alpha\beta}\right)$$

■ Twist–2 operator product expansion operators

```
Lagrangian["ogu"]
```

$$\frac{1}{2}\,i^{m-1}\left(F^a_{\alpha\Delta}\,(D^{ab}_{\Delta})^{m-2}\,F^b_{\alpha\Delta}\right)$$

```
Lagrangian["ogp"]
```

$$\frac{1}{2}\,i^{m}\left(\epsilon^{\alpha\beta\gamma\Delta}\,F^a_{\beta\gamma}\,(D^{ab}_{\Delta})^{m-2}\,F^b_{\alpha\Delta}\right)$$

```
Lagrangian["oqu"]
```

$$i^{m}\,(\bar{\psi}\,\gamma\cdot\Delta\,D_{\Delta}{}^{m-1}\,\psi)$$

```
Lagrangian["oqp"]
```

$$i^{m}\,(\bar{\psi}\,\gamma^5\,\gamma\cdot\Delta\,D_{\Delta}{}^{m-1}\,\psi)$$

■ LCD

■ Description

LCD[m,n,r,s] evaluates to D–dimensional LeviCivita[m,n,r,s] by virtue of FeynCalcInternal.
LCD[m,...][p,...] evaluates to D–dimensional
LeviCivita[m,...][p,...]
applying FeynCalcInternal.

You need to do SetOptions[Eps, Dimension->D] before LCD can be used as D–dimensional Levi–Civita input function.

■ See also: LeviCivita, LC.

■ Examples

```
SetOptions[Eps, Dimension → D];
```

```
LCD[μ, ν, ρ, σ]
```

$$\epsilon^{\mu\nu\rho\sigma}$$

```
% // FCI
```

$$\epsilon^{\mu\nu\rho\sigma}$$

```
% // InputForm
```

```
Eps[LorentzIndex[μ, D], LorentzIndex[ν, D],
  LorentzIndex[ρ, D], LorentzIndex[σ, D]]
```

```
LCD[μ, ν][p, q]
```

$$\epsilon^{\mu\nu pq}$$

```
% // FCI // InputForm
```

```
Eps[LorentzIndex[μ, D], LorentzIndex[ν, D],
  Momentum[p, D], Momentum[q, D]]
```

```
Factor2[Contract[LCD[μ, ν, ρ][p] LCD[μ, ν, ρ][q]]]
```

$$(1-D)\,(2-D)\,(3-D)\,p\cdot q$$

```
SetOptions[Eps, Dimension → 4]
```

{Dimension→ 4}

■ LC

■ Description

LC[m,n,r,s]evaluatesto 4–dimensionalLeviCivita[m,n,r,s]by virtueof applyingFeynCalcInternal.
LC[m,...][p,...] evaluatesto 4–dimensional
LeviCivita[m,...][p,...]applyingFeynCalcInternal.

■ Seealso: LeviCivita, LCD.

■ Examples

```
LC[μ, ν, ρ, σ]
```

$$\epsilon^{\mu\nu\rho\sigma}$$

```
% // FCI
```

$$\epsilon^{\mu\nu\rho\sigma}$$

```
% // InputForm
```

```
Eps[LorentzIndex[μ], LorentzIndex[ν],
  LorentzIndex[ρ], LorentzIndex[σ]]
```

```
LC[μ, ν][p, q]
```

$$\epsilon^{\mu\nu pq}$$

```
% // FCI // InputForm
```

```
Eps[LorentzIndex[μ], LorentzIndex[ν], Momentum[p],
  Momentum[q]]
```

```
Contract[LC[μ, ν, ρ][p] LC[μ, ν, ρ][q]]
```

$$-6\, p\cdot q$$

■ LeftPartialD

■ Description

LeftPartialD[μ] denotes $\overleftarrow{\partial}_\mu$ acting to the left.

■ See also: ExpandPartialD, PartialD, LeftRightPartialD, RightPartialD.

■ Examples

```
QuantumField[A, LorentzIndex[μ]].LeftPartialD[ν]
```

$$A_\mu \overleftarrow{\partial}_\nu$$

```
ExpandPartialD[%]
```

$$\partial_\nu A_\mu$$

```
InputForm[%]
```

```
QuantumField[PartialD[LorentzIndex[ν]], A,
  LorentzIndex[μ]]
```

```
InputForm[LeftPartialD[μ]]
```

```
LeftPartialD[LorentzIndex[μ]]
```

```
QuantumField[A, LorentzIndex[μ]].QuantumField[A, LorentzIndex[ν]].LeftPartialD[ρ]
```

$$A_\mu A_\nu \overleftarrow{\partial}_\rho$$

```
ExpandPartialD[%]
```

$$A_\mu \partial_\rho A_\nu + \partial_\rho A_\mu A_\nu$$

```
InputForm[%]
```

```
QuantumField[A, LorentzIndex[μ]] .
   QuantumField[PartialD[LorentzIndex[ρ]], A,
    LorentzIndex[ν]] +
  QuantumField[PartialD[LorentzIndex[ρ]], A,
    LorentzIndex[μ]] .
   QuantumField[A, LorentzIndex[ν]]
```

■ LeftRightPartialD

■ Description

LeftRightPartialD[mu] denotes $\overleftrightarrow{\partial}_\mu$, acting to the left and right.
ExplitPartialD[LeftRightPartialD[mu]] gives 1/2(RightPartialD[mu] - LeftPartialD[mu]).

■ See also: ExplicitPartialD, ExpandPartialD, PartialD, LeftPartialD, LeftRightPartialD2, RightPartialD.

■ Examples

```
LeftRightPartialD[μ]
```

$$\overleftrightarrow{\partial}_\mu$$

```
ExplicitPartialD[%]
```

$$\frac{1}{2}\left(\vec{\partial}_\mu - \overleftarrow{\partial}_\mu\right)$$

```
LeftRightPartialD[μ].QuantumField[A, LorentzIndex[ν]]
```

$$\overleftrightarrow{\partial}_\mu \; A_\nu$$

```
ExpandPartialD[%]
```

$$\frac{\partial_\mu A_\nu}{2} - \frac{1}{2}\overleftarrow{\partial}_\mu \; A_\nu$$

```
QuantumField[A, LorentzIndex[μ]].LeftRightPartialD[ν].QuantumField[A, LorentzIndex[ρ]]
```

$$A_\mu \overleftrightarrow{\partial}_\nu \; A_\rho$$

```
ExpandPartialD[%]
```

$$\frac{1}{2}\left(A_\mu \; \partial_\nu A_\rho\right) - \frac{1}{2}\left(\partial_\nu A_\mu \; A_\rho\right)$$

■ LeftRightPartialD2

■ Description

LeftRightPartialD2[mu] denotes $\overleftrightarrow{\partial}_\mu$, acting to the left and right.
ExplitPartialD[LeftRightPartialD2[mu]] gives (RightPartialD[mu] + LeftPartialD[mu]).

■ See also: ExplicitPartialD, ExpandPartialD, PartialD, LeftPartialD, RightPartialD.

■ Examples

```
LeftRightPartialD2[μ]
```

$\overleftrightarrow{\partial}_\mu$

```
ExplicitPartialD[%]
```

$\overleftarrow{\partial}_\mu + \overrightarrow{\partial}_\mu$

```
LeftRightPartialD2[μ].QuantumField[A, LorentzIndex[ν]]
```

$\overleftrightarrow{\partial}_\mu \; A_\nu$

```
ExpandPartialD[%]
```

$\overleftarrow{\partial}_\mu \; A_\nu + \partial_\mu A_\nu$

```
QuantumField[A, LorentzIndex[μ]].
 LeftRightPartialD2[ν].QuantumField[A, LorentzIndex[ρ]]
```

$A_\mu \overleftrightarrow{\partial}_\nu \; A_\rho$

```
ExpandPartialD[%]
```

$A_\mu \partial_\nu A_\rho + \partial_\nu A_\mu \; A_\rho$

■ LeviCivita

■ Description

LeviCivita[μ,ν, ρ, σ] is an input function for the totally antisymmetric Levi–Civita tensor. It evaluates automatically to the internal representation Eps[LorentzIndex[μ], LorentzIndex[ν], LorentzIndex[ρ], LorentzIndex[σ]] (or with a second argument in LorentzIndex for the Dimension, if the option Dimension of LeviCivita is changed). LeviCivita[μ,ν ...][p, ...] evaluates to Eps[LorentzIndex[μ], LorentzIndex[ν], ..., Momentum[p], ...].

```
Options[LeviCivita]
```

{Dimension→ *D*}

■ Seealso: LC, LCD.

■ Examples

```
LeviCivita[α, β, γ, δ]
```

$\epsilon^{\alpha\beta\gamma\delta}$

```
LeviCivita[μ, ν][OPEDelta, p]
```

$\epsilon^{\mu\nu\Delta p}$

```
LeviCivita[][p, q, r, s]
```

ϵ^{pqrs}

```
LeviCivita[α, β, β, δ]
```

0

```
LeviCivita[μ, ν][OPEDelta, p] // InputForm
```

```
Eps[LorentzIndex[μ], LorentzIndex[ν],
  Momentum[OPEDelta], Momentum[p]]
```

```
t1 = LeviCivita[α, β, γ, ρ].LeviCivita[α, β, γ, ρ]
```

$\epsilon^{\alpha\beta\gamma\rho}\,\epsilon^{\alpha\beta\gamma\rho}$

```
Contract[t1]
```

−24

```
SetOptions[LeviCivita, Dimension → D]; SetOptions[Eps, Dimension → D];
```

```
Contract[ LeviCivita[α, β, γ, ρ].LeviCivita[α, β, γ, ρ]] // Factor2
```

$(1-D)\,(2-D)\,(3-D)\,D$

```
SetOptions[LeviCivita, Dimension → 4]; SetOptions[Eps, Dimension → 4];
```

```
LC[α, β, γ, δ]
```

$\epsilon^{\alpha\beta\gamma\delta}$

```
Clear[t1]
```

■ LorentzIndex

■ Description

LorentzIndex[μ] denotes a four-dimensional Lorentz index.
For other than four dimensions LorentzIndex[μ, D] or LorentzIndex[n], etc. LorentzIndex[μ, 4] simplifies to LorentzIndex[μ].

■ See also: ChangeDimension, Momentum.

■ Examples

This denotes a four–dimensional Lorentz index.

```
LorentzIndex[α]
```

α

An optional second argument can be given for a dimension different from 4.

```
LorentzIndex[α, n]
```

α

```
LorentzIndex[ComplexIndex[α]]
```

α^*

```
ComplexConjugate[LorentzIndex[μ]]
```

μ^*

```
ComplexConjugate[LorentzIndex[ComplexIndex[β]]]
```

β

Setting $LorentzIndices=True displays the dimension as a subindex.

```
$LorentzIndices = True;
 LorentzIndex[α, D ]
```

α_D

```
LorentzIndex[ν, 4]
```

ν

```
$LorentzIndices = False;
```

■ Map2

■ Description

Map2[f,exp] is equivalent to Map if Nterms[exp] > 0, otherwise Map2[f,exp] gives f[exp].

■ See also: NTerms.

■ Examples

```
Map2[f, a - b]
```

$f(a) + f(-b)$

```
Map2[f, x]
```

$f(x)$

```
Map2[f, {a, b, c}]
```

$f(\{a, b, c\})$

```
Map2[f, 1]
```

$f(1)$

■ MetricTensor

■ Description

MetricTensor[μ,ν] is the metric tensor. The default dimension is 4.

■ See also: FeynCalcExternal, FCE, FCI, MT, MTD.

■ Examples

```
MetricTensor[α, β]
```

$g^{\alpha\beta}$

```
Contract[% %]
```

4

```
MetricTensor[α, β, Dimension → n]
```

$g^{\alpha \beta}$

```
Contract[% %]
```

n

```
InputForm[MetricTensor[a, b]]
```

```
Pair[LorentzIndex[a], LorentzIndex[b]]
```

```
InputForm[MetricTensor[a, b, Dimension → D]]
```

```
Pair[LorentzIndex[a, D], LorentzIndex[b, D]]
```

```
InputForm[FCE[MetricTensor[a, b]]]
```

```
MT[a, b]
```

```
InputForm[FCE[MetricTensor[a, b, Dimension → D]]]
```

```
MTD[a, b]
```

■ MomentumCombine

■ Description

MomentumCombine[exp] is the inverse operation to MomentumExpand and ExpandScalarProduct. MomentumCombine combines also Pair's.

■ See also: ExpandScalarProduct, Momentum, MomentumExpand.

■ Examples

```
Momentum[p] - 2 Momentum[q] // MomentumCombine // InputForm
```

```
Momentum[p - 2*q]
```

```
t1 = FV[p, μ] + 2 FV[q, μ]
```

$p^{\mu} + 2\, q^{\mu}$

```
MomentumCombine[t1]
```

$(p + 2\, q)^{\mu}$

```
MomentumCombine[t1] // InputForm
```

```
Pair[LorentzIndex[μ], Momentum[p + 2*q]]
```

```
MomentumCombine[t1] // ExpandScalarProduct
```

$p^\mu + 2\,q^\mu$

```
InputForm[%]
```

```
Pair[LorentzIndex[μ], Momentum[p]] +
  2*Pair[LorentzIndex[μ], Momentum[q]]
```

```
Clear[t1]
```

■ MomentumExpand

■ Description

MomentumExpand[expr] expands Momentum[a+b,...] in expr into Momentum[a], Momentum[b], ...

■ See also: ExpandScalarProduct, MomentumCombine.

■ Examples

```
MomentumExpand[Momentum[p + q]] // InputForm
```

```
Momentum[p] + Momentum[q]
```

```
ScalarProduct[p + q, r]
```

$(p+q)\cdot r$

```
% // InputForm
```

```
Pair[Momentum[p + q], Momentum[r]]
```

```
MomentumExpand[ScalarProduct[p + q, r]]
```

$(p+q)\cdot r$

```
% // InputForm
```

```
Pair[Momentum[p] + Momentum[q], Momentum[r]]
```

```
MomentumExpand[ScalarProduct[p + q, r - p]]
```

$(p+q)\cdot(r-p)$

```
% // InputForm
```

```
Pair[Momentum[p] + Momentum[q], -Momentum[p] + Momentum[r]]
```

```
Calc[%]
```

$-p^2 - p\cdot q + p\cdot r + q\cdot r$

■ Momentum

■ Description

Momentum[p] is the head of a four-momentum (p). The internal representation of a four-dimensional p is Momentum[p]. For other than four dimensions Momentum[p, dim]. Momentum[p, 4] simplifies to Momentum[p].

■ See also: DiracGamma, Eps, LorentzIndex, MomentumExpand.

■ Examples

This is a four-dimensional momentum.

```
Momentum[p]
```

p

As an optional second argument the dimension must be specified if it is different from 4.

```
Momentum[p, D]
```

p

The dimension index is supressed in the output.

```
Momentum[p, d] // InputForm
```

```
Momentum[p, d]
```

```
a1 = Momentum[-q]
```

$-q$

```
a1 // InputForm
```

```
-Momentum[q]
```

```
a2 = Momentum[p - q] + Momentum[2 q]
```

$(p-q)+2\,q$

```
a2 // InputForm
```

```
Momentum[p - q] + 2*Momentum[q]
```

```
a2 // MomentumExpand // InputForm
```

```
Momentum[p] + Momentum[q]
```

```
a2 // MomentumCombine // InputForm
```

```
Momentum[p + q]
```

```
ChangeDimension[Momentum[p], d] // InputForm
```

```
Momentum[p, d]
```

```
Clear[a1, a2]
```

■ MTD

■ Description

MTD[μ,ν] is the metric tensor in D dimensions.

■ See also: FeynCalcExternal, FCE, FCI, MetricTensor, MT.

■ Examples

```
MTD[α, β]
```

$g^{\alpha\beta}$

```
Contract[MTD[α, β] MTD[α, β]]
```

D

```
MTD[α, β] // InputForm
```

```
MTD[α, β]
```

```
FCI[MTD[α, β]] // InputForm
```

```
Pair[LorentzIndex[α, D], LorentzIndex[β, D]]
```

```
FCE[FCI[MTD[μ, ν]]] // InputForm
```

```
MTD[μ, ν]
```

■ MT

■ Description

MT[μ,ν] is the metric tensor in 4 dimensions.

■ **See also: FeynCalcExternal, FCE, FCI, MetricTensor, MTD.**

■ Examples

```
MT[α, β]
```

$g^{\alpha\beta}$

```
Contract[MT[α, β] MT[α, β]]
```

4

```
MT[a, b] // InputForm
```

```
MT[a, b]
```

```
FCI[MT[a, b]] // InputForm
```

```
Pair[LorentzIndex[a], LorentzIndex[b]]
```

```
FCE[FCI[MT[a, b]]] // InputForm
```

```
MT[a, b]
```

■ Nielsen

■ Description

Nielsen[i,j,x] denotes Nielsen's polylogarithm.

```
Options[Nielsen]
```

{PolyLog → False}

■ **See also: SimplifyPolyLog.**

■ Examples

```
Nielsen[1, 2, x]
```

$S_{1\,2}(x)$

Numerical evaluation is done via N[Nielsen[n_,p_,x_]:=
(-1)^(n+p-1)/(n-1)!NIntegrate[Log[1-t]^p Log[t]^(n-1)/t,{t,0,1}];

```
N[Nielsen[1, 2, .45]]
```

0.0728716

Some special values are built in.

```
{Nielsen[1, 2, 0], Nielsen[1, 2, -1], Nielsen[1, 2, 1 / 2], Nielsen[1, 2, 1]}
```

$$\left\{0, \frac{\zeta(3)}{8}, \frac{\zeta(3)}{8}, \zeta(3)\right\}$$

```
Nielsen[1, 2, x, PolyLog → True]
```

$$\frac{1}{2}\log(x)\log^2(1-x) + \mathrm{Li}_2(1-x)\log(1-x) - \mathrm{Li}_3(1-x) + \zeta(3)$$

```
Nielsen[1, 3, x, PolyLog → True]
```

$$-\frac{1}{6}\log(x)\log^3(1-x) - \frac{1}{2}\mathrm{Li}_2(1-x)\log^2(1-x) + \mathrm{Li}_3(1-x)\log(1-x) - \mathrm{Li}_4(1-x) + \frac{\pi^4}{90}$$

```
Nielsen[3, 1, x, PolyLog → True]
```

$$\mathrm{Li}_4(x)$$

■ NTerms

■ Description

NTerms[x] is equivalent to Length if x is a sum; otherwise NTerms[x] returns 1, except NTerms[0] –> 0.

■ Examples

```
NTerms[a - b]
```

2

```
NTerms[a b c]
```

1

```
NTerms[9]
```

1

```
NTerms[0]
```

0

■ OneLoop

Documentation unfinished...; see also the old FeynCalc documentation at http://www.mertig.com/oldfc.

NOTICE: While OneLoop is restricted to 't Hooft Feynman gauge the function OneLoopSimplify does not have this restriction (but is slower).
OneLoop handles self energies, vertex and box–graphs (those only up to 3rd rank tensors in the integration variable).

WARNING: If you encounter anomalies. The default setting of $West is True,
i.e., the way tr(γ^μ γ^ν γ^ρ γ^σ γ^τ γ^λ γ^5)
is calculated in D dimensions has changed compared to the old FeynCalc. Keep in mind that the issue of γ^5 schemes is tricky.

■ Description

OneLoop[q, amplitude] calculates the one–loop Feynman diagram amplitude. The argument q denotes the integration variable, i.e., the loop momentum. OneLoop[name, amplitude] has as first argument a name of the amplitude.

```
Options[OneLoop]
```

```
{Apart2→ True, CancelQP→ True, DenominatorOrder→ False, Dimension→ D, FinalSubstitutions→ {},
  Factoring→ False, FormatType→ InputForm, InitialSubstitutions→ {}, IntermediateSubstitutions→ {},
  IsolateNames→ False, Mandelstam→ {}, OneLoopSimplify→ False, Prefactor→ 1, ReduceGamma→ False,
  ReduceToScalars→ False, SmallVariables→ {}, WriteOut→ False, WriteOutPaVe→ False}
```

■ See also: B0, C0, D0, OneLoopSimplify, TID, TIDL.

■ Examples

```
SP[k, r] FAD[{k, m}, k - p] // FCI
```

$$\frac{k\cdot r}{(k^2-m^2)\,(k-p)^2}$$

```
OneLoop[k, %]
```

$$i\,\pi^2\left(-\frac{B_0(0,m^2,m^2)\,p\cdot r\,m^2}{2\,p^2}+\frac{B_0(p^2,0,m^2)\,p\cdot r\,m^2}{2\,p^2}-\frac{p\cdot r\,m^2}{2\,p^2}+\frac{1}{2}\,B_0(p^2,0,m^2)\,p\cdot r\right)$$

■ OneLoopSimplify

■ Description

OneLoopSimplify[amp, q] simplifies the one–loop amplitude amp. The second argument denotes the integration momentum

```
Options[OneLoopSimplify]
```

{Collecting→ False, Dimension→ D, DimensionalReduction→False, DiracSimplify→ True, FinalSubstitutions→ {},
IntegralTable→ {}, OPE1Loop→ False, ScalarProductCancel→True, SUNNToCACF→ True, SUNTrace→ False}

■ See also: OneLoop, TID, TIDL.

■ Examples

```
SP[k, r] FAD[{k, m}, k - p] // FCI
```

$$\frac{k\cdot r}{(k^2-m^2)\,(k-p)^2}$$

```
OneLoopSimplify[%, k]
```

$$\frac{(m^2+p^2)\,p\cdot r}{2\,((k^2-m^2)\,(k-p)^2)\,p^2}-\frac{p\cdot r}{2\,(k^2-m^2)\,p^2}$$

```
OneLoopSimplify[% /. m → 0, k]
```

$$\frac{p\cdot r}{2\,(k^2\,(k-p)^2)}$$

```
FAD[k, k, k - p1, k - p2] FVD[k, μ] // FCI
```

$$\frac{k^\mu}{k^2\,k^2\,(k-p_1)^2\,(k-p_2)^2}$$

```
OneLoopSimplify[ %, k]
```

$$\frac{p_2^\mu\,p_1^2-p_1^\mu\,p_1\cdot p_2}{2\,(k^2\,k^2\,(k-p_1)^2)\,((p_1\cdot p_2)^2-p_1^2\,p_2^2)}+\frac{p_2^\mu\,p_1^2\,p_1\cdot p_2+p_1^\mu\,p_2^2\,p_1\cdot p_2-p_1^\mu\,p_1^2\,p_2^2-p_2^\mu\,p_1^2\,p_2^2}{2\,(k^2\,k^2\,(k-p_1)^2\,(k-p_2)^2)\,((p_1\cdot p_2)^2-p_1^2\,p_2^2)}-$$
$$\frac{p_2^\mu\,p_1\cdot p_2-p_1^\mu\,p_2^2}{2\,(k^2\,k^2\,(k-p_2)^2)\,((p_1\cdot p_2)^2-p_1^2\,p_2^2)}-\frac{p_2^\mu\,p_1^2-p_1^\mu\,p_1\cdot p_2-p_2^\mu\,p_1\cdot p_2+p_1^\mu\,p_2^2}{2\,(k^2\,(k-p_1)^2\,(k-p_2)^2)\,((p_1\cdot p_2)^2-p_1^2\,p_2^2)}$$

```
FCE[%] /. SPD[p1] → 0 // FCI
```

$$\frac{p_1^\mu\,p_2^2}{2\,(k^2\,k^2\,(k-p_1)^2\,(k-p_2)^2)\,p_1\cdot p_2}-\frac{p_1^\mu}{2\,(k^2\,k^2\,(k-p_1)^2)\,p_1\cdot p_2}-$$
$$\frac{p_2^\mu\,p_1\cdot p_2-p_1^\mu\,p_2^2}{2\,(k^2\,k^2\,(k-p_2)^2)\,(p_1\cdot p_2)^2}-\frac{-p_1^\mu\,p_1\cdot p_2-p_2^\mu\,p_1\cdot p_2+p_1^\mu\,p_2^2}{2\,(k^2\,(k-p_1)^2\,(k-p_2)^2)\,(p_1\cdot p_2)^2}$$

```
OneLoopSimplify[FAD[k - p1, k - p2] SP[k, l]^2, k]
```

$$-\frac{D(l\cdot p_1)^2+2\,D\,l\cdot p_2\;l\cdot p_1-4\,l\cdot p_2\;l\cdot p_1+D(l\cdot p_2)^2-l^2\,p_1^2+2\,l^2\,p_1\cdot p_2-l^2\,p_2^2}{4\,(1-D)\,(k^2\,(k-p_1+p_2)^2)}$$

■ OPEDelta

■ Description

OPEDelta is a lightlike axial vector as used e.g. in the operator product expansion in QCD.

■ See also: Twist2QuarkOperator.

■ Examples

```
FourVector[OPEDelta, μ]
```

Δ^{μ}

```
Contract[% %]
```

0

```
SP[OPEDelta, OPEDelta]
```

0

■ OPEi, OPEj, OPEk, OPEl, OPEm, OPEn, OPEo

■ Description

OPEi, etc. are variables with DataType PositiveInteger which are used in functions relating to the operator product expansion.

■ Examples

```
OPEi
```

i

```
DataType[OPEi, OPEj, OPEk, OPEl, OPEm, OPEn, OPEo, PositiveInteger]
```

{True, True, True, True, True, True, True}

```
PowerSimplify[{(-1)^(2 OPEi), (-1)^(2 OPEj), (-1)^(2 OPEk),
   (-1)^(2 OPEl), (-1)^(2 OPEm), (-1)^(2 OPEn), (-1)^(2 OPEo)}]
```

{1, 1, 1, 1, 1, 1, 1}

Re has been changed:

```
{Re[OPEi] > -3, Re[OPEi] > -2, Re[OPEi] > -1,   Re[OPEi] > 0, Re[OPEi] > 1}
```

{True, True, True, True, True}

```
{Re[-OPEi + OPEm] > 0, Re[-OPEi + OPEm] > 1, Re[-OPEi + OPEm] > 2}
```

{True, True, True}

```
{Re[OPEm] > -3, Re[OPEm] > -2, Re[OPEm] > -1,   Re[OPEm] > 0, Re[OPEm] > 1}
```

{True, True, True, True, True}

```
$Color = True;
    OPEm
```

m

```
$Color = False;
```

■ OPESumExplicit

■ Description

OPESumExplicit[exp] calculates OPESum's.

■ See also: OPESum, OPESumSimplify.

■ Examples

```
t1 = OPESum[A^i B^(m - i - 3), {i, 0, m - 3}]
```

$$\sum_{i=0}^{m-3} A^i\, B^{-i+m-3}$$

```
OPESumExplicit[t1]
```

$$\frac{A^{m-2}}{A-B} - \frac{B^{m-2}}{A-B}$$

```
t2 = OPESum[a^i b^(j - i) c^(m - j - 4), {i, 0, j}, {j, 0, m - 4}]
```

$$\sum_{j=0}^{m-4} \sum_{i=0}^{j} a^i\, b^{j-i}\, c^{-j+m-4}$$

```
OPESumExplicit[t2]
```

$$-\frac{c\,a^{m-2}}{(a-b)(a-c)(b-c)}+\frac{b\,a^{m-2}}{(a-b)(a-c)(b-c)}+\frac{c\,b^{m-2}}{(a-b)(a-c)(b-c)}-\frac{a\,b^{m-2}}{(a-b)(a-c)(b-c)}-\frac{b\,c^{m-2}}{(a-b)(a-c)(b-c)}+\frac{a\,c^{m-2}}{(a-b)(a-c)(b-c)}$$

```
Clear[t1, t2];
```

■ OPESum

■ Description

OPESum[exp{i, 0, m}] denotes a symbolic sum. The syntax is the same as for Sum.

■ See also: OPESumExplicit, OPESumSimplify.

■ Examples

```
t1 = OPESum[SO[p]^OPEi SO[k]^(OPEm - OPEi - 3), {OPEi, 0, OPEm - 3}]
```

$$\sum_{i=0}^{m-3}(\Delta\cdot k)^{-i+m-3}\,(\Delta\cdot p)^{i}$$

```
OPESumExplicit[t1]
```

$$\frac{(\Delta\cdot k)^{m-2}}{\Delta\cdot k-\Delta\cdot p}-\frac{(\Delta\cdot p)^{m-2}}{\Delta\cdot k-\Delta\cdot p}$$

```
t2 = OPESum[a^i b^(j - i) c^(m - j - 4), {i, 0, j}, {j, 0, m - 4}]
```

$$\sum_{j=0}^{m-4}\sum_{i=0}^{j}a^{i}\,b^{j-i}\,c^{-j+m-4}$$

```
OPESumExplicit[t2]
```

$$-\frac{c\,a^{m-2}}{(a-b)(a-c)(b-c)}+\frac{b\,a^{m-2}}{(a-b)(a-c)(b-c)}+\frac{c\,b^{m-2}}{(a-b)(a-c)(b-c)}-\frac{a\,b^{m-2}}{(a-b)(a-c)(b-c)}-\frac{b\,c^{m-2}}{(a-b)(a-c)(b-c)}+\frac{a\,c^{m-2}}{(a-b)(a-c)(b-c)}$$

```
Clear[t1, t2]
```

■ OPESumSimplify

■ Description

OPESumSimplify[exp] simplifies OPESum's in exp.

■ See also: OPESum, OPESumExplicit.

■ Examples

```
OPESum[(-SOD[p])^(OPEi+1) SOD[p-q]^(OPEm-OPEi-2),{OPEi,0,OPEm}]
```

$$\sum_{i=0}^{m} (-(\Delta \cdot p))^{i+1} (\Delta \cdot (p-q))^{-i+m-2}$$

```
OPESumSimplify[%]
```

$$-\sum_{i=0}^{m} (-1)^i (\Delta \cdot p)^i (\Delta \cdot (p-q))^{-i+m-2} \Delta \cdot p$$

```
OPESumSimplify[OPESum[{OPEi, 0, OPEm}] a^OPEi]
```

$$\sum_{i=0}^{m} a^i$$

```
OPESumSimplify[OPESum[{j, 0, i}, {i, 0, m}] a^(j - i) b^i]
```

$$\sum_{i=0}^{m} \sum_{j=0}^{i} a^{j-i} b^j$$

```
% // InputForm
```

```
OPESum[a^(-i + j)*b^i, {i, 0, m}, {j, 0, i}]
```

■ Pair

■ Description

Pair[x,y] is the head of a special pairing used in the internal representation:
x and y may have heads LorentzIndex or Momentum. If both x and y have head LorentzIndex, the metric tensor is understood. If x and y have head Momentum, a scalar product is meant. If one of x and y has head LorentzIndex and the other Momentum, a Lorentz vector p^μ is understood.

■ **See also: FourVector, FV, FVD, MetricTensor, MT, MTD, ScalarProduct, SP, SPD.**

■ Examples

This represents a four–dimensional metric tensor.

```
Pair[LorentzIndex[α], LorentzIndex[β]]
```

$g^{\alpha\beta}$

This is a D–dimensional metric tensor.

```
Pair[LorentzIndex[α, D], LorentzIndex[β, D]]
```

$g^{\alpha\beta}$

```
Pair[LorentzIndex[α], Momentum[p]]
```

p^{α}

```
Pair[Momentum[q], Momentum[p]]
```

$p\cdot q$

```
Pair[Momentum[p], Momentum[p]]
```

p^2

```
Pair[Momentum[p - q], Momentum[p]]
```

$p\cdot(p-q)$

```
Pair[Momentum[p], Momentum[p]]^2
```

p^4

```
Pair[Momentum[p], Momentum[p]]^3
```

p^6

```
Pair[LorentzIndex[α, n - 4], LorentzIndex[β]] // Contract
```

0

```
ExpandScalarProduct[Pair[Momentum[p - q], Momentum[p]]]
```

$p^2 - p\cdot q$

```
Pair[Momentum[-q], Momentum[p]] + Pair[Momentum[q], Momentum[p]]
```

0

■ PartialD

■ Description

PartialD[μ] denotes the four–dimensional ∂_μ. PartialD is used to denote derivative fields. PartialD[LorentzIndex[μ,D]] denotes the D–dimensional ∂_μ.

■ See also: ExpandPartialD, LeftRightPartialD, RightPartialD.

■ Examples

```
QuantumField[A, {μ}].LeftPartialD[ν]
```

$A_\mu \overleftarrow{\partial}_\nu$

```
ExpandPartialD[%]
```

$\partial_\nu A_\mu$

```
InputForm[%]
```

```
QuantumField[PartialD[LorentzIndex[ν]], A,
  LorentzIndex[μ]]
```

■ PaVe

■ Description

PaVe[i,j,... {p10,p12,...},{m1^2,mw^2,...}] denotes the invariant (and scalar) Passarino–Veltman integrals, i.e. the coefficient functions of the tensor integral decomposition. Joining plist and mlist gives the same conventions as for A0, B0, C0, D0. Automatic simlifications are performed for the coefficient functions of two–point integrals and for the scalar integrals.

For a more detailed description see the manual (no changes from there basically):
http://www.mertig.com/oldfc

■ See also: PaVeReduce.

■ Examples

Some of the PaVe's reduce to special cases.

```
PaVe[0, {a, b, c, d, e, f}, {m1, m2, m3, m4}]
```

$D_0(a, b, c, d, e, f, m1, m2, m3, m4)$

```
PaVe[0, 0, {pp}, {m^2, M^2}]
```

$$\frac{1}{3} B_0(pp, m^2, M^2)\, m^2 + \frac{1}{18}(3\, m^2 + 3\, M^2 - pp) + \frac{1}{6}\left(B_0(0, M^2, M^2)\, M^2 + M^2 + (m^2 - M^2 + pp)\left(\frac{(M^2 - m^2)\,(B_0(pp, m^2, M^2) - B_0(0, m^2, M^2))}{2\, pp} - \frac{1}{2} B_0(pp, m^2, M^2)\right)\right)$$

■ PaVeOrder

■ Description

PaVeOrder[expr] orders the arguments of all D0 in expr in a standard way.
PaVeOrder[expr, PaVeOrderList -> { {..., s, u, ...}, {... m1^2,m2^2,...}, ...}] orders the arguments of all D0 in expr according to the specified ordering lists.
The lists may contain only a subsequence of the D0–variables.

■ See also: PaVeReduce.

■ Examples

```
PaVeOrder[D0[me2, me2, mw2, mw2, t, s, me2, 0, me2, 0], PaVeOrderList → {me2, me2, 0, 0}]
```

$D_0(me2, s, mw2, t, mw2, me2, me2, 0, 0, me2)$

```
PaVeOrder[D0[me2, me2, mw2, mw2, t, s, me2, 0, me2, 0], PaVeOrderList → {me2, me2, 0, 0}]
```

$D_0(me2, s, mw2, t, mw2, me2, me2, 0, 0, me2)$

```
PaVeOrder[D0[a, b, c, d, e, f, m12, m22, m32, m42] +
  D0[me2, me2, mw2, mw2, t, s, me2, 0, me2, 0], PaVeOrderList → {{me2, me2, 0, 0}, {f, e}}]
```

$D_0(a, d, c, b, f, e, m22, m12, m42, m32) + D_0(me2, s, mw2, t, mw2, me2, me2, 0, 0, me2)$

■ PaVeReduce

■ Description

PaVeReduce[expr] reduces all Passarino–Veltman integrals (i.e. all PaVe's) in expr down to scalar A0, B0, C0 and D0.

```
Options[PaVeReduce]
```

{Dimension → True, IsolateNames → False, Mandelstam → {}, PaVeOrderList → {}, WriteOutPaVe → False}

```
? WriteOutPaVe
```

WriteOutPaVe is an option for PaVeReduce and OneLoopSum. If set to a string, the results of all Passarino-Veltman PaVe's are written out

■ See also: FRH, PaVeOrder.

■ Examples

```
PaVeReduce[PaVe[1, 2, {s, m^2, m^2}, {m^2, m^2, M^2}], IsolateHead → L]
```

$L(14)$

```
L[10]
```

$4\,m^2 - s$

```
FRH[%]
```

$4\,m^2 - s$

```
PaVeReduce[PaVe[2, {SmallVariable[me2], mw2, t}, {SmallVariable[me2], 0, mw2}],
 WriteOutPaVe → "pte"]
```

$$\frac{B_0(0, \text{mw2}, \text{mw2})}{\text{mw2} - t} - \frac{B_0(t, \text{mw2}, \text{me2})}{\text{mw2} - t} + \frac{2}{\text{mw2} - t}$$

```
TableForm[ReadList["ptePaVe2Csmame2mw2tCsmame20mw2.s", String]]
```

(2/(mw2 - t) + B0[0, mw2, mw2]/(mw2 - t) -
  B0[t, mw2, SmallVariable[me2]]/(mw2 - t)
)

```
DeleteFile/@FileNames["ptePaVe2Csmame2mw2tCsmame20mw2.s"];
```

```
se=SmallVariable[ME2];
```

```
d122=PaVeReduce[PaVe[1,2,2,{se,MW2,MW2,se,S,T},{0,se,0,se}],Mandelstam->{S,T,
U,2 MW2},IsolateHead->F]
```

$F(20)$

```
Write2["fctd122.for",d122res==d122,FormatType->FortranForm];
```

FORTRAN generation::WARNING! Too long line encountered Checkline 15

FORTRAN generation::WARNING! Too long line encountered Checkline 21

```
ReadList["fctd122.for",String]//TableForm
```

```
      F(1) = F(1)
      F(2) = F(2)
      F(3) = y + z
      F(4) = b + c*F(3)
      F(5) = B0(MW2,ME2,0D0)
      F(6) = B0(S,0D0,0D0)
      F(7) = B0(T,ME2,ME2)
      F(8) = C0(MW2,MW2,T,ME2,0D0,ME2)
      F(9) = C0(MW2,S,ME2,ME2,0D0,0D0)
      F(10) = C0(T,ME2,ME2,ME2,ME2,0D0)
      F(11) = D0(MW2,MW2,ME2,ME2,T,S,ME2,0D0,ME2,0D0)
      F(12) = MW2 + S
      F(13) = 4*MW2 - T
      F(14) = MW2**2 - S*U
      F(15) = MW2 - S
      F(16) = 4*MW2**5 - 5*MW2**4*S - 16*MW2**3*S**2 + 4*MW2**2*S**3 +
     &  4*MW2*S**4 - MW2**4*U - 4*MW2**2*S**2*U + 8*MW2*S**3*U +
     &  4*MW2**2*S*U**2 + S**3*U**2 + S**2*U**3
      F(17) = MW2**2 - 4*MW2*S + 2*S**2 + S*U
      F(18) = 4*MW2**3 - 9*MW2**2*S + 2*S**3 - MW2**2*U - 4*MW2*S*U +
     &  5*S**2*U + 3*S*U**2
      F(19) = 2*MW2**6 - 8*MW2**5*S + 12*MW2**4*S**2 - 8*MW2**3*S**3 +
     &  2*MW2**2*S**4 - 2*MW2**5*T + 20*MW2**4*S*T -
     &  36*MW2**3*S**2*T + 20*MW2**2*S**3*T - 2*MW2*S**4*T -
     &  6*MW2**3*S*T**2 + 6*MW2**2*S**2*T**2 - 6*MW2*S**3*T**2 +
     &  4*MW2*S**2*T**3 - S**2*T**4
      F(20) = -F(12)/(2D0*F(13)*F(14)) -
     &  (S**2*T**2*F(10)*F(15))/(2D0*F(14)**3) +
     &  (S**3*T**2*F(11)*F(15))/(2D0*F(14)**3) +
     &  (S**2*T*F(9)*F(15)**2)/F(14)**3 -
     &  (F(5)*F(16))/(2D0*F(13)**2*F(14)**2*F(15)) +
     &  (S*F(6)*F(17))/(2D0*F(14)**2*F(15)) +
     &  (F(7)*F(12)*F(18))/(2D0*F(13)**2*F(14)**2) -
     &  (F(8)*F(12)*F(19))/(2D0*F(13)**2*F(14)**3)
      d122res= F(20)
```

```
DeleteFile/@FileNames["fctd122.for"];
```

■ PlusDistribution

■ Description

PlusDistribution[1/(1-x)^n] denotes a distribution (in the sense of the "+" prescription).

■ See also: Integrate2.

■ Examples

```
PlusDistribution[1 / (1 - x)]
```

$$\left(\frac{1}{1-x}\right)_+$$

```
PlusDistribution[Log[1 - x] / (1 - x)]
```

$$\left(\frac{\log(1-x)}{1-x}\right)_+$$

```
Integrate2[PlusDistribution[1 / (1 - x)], {x, 0, 1}]
```

0

```
Integrate2[PlusDistribution[Log[1 - x] / (1 - x)], {x, 0, 1}]
```

0

```
Integrate2[PlusDistribution[Log[1 - x]^2 / (1 - x)], {x, 0, 1}]
```

0

```
PlusDistribution[Log[x (1 - x)] / (1 - x)]
```

$$\frac{\log(x)}{1-x} + \left(\frac{\log(1-x)}{1-x}\right)_+$$

■ Polarization

■ Description

Polarization[k] = Polarization[k, I] is the head of a polarization momentum with (incoming) momentum k. A slashed polarization vector (e1 (k) slash) has
to be entered as DiracSlash[Polarization[k]]. The internal representation for a polarization vector e1 corresponding to a boson with four momentum k is:
Momentum[Polarization[k, I]].
With this notation transversality of polarization vectors is provided, i.e.,
Pair[Momentum[k], Momentum[Polarization[k, I]]] yields 0.
Polarization[k, -I] denotes the complex conjugate polarization.
Polarization is also an option of various functions related to the operator product expansion.

The setting 0 denotes the unpolarized and 1 the polarized case.

■ See also: PolarizationVector.

■ Examples

```
Polarization[k]
```

$\varepsilon(k)$

```
Polarization[k] // InputForm
```

```
Polarization[k, I]
```

```
Polarization[k] // ComplexConjugate
```

$\varepsilon^*(k)$

```
Polarization[k] // ComplexConjugate // InputForm
```

```
Polarization[k, -I]
```

```
DiracSlash[Polarization[k]]
```

$\gamma \cdot \varepsilon(k)$

```
DiracSlash[Polarization[k]] // InputForm
```

```
DiracGamma[Momentum[Polarization[k, I]]]
```

```
Pair[ Momentum[k], Momentum[ Polarization[k,  I] ] ]
```

0

■ PolarizationSum

■ Description

PolarizationSum[nu,nu,...] defines (as abbreviations) several polarization sums. The first two arguments are the interpreted as Lorentz indices, all further ones are momenta. PolarizationSum performs no calculations.

```
Options[PolarizationSum]
```

{Dimension→ 4}

■ See also: Polarization.

■ Examples

```
PolarizationSum[μ, ν]
```

$$-g^{\mu\nu}$$

```
PolarizationSum[μ, ν, k]
```

$$\frac{k^\mu\, k^\nu}{k^2} - g^{\mu\nu}$$

```
PolarizationSum[μ, ν, k, n]
```

$$-g^{\mu\nu} - \frac{k^\mu\, k^\nu\, n^2}{(k\cdot n)^2} + \frac{n^\mu\, k^\nu}{k\cdot n} + \frac{k^\mu\, n^\nu}{k\cdot n}$$

```
PolarizationSum[μ, ν, k, p1 - p2]
```

$$-g^{\mu\nu} - \frac{k^\mu\, k^\nu\, p_1{}^2}{(k\cdot p_1 - k\cdot p_2)^2} + \frac{2\, k^\mu\, k^\nu\, p_1\cdot p_2}{(k\cdot p_1 - k\cdot p_2)^2} - \frac{k^\mu\, k^\nu\, p_2{}^2}{(k\cdot p_1 - k\cdot p_2)^2} + \frac{((p_1 - p_2)^\mu)\, k^\nu}{k\cdot p_1 - k\cdot p_2} + \frac{k^\mu\, ((p_1 - p_2)^\nu)}{k\cdot p_1 - k\cdot p_2}$$

■ PolarizationVector

■ Description

PolarizationVector[p,mu]gives a polarizationvector.

■ See also: Pair, Polarization.

■ Examples

```
PolarizationVector[k, μ]
```

$$\varepsilon_\mu(k)$$

```
Conjugate[PolarizationVector[k, μ]]
```

$$\varepsilon^*_\mu(k)$$

```
PolarizationVector[k, mu] // InputForm
```

```
Pair[LorentzIndex[mu], Momentum[Polarization[k, I]]]
```

```
a1 = PolarizationVector[k, μ] FourVector[k, μ]
```

$$k^\mu\, \varepsilon_\mu(k)$$

```
Contract[a1]
```

0

```
a2 = PolarizationVector[k - p, μ] FourVector[k, μ]
```

$k^{\mu}\ \varepsilon_{\mu}(k-p)$

```
Contract[a2]
```

$k\cdot(\varepsilon(k-p))$

```
a3 = PolarizationVector[k - p, μ] FourVector[p, μ]
```

$p^{\mu}\ \varepsilon_{\mu}(k-p)$

```
Contract[a3]
```

$k\cdot(\varepsilon(k-p))$

■ PowerSimplify

■ Description

PowerSimplify[exp] simplifies (-x)^a to (-1)^a x^a and (y-x)^n to (-1)^n (x-y)^n, thus assuming that the exponent is an integer (even if it is symbolic). Furthermore (-1)^(a+n) and I^(a+n) are expanded and (I)^(2 m) -> (-1)^m and (-1)^(n_Integer?EvenQ) ->1 and (-1)^(n_Integer?OddQ) -> (-1)^m and (-1)^(-n) ->(-1)^n and Exp[I m Pi] ->(-1)^m.

■ See also: DataType, OPEm.

■ Examples

```
PowerSimplify[(-1)^(2 OPEm)]
```

1

```
PowerSimplify[(-1)^(OPEm + 2)]
```

$(-1)^m$

```
PowerSimplify[(-1)^(OPEm - 2)]
```

$(-1)^m$

```
PowerSimplify[I^(2 OPEm)]
```

$(-1)^m$

■ PropagatorDenominatorExplicit

■ Description

PropagatorDenominatorExplicit[exp] changes each occurence of PropagatorDenominator[a,b] in exp into 1/(ScalarProduct[a,a]−b^2) and replaces FeynAmpDenominator by Identity.

■ See also: FeynAmpDenominator, PropagatorDenominator.

■ Examples

```
FAD[{q, m}, {q - p, 0}] // FCI
```

$$\frac{1}{(q^2-m^2)(q-p)^2}$$

```
PropagatorDenominatorExplicit[%] // FCE
```

$$\frac{1}{(q^2-m^2)(p^2-2\,p\cdot q+q^2)}$$

```
% // InputForm
```

```
1/((-m^2 + SPD[q, q])*(SPD[p, p] - 2*SPD[p, q] + SPD[q, q]))
```

■ PropagatorDenominator

■ Description

PropagatorDenominator[q, m] is a factor of the denominator of a propagator. If p is supposed to be D−dimensional enter: PropagatorDenominator[Momentum[p, D], m].
What is meant is 1/(q^2 − m^2). PropagatorDenominator[p] evaluates to PropagatorDenominator[p, 0].

■ See also: FeynAmpDenominator, PropagatorDenominatorExplicit.

■ Examples

```
PropagatorDenominator[p, m]
```

$$\frac{1}{p^2-m^2}$$

```
PropagatorDenominator[p]
```

$$\frac{1}{p^2}$$

```
t1 = PropagatorDenominator[q, m]
```

$$\frac{1}{q^2 - m^2}$$

```
InputForm[FCI[t1]]
```

```
PropagatorDenominator[Momentum[q, D], m]
```

```
InputForm[ChangeDimension[t1, D]]
```

```
PropagatorDenominator[Momentum[q, D], m]
```

```
PropagatorDenominatorExplicit[t1]
```

$$\frac{1}{q^2 - m^2}$$

```
InputForm[%]
```

```
(-m^2 + Pair[Momentum[q], Momentum[q]])^(-1)
```

```
Clear[t1]
```

■ QuantumField

■ Description

QuantumField is the head of quantized fields and their derivatives.
QuantumField[par, ftype, {lorind}, {sunind}] denotes a quantum field of type ftype with (possible) Lorentz–indices lorind and SU(N)–indices sunind. The optional first argument par denotes a partial derivative acting on the field.

■ See also: FeynRule, PartialD, ExpandPartialD.

■ Examples

This denotes a scalar field.

```
QuantumField[S]
```

S

```
QuantumField[AntiQuarkField]
```

$\bar{\psi}$

```
QuantumField[QuarkField]
```

ψ

This is a field with a Lorentz index.

```
QuantumField[B, {μ}]
```

B_μ

Color indices should be put after the Lorentz ones.

```
QuantumField[GaugeField, {μ}, {a}]
```

A_μ^a

```
% // InputForm
```

```
QuantumField[GaugeField, LorentzIndex[μ], SUNIndex[a]]
```

A_Δ^a is a short form for $\Delta \text{^} \mu\ A_\mu^a$.

```
QuantumField[A, {OPEDelta}, {a}]
```

A_Δ^a

The first list of indices is usually interpreted as type LorentzIndex, except for OPEDelta, which gets converted to type Momentum.

```
QuantumField[A, {OPEDelta}, {a}] // InputForm
```

```
QuantumField[A, Momentum[OPEDelta], SUNIndex[a]]
```

Derivatives of fields are denoted as follows.

```
QuantumField[PartialD[μ], A, {μ}]
```

$\partial_\mu A_\mu$

```
QuantumField[PartialD[OPEDelta], S]
```

$\partial_\Delta S$

```
QuantumField[PartialD[OPEDelta], A, {OPEDelta}, {a}]
```

$\partial_\Delta A_\Delta^a$

```
QuantumField[PartialD[OPEDelta]^OPEm, A, {OPEDelta}, {a}]
```

$(\partial_\Delta)^m A_\Delta^a$

```
QuantumField[QuantumField[A]] === QuantumField[A]
```

True

■ QuarkField

■ Description

QuarkField is the name of a fermionic field. QuarkField is just a name with no functional properties. Only typesetting rules are attached.

■ See also: AntiQuarkField, QuantumField.

■ Examples

```
QuarkField
```

ψ

■ QuarkGluonVertex

■ Description

QuarkGluonVertex[μ, a] gives the Feynman rule for the quark–gluon vertex.

```
Options[QuarkGluonVertex]
```

{CounterTerm → False, CouplingConstant → g_s, Dimension → D, Ψ → False, Polarization → 0}

■ See also: GluonVertex.

■ Examples

```
QuarkGluonVertex[μ, a]
```

$i\, g_s\, (T_a\, \gamma^\mu)$

```
QuarkGluonVertex[μ, a, CounterTerm → 1]
```

$$\frac{2\, i\, (C_F - \frac{C_A}{2})\, g_s^3\, S_n\, (T_a\, \gamma^\mu)}{\varepsilon}$$

```
QuarkGluonVertex[μ, a, CounterTerm → 2]
```

$$\frac{3\,i\,C_A\,g_s^3\,S_n\,(T_a\,\gamma^\mu)}{\varepsilon}$$

```
QuarkGluonVertex[μ, a, CounterTerm → 3]
```

$$\frac{2\,i\,(C_A + C_F)\,g_s^3\,S_n\,(T_a\,\gamma^\mu)}{\varepsilon}$$

```
QuarkGluonVertex[{p, μ, a}, {q}, {k}, OPE → True]
```

$$i\,g_s\,(T_a\,\gamma^\mu) + g_s\,\text{Q}\,(\gamma\cdot\Delta\;T_a)\left(\sum_{i=0}^{m-2}(-1)^i\,(k\cdot\Delta)^i\,(\Delta\cdot q)^{-i+m-2}\right)\Delta^\mu$$

```
QuarkGluonVertex[{p, μ, a}, {q}, {k}, OPE → False]
```

$$i\,g_s\,(T_a\,\gamma^\mu)$$

■ QuarkPropagator

■ Description

QuarkPropagator[p] is the massless quark propagator.
QuarkPropagator[{p,m}] gives the quark propagator with mass m.

```
Options[QuarkPropagator]
```

{CounterTerm→ False, CouplingConstant→ g_s, Dimension→ D, Loop→ 0, Q → False, Polarization→ 0}

■ See also: GluonPropagator, QuarkGluonVertex.

■ Examples

```
QuarkPropagator[p]
```

$$\frac{i\,\gamma\cdot p}{p^2}$$

```
QuarkPropagator[{p, m}]
```

$$\frac{i\,(m + \gamma\cdot p)}{p^2 - m^2}$$

```
QuarkPropagator[p, 1, 2]
```

$$\frac{i\,\gamma\cdot p}{p^2}$$

■ RightPartialD

■ Description

RightPartialD[μ] denotes ∂_μ, acting to the right.

■ See also: ExpandPartialD, PartialD, LeftPartialD.

■ Examples

```
RightPartialD[μ]
```

$\vec{\partial}_\mu$

```
RightPartialD[μ].QuantumField[A, LorentzIndex[μ]]
```

$\vec{\partial}_\mu \; A_\mu$

```
ExpandPartialD[%]
```

$\partial_\mu A_\mu$

```
% // InputForm
```

```
QuantumField[PartialD[LorentzIndex[μ]], A,
  LorentzIndex[μ]]
```

```
RightPartialD[μ] // InputForm
```

```
RightPartialD[LorentzIndex[μ]]
```

■ RussianTrick

■ Description

RussianTrick[exp,k, {q1,q2,p}] (=RussianTrick[exp,p,p,{q1,q2,p}]) does the integration by parts where p is the external momentum.
RussianTrick[exp,k,l, {q1,q2,p}] (=RussianTrick[exp,k,l]) does integration by parts where l is the momentum to be differentiated.

The result is an expression which is vanishing.

■ See also: FourDivergence, FourLaplacian.

■ Examples

```
t = RHI[{OPEm, 0, 0, 0, 0}, {1, 1, 1, 1, 1}]
```

$$T^{m0000}_{11111}$$

```
t // RHI2FC
```

$$\frac{(\Delta\cdot q_1)^m}{q_1^2\, q_2^2\, (q_1-p)^2\, (q_2-p)^2\, (q_1-q_2)^2}$$

```
RussianTrick[% // RHI2FC, q2]
```

$$-\frac{(\Delta\cdot q_1)^m}{q_1^2\, (q_1-p)^2\, (q_2-p)^2\, (q_2-q_1)^2\, (q_2-q_1)^2} - \frac{(\Delta\cdot q_1)^m}{q_1^2\, (q_1-p)^2\, (q_2-q_1)^2\, (q_2-p)^2\, (q_2-p)^2} - \frac{(4-D)\,(\Delta\cdot q_1)^m}{q_1^2\, q_2^2\, (q_1-p)^2\, (q_2-q_1)^2\, (q_2-p)^2} + \frac{(\Delta\cdot q_1)^m}{q_2^2\, (q_1-p)^2\, (q_2-p)^2\, (q_2-q_1)^2\, (q_2-q_1)^2} + \frac{(\Delta\cdot q_1)^m\, p^2}{q_2^2\, q_1^2\, (q_2-q_1)^2\, (q_1-p)^2\, (q_2-p)^2\, (q_2-p)^2}$$

```
FC2RHI[%]
```

$$T^{m0000}_{01112} - T^{m0000}_{10112} - T^{m0000}_{10121} + (D-4)\, T^{m0000}_{11111} + p^2\, T^{m0000}_{11121}$$

```
Solve2[%, t]
```

$$\left\{T^{m0000}_{11111} \to \frac{T^{m0000}_{01112} - T^{m0000}_{10112} - T^{m0000}_{10121} + p^2\, T^{m0000}_{11121}}{4-D}\right\}$$

```
Clear[t]
```

■ ScalarProduct

■ Description

ScalarProduct[p,q] is the input for scalarproduct. ScalarProduct[p,p] is equivalent to ScalarProduct[p]. Expansion of sums of momenta in ScalarProduct is done with ExpandScalarProduct. Scalarproducts may be set, e.g., ScalarProduct[a, b] = m^2; but a and b must not contain sums. Note that ScalarProduct[a] = m^2 actually sets also: Pair[Momentum[a, ___], Momentum[b, ___]] = m^2.
It is encouraged to always set ScalarProduct's **before** any calculation.
This improves the performance of FeynCalc.

■ See also: Calc, ExpandScalarProduct, Pair, SP, SPD.

■ Examples

```
ScalarProduct[p, q]
```

$p \cdot q$

```
ScalarProduct[p + q, -q]
```

$-(q \cdot (p + q))$

```
ScalarProduct[p, p]
```

p^2

```
ScalarProduct[q]
```

q^2

```
ScalarProduct[p, q] // InputForm
```

```
Pair[Momentum[p], Momentum[q]]
```

```
ScalarProduct[p, q, Dimension → D] // InputForm
```

```
Pair[Momentum[p, D], Momentum[q, D]]
```

```
ScalarProduct[p1, p2] = s / 2
```

$\frac{s}{2}$

```
ExpandScalarProduct[ ScalarProduct[p1 - q, p2 - k]]
```

$\frac{s}{2} + k \cdot q - k \cdot p_1 - q \cdot p_2$

```
Calc[ ScalarProduct[p1 - q, p2 - k]]
```

$\frac{s}{2} + k \cdot q - k \cdot p_1 - q \cdot p_2$

■ ScalarProductCancel

■ Description

ScalarProductCancel[exp, q1, q2, ...] cancels scalar products with propagators. ScalarProductCancel[exp] cancels simple cases.

```
SPC
```

ScalarProductCancel

```
SPC // Options
```

{ChangeDimension → *D*, Collecting → True,
FeynAmpDenominatorSimplify → False, FeynAmpDenominatorCombine → True}

```
? SPC
```

SPC is an abbreviation for ScalarProductCancel

■ Examples

■ See also: FeynAmpDenominatorSimplify.

```
t1 = SPD[q, p] FAD[{q, m}, {q - p, 0}] // FCI
```

$$\frac{p\cdot q}{(q^2-m^2)(q-p)^2}$$

```
ScalarProductCancel[t1, q]
```

$$-\frac{1}{2(q^2-m^2)}+\frac{1}{2(q-p)^2}+\frac{\frac{m^2}{2}+\frac{p^2}{2}}{(q^2-m^2)(q-p)^2}$$

```
FeynAmpDenominatorSimplify[%, q]
```

$$\frac{\frac{m^2}{2}+\frac{p^2}{2}}{(q^2-m^2)(q-p)^2}-\frac{1}{2(q^2-m^2)}$$

```
SPC[t1, q, FDS -> True]
```

$$\frac{\frac{m^2}{2}+\frac{p^2}{2}}{(q^2-m^2)(q-p)^2}-\frac{1}{2(q^2-m^2)}$$

```
t2 =  SPD[q2, p] SPD[q1, p] FAD[{q1, m}, {q2, m}, q1 - p, q2 - p, q2 - q1] // FCI
```

$$\frac{p\cdot q_1\ p\cdot q_2}{(q_1^2-m^2)(q_2^2-m^2)(q_1-p)^2(q_2-p)^2(q_2-q_1)^2}$$

```
SPC[t2, q1, q2, FDS → True]
```

$$\frac{1}{4((q_1^2-m^2)(q_2^2-m^2)(q_2-q_1)^2)}-\frac{1}{2((q_1^2-m^2)(q_2-p)^2(q_2-q_1)^2)}+\frac{-\frac{m^2}{2}-\frac{p^2}{2}}{(q_1^2-m^2)(q_2^2-m^2)(q_1-p)^2(q_2-q_1)^2}+$$
$$\frac{\frac{m^2}{2}+\frac{p^2}{2}}{q_1^2\,q_2^2((q_1-p)^2-m^2)(q_2-q_1)^2}+\frac{\frac{m^4}{4}+\frac{p^2\,m^2}{2}+\frac{p^4}{4}}{(q_1^2-m^2)(q_2^2-m^2)(q_1-p)^2(q_2-p)^2(q_2-q_1)^2}$$

```
Clear[t1, t2];
```

■ SD

■ Description

SD[i, j] is the (FeynCalc–external) Kronecker–delta for SU(N) with color indices i and j. SD[i,j] is transformed into SUNDelta[SUNIndex[i],SUNIndex[j]] by FeynCalcInternal.

■ See also: SUNDelta.

■ Examples

```
SD[a, b]
```

δ_{ab}

```
% // FCI // InputForm
```

```
SUNDelta[SUNIndex[a], SUNIndex[b]]
```

```
% // FCE // InputForm
```

```
SD[a, b]
```

■ SelectFree

■ Description

SelectFree[expr,a, b, ...] is equivalent to Select[expr,FreeQ2[#,{a,b, ...}]&], except the special cases: SelectFree[a,b] returns a and SelectFree[a,a] returns 1 (where a is not a product or a sum).

SelectFree is equivalent to Select1.

■ See also: FreeQ2, SelectNotFree.

■ Examples

```
SelectFree[a + b + f[a] + d, a]
```

$b+d$

```
SelectFree[x y, x]
```

y

```
SelectFree[2 x y z f[x], {x, y}]
```

$2 z$

```
SelectFree[a, b]
```

a

```
SelectFree[a, a]
```

1

```
SelectFree[1, c]
```

1

```
SelectFree[f[x], x]
```

1

■ SelectNotFree

■ Description

SelectNotFree[expr, x] returns that part of expr which is not free of any occurance of x.

SelectNotFree[expr, a, b, ...] is equivalent to Select[expr, !FreeQ2[#, {a, b, ...}]&], except the special cases: SelectNotFree[a, b] returns 1 and SelectNotFree[a, a] returns a (where a is not a product or a sum).

SelectNotFree is equivalent to Select2.

■ See also: FreeQ2, SelectFree.

■ Examples

```
SelectNotFree[a + b + f[a], a]
```

$a + f(a)$

```
SelectNotFree[2 x y f[x] z, {x, y}]
```

$x y f(x)$

```
SelectNotFree[a, b]
```

1

```
SelectNotFree[a + x, b]
```

0

```
SelectNotFree[a, a]
```

a

```
SelectNotFree[1, c]
```

1

```
SelectNotFree[f[x], x]
```

$f(x)$

■ Series2

■ Description

Series2 performs a series expansion around 0. Series2 is (up to the Gamma-bug) equivalent to Series, except that it applies Normal on the result and has an option FinalSubstitutions. Series2[f,e, n] is equivalent to Series2[f,{e, 0, n}].

```
Options[Series2]
```

{Collecting→ False, Factoring→ True, FinalSubstitutions→ {γ → 0}, Print→ False}

■ Examples

```
Series2[(x (1 - x)) ^ (δ / 2), δ, 1]
```

$$\frac{1}{2}\,\delta \log((1-x)\,x)+1$$

```
Series2[Gamma[x], x, 1]
```

$$\frac{\pi^2\,x}{12}+\frac{1}{x}$$

```
Series[Gamma[x], {x, 0, 1}]
```

$$\frac{1}{x}+O(x^0)$$

```
Series2[Gamma[x], x, 2]
```

$$-\frac{1}{3}\,\zeta(3)\,x^2+\frac{\pi^2\,x}{12}+\frac{1}{x}$$

```
Series2[Gamma[x], x, 2, FinalSubstitutions → {}] // FullSimplify
```

$$\frac{-(2\,\gamma^3+\gamma\,\pi^2+4\,\zeta(3))\,x^3+(6\,\gamma^2+\pi^2)\,x^2-12\,\gamma\,x+12}{12\,x}$$

```
Series[Gamma[x], {x, 0, 4}] // Normal // FullSimplify
```

$$\frac{-(2\gamma^3+\gamma\pi^2+4\zeta(3))x^3+(6\gamma^2+\pi^2)x^2-12\gamma x+12}{12x}$$

There is a table of expansions of special hypergeometric functions.

```
Series2[HypergeometricPFQ[{1, OPEm - 1, Epsilon / 2 + OPEm},
  {OPEm, OPEm + Epsilon}, 1],        {Epsilon, 0, 1}]
```

$$\frac{1}{2}\varepsilon\psi^{(1)}(m)(m-1)+\frac{2(m-1)}{\varepsilon}+1$$

```
Series2[HypergeometricPFQ[{1, OPEm, Epsilon / 2 + OPEm},
  {1 + OPEm, Epsilon + OPEm},  1], {Epsilon, 0, 1}]
```

$$\frac{1}{4}\varepsilon m\psi^{(0)}(m)^2+\frac{2m}{\varepsilon}+\frac{3}{4}\varepsilon m\psi^{(1)}(m)-\frac{1}{2}\varepsilon mS_{11}(m-1)+\frac{1}{24}\varepsilon m\pi^2$$

```
Hypergeometric2F1[1, Epsilon, 1 + 2 Epsilon, x]
```

$${}_2F_1(1,\varepsilon;2\varepsilon+1;x)$$

```
Series2[%, {Epsilon, 0, 3}]
```

$$\left(-\frac{1}{6}\log^3(1-x)+\log(x)\log^2(1-x)-\zeta(2)\log(1-x)-\frac{1}{6}\pi^2\log(1-x)+2\,\mathrm{Li}_3(1-x)+4\,\mathrm{Li}_3(x)-2\zeta(3)\right)\varepsilon^3+$$
$$\left(-\frac{1}{2}\log^2(1-x)+2\log(x)\log(1-x)-3\zeta(2)+2\,\mathrm{Li}_2(1-x)+\frac{\pi^2}{6}\right)\varepsilon^2-\log(1-x)\varepsilon+1$$

There are over 100 more special expansions of ${}_2F_1$ tabulated in Series2.m. The interested user can consult the source code (search for HYPERLIST).

■ SimplifyDeltaFunction

■ Description

SimplifyDeltaFunction[exp], simplifies f[x]*DeltaFunction[1-x] to Limit[f[x],x->1]DeltaFunction[1-x] and applies a list of transformation rules for DeltaFunctionPrime[1-x]*x^(OPEm-1)*f[x] where x^(OPEm-1) is suppressed in exp.

■ See also: DeltaFunction, DeltaFunctionPrime.

■ Examples

```
g[x] DeltaFunction[1 - x]
```

$$\delta(1-x)\,g(x)$$

```
SimplifyDeltaFunction[ %, x]
```

$$\delta(1-x)\left(\lim_{x\to 1} g(x)\right)$$

```
g[x] DeltaFunctionPrime[1 - x]
```

$$\delta'(1-x)\,g(x)$$

```
SimplifyDeltaFunction[ %, x]
```

$$\delta'(1-x)\left(\lim_{x\to 1} g(x)\right)+\delta(1-x)\left(\lim_{x\to 1} g'(x)\right)$$

```
x Log[x] DeltaFunctionPrime[1 - x]
```

$$x\,\delta'(1-x)\log(x)$$

```
SimplifyDeltaFunction[ %, x]
```

$$\delta(1-x)$$

```
PolyLog[2, 1 - x] DeltaFunctionPrime[1 - x]
```

$$\delta'(1-x)\,\mathrm{Li}_2(1-x)$$

```
SimplifyDeltaFunction[ %, x]
```

$$-\delta(1-x)$$

```
Log[x] PolyLog[2, 1 - x] DeltaFunctionPrime[1 - x]
```

$$\delta'(1-x)\log(x)\,\mathrm{Li}_2(1-x)$$

```
SimplifyDeltaFunction[ %, x]
```

$$0$$

```
PolyLog[3, 1 - x] DeltaFunctionPrime[1 - x]
```

$$\delta'(1-x)\,\mathrm{Li}_3(1-x)$$

```
SimplifyDeltaFunction[ %, x]
```

$$-\delta(1-x)$$

■ SimplifyPolyLog

■ Description

SimplifyPolyLog[y] performs several simplifications assuming that the variables occuring in the Log and PolyLog functions are between 0 and 1.

■ See also: Nielsen.

■ Examples

```
sip[y_] := y == SimplifyPolyLog[y]
```

```
sip[PolyLog[2, 1 / x]]
```

$$\text{Li}_2\left(\frac{1}{x}\right) == -\frac{1}{2}\log^2(x) + \log(1-x)\log(x) + i\pi\log(x) + \zeta(2) + \text{Li}_2(1-x)$$

```
sip[PolyLog[2, x]]
```

$$\text{Li}_2(x) == \zeta(2) - \log(1-x)\log(x) - \text{Li}_2(1-x)$$

```
sip[PolyLog[2, 1 - x^2]]
```

$$\text{Li}_2(1-x^2) == -\zeta(2) - 2\log(x)\log(x+1) + 2\,\text{Li}_2(1-x) - 2\,\text{Li}_2(-x)$$

```
sip[PolyLog[2, x^2]]
```

$$\text{Li}_2(x^2) == 2\zeta(2) - 2\log(1-x)\log(x) - 2\,\text{Li}_2(1-x) + 2\,\text{Li}_2(-x)$$

```
sip[PolyLog[2, -x / (1 - x)]]
```

$$\text{Li}_2\left(-\frac{x}{1-x}\right) == -\frac{1}{2}\log^2(1-x) + \log(x)\log(1-x) - \zeta(2) + \text{Li}_2(1-x)$$

```
sip[PolyLog[2, x / (x - 1)]]
```

$$\text{Li}_2\left(\frac{x}{x-1}\right) == -\frac{1}{2}\log^2(1-x) + \log(x)\log(1-x) - \zeta(2) + \text{Li}_2(1-x)$$

```
sip[Nielsen[1, 2, -x / (1 - x)]]
```

$$S_{1\,2}\left(-\frac{x}{1-x}\right) == S_{1\,2}(x) - \frac{1}{6}\log^3(1-x)$$

```
sip[PolyLog[3, -1 / x]]
```

$$\text{Li}_3\left(-\frac{1}{x}\right) == \frac{\log^3(x)}{6} + \zeta(2)\log(x) + \text{Li}_3(-x)$$

```
sip[PolyLog[3, 1 - x]]
```

True

```
sip[PolyLog[3, x^2]]
```

$$\text{Li}_3(x^2) == -2\log(1-x)\log^2(x) + 4\zeta(2)\log(x) - 4\,\text{Li}_2(1-x)\log(x) - 4\,S_{1\,2}(1-x) + 4\,\text{Li}_3(-x) + 4\zeta(3)$$

```
sip[PolyLog[3, -x / (1 - x)]]
```

$$\text{Li}_3\left(-\frac{x}{1-x}\right) == \frac{1}{6}\log^3(1-x) - \frac{1}{2}\log(x)\log^2(1-x) + \frac{1}{2}\log^2(x)\log(1-x) + \zeta(2)\log(1-x) - \zeta(2)\log(x) + S_{1\,2}(1-x) + \log(x)\,\text{Li}_2(1-x) - \text{Li}_3(1-x)$$

```
sip[PolyLog[3, 1 - 1 / x]]
```

$$\text{Li}_3\left(1-\frac{1}{x}\right) == \frac{\log^3(x)}{6} - \frac{1}{2}\log^2(1-x)\log(x) + \text{Li}_2(1-x)\log(x) + S_{1\,2}(1-x) + S_{1\,2}(x) - \log(1-x)\,\text{Li}_2(1-x) - \zeta(3)$$

```
sip[PolyLog[4, -x / (1 - x)]]
```

True

```
sip[Log[a + b / c]]
```

$$\log\left(a+\frac{b}{c}\right) == \log\left(\frac{b+a\,c}{c}\right)$$

```
sip[Log[1 / x]]
```

$$\log\left(\frac{1}{x}\right) == -\log(x)$$

```
sip[ArcTanh[x]]
```

$$\bar{\tanh}^{-1}(x) == \frac{1}{2}\log\left(-\frac{x+1}{1-x}\right)$$

```
sip[ArcSinh[x]]
```

$$\bar{\sinh}^{-1}(x) == \log\left(x+\sqrt{x^2+1}\right)$$

```
sip[ArcCosh[x]]
```

$$\bar{\cosh}^{-1}(x) == \log\left(x+\sqrt{x^2-1}\right)$$

```
Clear[sip]
```

■ SOD

■ Description

SOD[q] is a D–dimensional scalar product of OPEDelta with q. It is transformed into Pair[Momentum[q,D], Momentum[OPEDelta,D]] by FeynCalcInternal.

▪ See also: OPEDelta, Pair, ScalarProduct, SOD.

▪ Examples

```
SOD[p]
```

$\Delta \cdot p$

```
SOD[p - q]
```

$\Delta \cdot (p - q)$

```
SOD[p] // FCI // InputForm
```

```
Pair[Momentum[OPEDelta, D], Momentum[p, D]]
```

■ Solve2

▪ Description

Solve2 is equivalent to Solve, except that it works only for linear equations (and returns just a list) and accepts the options Factoring and FinalSubstitutions.
Solve2 uses the "high school algorithm" and factors intermediate results. Therefore it can be drastically more useful than Solve.

```
Options[Solve2]
```

{Factoring → Factor2, FinalSubstitutions → {}}

▪ Examples

```
Solve2[{2 x == b - w / 2, y - d == p}, {x, y}]
```

$\left\{x \to \frac{1}{4}(2\,b - w),\ y \to d + p\right\}$

If no equation sign is given the polynomials are supposed to be 0.

```
Solve2[x + y, x]
```

$\{x \to -y\}$

```
Solve2[x + y, x, FinalSubstitutions → {y → h}]
```

$\{x \to -h\}$

```
Solve2[{2 x == b - w / 2, y - d == p}, {x, y}, Factoring → Expand]
```

$$\left\{x \to \frac{b}{2} - \frac{w}{4}, y \to d + p\right\}$$

```
Solve[{2 x == b - w / 2, y - d == p}, {x, y}]
```

$$\left\{\left\{x \to \frac{1}{4}(2\,b - w), y \to d + p\right\}\right\}$$

■ SO

■ Description

SO[q] is a four–dimensional scalar product of OPEDelta with q.
It is transformed into Pair[Momentum[q], Momentum[OPEDelta]] by FCI.

■ See also: FCI, OPEDelta, Pair, ScalarProduct, SOD.

■ Examples

```
SO[p]
```

$$\Delta \cdot p$$

```
SO[p - q]
```

$$\Delta \cdot (p - q)$$

```
SO[p] // FCI // InputForm
```

```
Pair[Momentum[OPEDelta], Momentum[p]]
```

■ SPD

■ Description

SPD[a, b] denotes a D–dimensional scalar product SPD[a, b] is transformed into ScalarProduct[a, b, Dimension → D] by FeynCalcInternal. SPD[p] is the same as SPD[p,p] (= p^2).

■ **See also: Calc, ExpandScalarProduct, ScalarProduct.**

■ Examples

```
SPD[p, q] + SPD[q]
```

$$p\cdot q+q^2$$

```
SPD[p - q, q + 2 p]
```

$$(p-q)\cdot(2\,p+q)$$

```
Calc[ SPD[p - q, q + 2 p] ]
```

$$2\,p^2-p\cdot q-q^2$$

```
ExpandScalarProduct[SPD[p - q]]
```

$$p^2-2\,p\cdot q+q^2$$

```
SPD[a, b] // InputForm
```

```
SPD[a, b]
```

```
SPD[a, b] // FCI // InputForm
```

```
Pair[Momentum[a, D], Momentum[b, D]]
```

```
SPD[a, b] // FCI // FCE // InputForm
```

```
SPD[a, b]
```

```
FCE[ChangeDimension[SP[p, q], D]] // InputForm
```

```
SPD[p, q]
```

■ Spinor

■ Description

Spinor[p, m, optarg] is the head of Dirac spinors.
Which of the spinors u, v, $\bar{u}$, or $\bar{v}$ is understood, depends on the sign of the momentum (p) argument and the relative position of DiracSlash[p] Spinor[sign p, mass] is that spinor which yields: sign*mass*Spinor[p, mass] if the Dirac equation is applied (by DiracEquation or DiracSimplify).

■ See also: DiracSimplify, SpinorU, SpinorV, SpinorUBar, SpinorVBar.

■ Examples

```
Spinor[p]
```

$\varphi(p)$

```
Spinor[p, m]
```

$\varphi(p, m)$

```
DiracSimplify[Spinor[-p, m].DiracSlash[p]]
```

$-m\varphi(-p, m)$

```
Spinor[p] // InputForm
```

```
Spinor[Momentum[p], 0, 1]
```

```
ChangeDimension[Spinor[p], D] // InputForm
```

```
Spinor[Momentum[p], 0, 1]
```

```
Spinor[p, m] // InputForm
```

```
Spinor[Momentum[p], m, 1]
```

```
Spinor[p, SmallVariable[m]] // InputForm
```

```
Spinor[Momentum[p], 0, 1]
```

■ SpinorUBar

■ Description

SpinorUBar[p,m] denotes a $\bar{u}$ –spinor.

■ See also: Spinor, SpinorU, SpinorV, SpinorVBar.

■ Examples

One argument only assumes a massless spinor.

```
SpinorUBar[p]
```

$\bar{u}(p)$

```
SpinorUBar[p, m]
```

$\bar{u}(p, m)$

```
SpinorUBar[p, m] // InputForm
```

```
SpinorUBar[p, m]
```

```
SpinorUBar[p, m] // FCI // InputForm
```

```
Spinor[Momentum[p], m, 1]
```

```
SpinorUBar[p, m] // FCI // FCE // InputForm
```

```
Spinor[Momentum[p], m, 1]
```

■ SpinorU

■ Description

SpinorU[p, m, optarg] denotes a u–spinor.

■ See also: Spinor, SpinorUBar, SpinorV, SpinorVBar.

■ Examples

```
SpinorU[p]
```

$u(p)$

```
SpinorU[p, m]
```

$u(p, m)$

```
SpinorU[p, m] // InputForm
```

```
SpinorU[p, m]
```

```
SpinorU[p, m] // FCI // InputForm
```

```
Spinor[Momentum[p], m, 1]
```

```
SpinorU[p, m] // FCI // FCE // InputForm
```

```
Spinor[Momentum[p], m, 1]
```

■ SpinorVBar

■ Description

SpinorVBar[p,m] denotes a $\bar{v}$(p,m)–spinor.

■ See also: Spinor, SpinorU, SpinorV, SpinorUBar.

■ Examples

```
SpinorVBar[p]
```

$\bar{v}(p)$

```
SpinorVBar[p, m]
```

$\bar{v}(p, m)$

```
SpinorVBar[p, m] // InputForm
```

```
SpinorVBar[p, m]
```

```
SpinorVBar[p, m] // FCI // InputForm
```

```
Spinor[-Momentum[p], m, 1]
```

```
SpinorVBar[p, m] // FCI // FCE // InputForm
```

```
Spinor[-Momentum[p], m, 1]
```

■ SpinorV

■ Description

SpinorV[p,m, optarg] denotes a v–spinor.

■ See also: Spinor, SpinorUBar, SpinorU, SpinorVBar.

■ Examples

```
SpinorV[p]
```

$v(p)$

```
SpinorV[p, m]
```

$u(p, m)$

```
SpinorV[p, m] // InputForm
```

```
SpinorV[p, m]
```

```
SpinorV[p, m] // FCI // InputForm
```

```
Spinor[-Momentum[p], m, 1]
```

```
SpinorV[p, m] // FCI // FCE // InputForm
```

```
Spinor[-Momentum[p], m, 1]
```

■ SplittingFunction

■ Description

SplittingFunction[pxij] is a database of splitting functions in the $\overline{\text{MS}}$ scheme.

```
Options[SplittingFunction]
```

{Polarization → 1}

■ See also: AnomalousDimension.

■ Examples

Unpolarized case:

```
SetOptions[SplittingFunction, Polarization → 0];
```

```
SplittingFunction[Pqq]
```

$$C_F \left(-4\,x+6\,\delta(1-x)+8\left(\frac{1}{1-x}\right)_+ -4\right)$$

```
SplittingFunction[Pqg]
```

$$T_f\,(16\,x^2-16\,x+8)$$

```
SplittingFunction[Pgq]
```

$$C_F \left(4\,x-8+\frac{8}{x}\right)$$

```
SplittingFunction[Pgg]
```

$$8\,C_A\left(-x^2+x+\frac{11}{12}\,\delta(1-x)+\left(\frac{1}{1-x}\right)_+-2+\frac{1}{x}\right)-\frac{8}{3}\,N_f\,T_f\,\delta(1-x)$$

```
SplittingFunction[aqq]
```

$$C_F\left(2\,x+(7-4\,\zeta(2))\,\delta(1-x)+(2\,x+2)\log((1-x)\,x)-4\left(\frac{\log(x)}{1-x}+\left(\frac{\log(1-x)}{1-x}\right)_+\right)-4\right)$$

```
SplittingFunction[agq]
```

$$C_F\left(-2\,x+\left(-2\,x+4-\frac{4}{x}\right)\log((1-x)\,x)+2-\frac{4}{x}\right)$$

```
SplittingFunction[aqg]
```

$$T_f\,((-8\,x^2+8\,x-4)\log((1-x)\,x)-4)$$

```
SplittingFunction[agg]
```

Polarizedcase:

```
SetOptions[SplittingFunction, Polarization → 1];
```

```
SplittingFunction[Pqq]
```

$$C_F\left(-4\,x+6\,\delta(1-x)+8\left(\frac{1}{1-x}\right)_+-4\right)$$

```
SplittingFunction[Pqg]
```

$$T_f\,(16\,x-8)$$

```
SplittingFunction[Pgq]
```

$$C_F\,(8-4\,x)$$

```
SplittingFunction[Pgg]
```

$$C_A\left(-16\,x+\frac{22}{3}\,\delta(1-x)+8\left(\frac{1}{1-x}\right)_++8\right)-\frac{8}{3}\,N_f\,T_f\,\delta(1-x)$$

```
SplittingFunction[aqq]
```

$$C_F\left(8\,(1-x)+2\,x+(7-4\,\zeta(2))\,\delta(1-x)+(2\,x+2)\log((1-x)\,x)-4\log(x)\left(\frac{1}{1-x}\right)_+-4\left(\frac{\log(1-x)}{1-x}\right)_+-4\right)$$

```
SplittingFunction[agq]
```

$$C_F\,(-4\,x+(2\,x-4)\log((1-x)\,x)+2)$$

```
SplittingFunction[agqd]
```

$$C_F\,((2\,x-4)\log((1-x)\,x)-2)$$

SplittingFunction[aqg]

$$T_f\,((4-8\,x)\log((1-x)\,x)-4)$$

SplittingFunction[aqgd]

$$T_f\,((4-8\,x)\log((1-x)\,x)-4)$$

SplittingFunction[agg]

$$C_A\left(\left(\frac{67}{9}-4\,\zeta(2)\right)\delta(1-x)+(8\,x-4)\log((1-x)\,x)-4\left(\frac{\log(x)}{1-x}+\left(\frac{\log(1-x)}{1-x}\right)_+\right)+2\right)-\frac{20}{9}\,T_f\,\delta(1-x)$$

SplittingFunction[aggd]

$$C_A\left(\left(\frac{67}{9}-4\,\zeta(2)\right)\delta(1-x)+(8\,x-4)\log((1-x)\,x)-4\left(\frac{\log(x)}{1-x}+\left(\frac{\log(1-x)}{1-x}\right)_+\right)+2\right)-\frac{20}{9}\,T_f\,\delta(1-x)$$

SplittingFunction[PQQS]

$$C_F\,T_f\,(-16\,(x+1)\log^2(x)+(48\,x-16)\log(x)+16\,(1-x))$$

SplittingFunction[PQQNS]

$$\left(-4\,(x+1)\log^2(x)-8\left(2\,x+\frac{3}{1-x}\right)\log(x)-\frac{16\,(x^2+1)\log(1-x)\log(x)}{1-x}-40\,(1-x)+\delta(1-x)\,(-24\,\zeta(2)+48\,\zeta(3)+3)\right)C_F^2+$$
$$N_f\left(\frac{88\,x}{9}+\left(-\frac{16\,\zeta(2)}{3}-\frac{2}{3}\right)\delta(1-x)-\frac{8\,(x^2+1)\log(x)}{3\,(1-x)}-\frac{80}{9}\left(\frac{1}{1-x}\right)_+-\frac{8}{9}\right)C_F-$$
$$8\left(C_F-\frac{C_A}{2}\right)\left(4\,(1-x)+2\,(x+1)\log(x)+\frac{(x^2+1)\,(\log^2(x)-4\log(x+1)\log(x)-2\,\zeta(2)-4\,\mathrm{Li}_2(-x))}{x+1}\right)C_F+$$
$$C_A\left(\frac{4\,(x^2+1)\log^2(x)}{1-x}-\frac{4}{3}\left(5\,x-\frac{22}{1-x}+5\right)\log(x)+\frac{4}{9}\,(53-187\,x)+\right.$$
$$\left.8\,(x+1)\,\zeta(2)+\left(\frac{536}{9}-16\,\zeta(2)\right)\left(\frac{1}{1-x}\right)_+\!+\delta(1-x)\left(\frac{88\,\zeta(2)}{3}-24\,\zeta(3)+\frac{17}{3}\right)\right)C_F$$

SplittingFunction[PQG]

$$4\,C_F\,T_f\,((8\,x-4)\log^2(1-x)+(16-16\,x)\log(1-x)+$$
$$(8-16\,x)\log(x)\log(1-x)+(4\,x-2)\log^2(x)+54\,x-16\,x\zeta(2)+8\,\zeta(2)-18\log(x)-44)+$$
$$4\,C_A\,T_f\,((4-8\,x)\log^2(1-x)+(16\,x-16)\log(1-x)+(-8\,x-4)\log^2(x)-44\,x-8\,\zeta(2)+$$
$$(32\,x+4)\log(x)+(-16\,x-8)\log(x)\log(x+1)+(-16\,x-8)\,\mathrm{Li}_2(-x)+48)$$

SplittingFunction[PGQ]

$$(C_A\,C_F)\left((16-8\,x)\log^2(1-x)+\left(\frac{8\,x}{3}+\frac{80}{3}\right)\log(1-x)+(16\,x-32)\log(x)\log(1-x)+(8\,x+16)\log^2(x)+\right.$$
$$\left.\frac{280\,x}{9}+16\,x\zeta(2)+(32-104\,x)\log(x)+(16\,x+32)\log(x)\log(x+1)+(16\,x+32)\,\mathrm{Li}_2(-x)+\frac{328}{9}\right)+$$
$$C_F^2\,((8\,x-16)\log^2(1-x)+(-8\,x-16)\log(1-x)+(8-4\,x)\log^2(x)+32\,x-(32\,x+64)\log(x)+(36\,x+48)\log(x)-68)+$$
$$(C_F\,T_f)\left(-\frac{32\,x}{9}+\left(\frac{32\,x}{3}-\frac{64}{3}\right)\log(1-x)-\frac{128}{9}\right)$$

```
SplittingFunction[PGG]
```

$$\left(\left(-\frac{8}{x+1}+32+\frac{8}{1-x}\right)\log^2(x)+\left(\frac{232}{3}-\frac{536\,x}{3}\right)\log(x)+\left(64\,x-\frac{32}{1-x}-32\right)\log(1-x)\log(x)+\left(64\,x+\frac{32}{x+1}+32\right)\log(x+1)\log(x)-\frac{388\,x}{9}+\frac{64}{3}\,\delta(1-x)+\zeta(2)\left(64\,x-16\left(\frac{1}{1-x}\right)_+ +\frac{16}{x+1}\right)+\frac{536}{9}\left(\frac{1}{1-x}\right)_+ +\left(64\,x+\frac{32}{x+1}+32\right)\mathrm{Li}_2(-x)+24\,\delta(1-x)\,\zeta(3)-\frac{148}{9}\right)C_A^2+T_f\left(\frac{608\,x}{9}-\frac{32}{3}\,\delta(1-x)+\left(-\frac{32\,x}{3}-\frac{32}{3}\right)\log(x)-\frac{160}{9}\left(\frac{1}{1-x}\right)_+ -\frac{448}{9}\right)C_A+C_F\,T_f\left((-16\,x-16)\log^2(x)+(16\,x-80)\log(x)+80\,x-8\,\delta(1-x)-80\right)$$

■ SP

■ Description

SP[a, b] denotes a four–dimensional scalar product. SP[a, b] is transformed into ScalarProduct[a, b] by FeynCalcInternal. SP[p] is the same as SP[p, p] (= p^2).

■ See also: Calc, ExpandScalarProduct, ScalarProduct.

■ Examples

```
SP[p, q] + SP[q]
```

$$p\cdot q+q^2$$

```
SP[p - q, q + 2 p]
```

$$(p-q)\cdot(2\,p+q)$$

```
Calc[ SP[p - q, q + 2 p] ]
```

$$2\,p^2-p\cdot q-q^2$$

```
ExpandScalarProduct[SP[p - q]]
```

$$p^2-2\,p\cdot q+q^2$$

```
SP[a, b] // InputForm
```

```
SP[a, b]
```

```
SP[a, b] // FCI // InputForm
```

```
Pair[Momentum[a], Momentum[b]]
```

```
SP[a, b] // FCI // FCE // InputForm
```

```
SP[a, b]
```

■ SumP

■ Description

SumP[k,m] is 2^(*k*− 1) $\sum_{i=1}^{2\,m}$(1 + (−1)^*i*) / *i*^*k*

■ See also: SumS, SumT.

■ Examples

```
SumP[1, m - 1]
```

$S_1^{'}(m-1)$

```
SumP[2, m - 1]
```

$S_2^{'}(m-1)$

```
SumP[1, m]
```

$S_1^{'}(m)$

```
SumP[1, 4]
```

$\frac{25}{12}$

```
Sum[(1 + (-1) ^ i) / i, {i, 8}]
```

$\frac{25}{12}$

```
Explicit[SumP[1, n / 2]]
```

$\frac{1}{2}(1-(-1)^n)\,S_1\left(\frac{n-1}{2}\right)+\frac{1}{2}(1+(-1)^n)\,S_1\left(\frac{n}{2}\right)$

```
% /. n → 8
```

$\frac{25}{12}$

■ SumS

■ Description

SumS[1,m] is the harmonic number $\sum_{i=1}^{m} i\text{^}(-1)$ ($= S_1(m)$).
SumS[1,1,m] is $\sum_{i=1}^{m} S_1(i)/i$.
SumS[k,l,m] is $\sum_{i=1}^{m} S_l(i)/i^k$.
SumS[r,n] represents Sum[Sign[r]^i/i^Abs[r], {i, 1, n}].
SumS[r,s, n] is Sum[Sign[r]^k/k^Abs[r] Sign[s]^j/j^Abs[s], {k, 1, n}, {j, 1, k}], etc.

```
Options[SumS]
```

{Reduce→ False}

■ See also: SumP, SumT.

■ Examples

```
SumS[1, m - 1]
```

$S_1(m-1)$

```
SumS[2, m - 1]
```

$S_2(m-1)$

```
SumS[1, m]
```

$S_1(m)$

```
SumS[1, m, Reduce → True]
```

$$S_1(m-1)+\frac{1}{m}$$

```
SumS[3, m + 2, Reduce → True]
```

$$S_3(m+1)+\frac{1}{(m+2)^3}$$

```
SetOptions[SumS, Reduce → True];
SumS[3, m + 2]
```

$$S_3(m-1)+\frac{1}{m^3}+\frac{1}{(m+1)^3}+\frac{1}{(m+2)^3}$$

```
SetOptions[SumS, Reduce → False];
```

```
SumS[1, 4]
```

$$\frac{25}{12}$$

$$\sum_{i=1}^{4} 1 / i$$

$$\frac{25}{12}$$

```
SumS[1, 2, m - 1]
```

$S_{12}(m-1)$

```
SumS[1, 1, 1, 11]
```

$$\frac{31276937512951}{4260000729600}$$

```
SumS[-1, 4]
```

$$-\frac{7}{12}$$

```
SumT[1, 4]
```

$$-\frac{7}{12}$$

■ SumT

■ Description

SumT[1,m] is the alternative harmonic number $\sum_{i=1}^{m} (-1)\text{^}i/i$
SumT[r,n] represents Sum[(−1)^i/i^r,{i,1,n}],
SumT[r,s,n] is Sum[1/k^r(−1)^j/j^s,{k, 1, n}, {j, 1, k}].

■ See also: SumP, SumS.

■ Examples

```
SumT[1, m - 1]
```

$\tilde{S}_1(m-1)$

```
SumT[2, m - 1]
```

$\tilde{S}_2(m-1)$

```
SumT[1, m]
```

$\tilde{S}_1(m)$

```
SumT[1, m, Reduce → True]
```

$\tilde{S}_1(m-1)+\frac{(-1)^m}{m}$

```
SumT[1, 4]
```

$-\frac{7}{12}$

```
Sum[(-1)^i / i, {i, 1, 4}]
```

$-\frac{7}{12}$

```
SumT[1, 2, m - 1]
```

$\tilde{S}_{12}(m-1)$

```
SumT[1, 2, 42]
```

$-\frac{389879586970550133604898642987036214296101521386839 27}{105121216607023784053160049644837610808791905280000 00}$

```
SumT[1, 4]
```

$-\frac{7}{12}$

```
SumS[-1, 4]
```

$-\frac{7}{12}$

```
Sum[(-1)^i / i, {i, 1, m-1}]
```

$\frac{1}{2}(-1)^m\left(\psi^{(0)}\left(\frac{m}{2}\right)-\psi^{(0)}\left(\frac{m+1}{2}\right)\right)-\frac{\log(4)}{2}$

```
SumT[1, 2, 12]
```

$-\frac{57561743656913}{21300003648000}$

```
SumS[1, -2, 42]
```

$-\frac{389879586970550133604898642987036214296101521386839 27}{105121216607023784053160049644837610808791905280000 00}$

```
Array[SumT, 6]
```

$$\left\{-1, -\frac{5}{8}, -\frac{179}{216}, -\frac{1207}{1728}, -\frac{170603}{216000}, -\frac{155903}{216000}\right\}$$

```
Array[SumS[-2, 1, #1] &, 6]
```

$$\left\{-1, -\frac{5}{8}, -\frac{179}{216}, -\frac{1207}{1728}, -\frac{170603}{216000}, -\frac{155903}{216000}\right\}$$

■ SUNDelta

■ Description

SUNDelta[a, b] is the Kronecker–delta for SU(N) with color indices a and b.

■ See also: SD, SUNF, SUNSimplify, Trick.

■ Examples

```
SUNDelta[a, b]
```

δ_{ab}

```
Trick[ SUNDelta[a, b] SUNDelta[b, c] ]
```

δ_{ac}

```
SUNDelta[SUNIndex[a], SUNIndex[b]]
```

δ_{ab}

```
SUNDelta[a, b] // InputForm
```

```
SUNDelta[a, b]
```

```
SUNDelta[a, b] // FCI // InputForm
```

```
SUNDelta[SUNIndex[a], SUNIndex[b]]
```

```
SUNDelta[a, b] // FCI // FCE // InputForm
```

```
SD[a, b]
```

```
SD[a, b] // FCI // InputForm
```

```
SUNDelta[SUNIndex[a], SUNIndex[b]]
```

■ SUND

■ Description

SUND[a,b,c] are the symmetric SU(N) d_{abc}.

■ See also: SUNDelta, SUNF, SUNSimplify.

■ Examples

```
SUND[a, b, c]
```

d_{abc}

```
tt = SUND[a, b, c, Explicit -> True]
```

$2\,\text{tr}(T_a\,T_b\,T_c) + 2\,\text{tr}(T_b\,T_a\,T_c)$

```
SUND[c, a, b]
```

d_{abc}

```
SUND[a, b, b]
```

0

```
SUNSimplify[SUND[a, b, c] SUND[a, b, c]]
```

$-2\,(4 - C_A^2)\,C_F$

```
SUNSimplify[SUND[a, b, c] SUND[a, b, c], SUNNToCACF -> False] // Factor2
```

$$\frac{(1 - N^2)\,(4 - N^2)}{N}$$

```
SUNSimplify[SUND[a, b, c] SUND[e, b, c], SUNNToCACF -> False] // Factor2
```

$$-\frac{(4 - N^2)\,\delta_{ae}}{N}$$

```
SUND[a, b, c] // InputForm
```

```
SUND[a, b, c]
```

```
SUND[a, b, c] // FCI // InputForm
```

```
SUND[SUNIndex[a], SUNIndex[b], SUNIndex[c]]
```

```
SUND[a, b, c] // FCI // FCE // InputForm
```

```
SUND[a, b, c]
```

■ SUNF

■ Description

SUNF[a,b,c] are the structure constants of SU(N). The arguments a,b,c should be of symbol type.

```
Options[SUNF]
```

{Explicit→ False}

■ See also: SUND, SUNDelta, SUNIndex, SUNSimplify, SUNT, Trick.

■ Examples

```
t1 = SUNF[a, b, c] x + SUNF[b, a, c] y
```

$x\, f_{abc} + y\, f_{bac}$

```
Calc[t1]
```

$x\, f_{abc} - y\, f_{abc}$

```
SUNSimplify[t1]
```

$(x-y)\, f_{abc}$

```
SUNF[a, a, b]
```

f_{aab}

```
SUNF[a, a, b] // Calc
```

0

This is a consequence of the usual choice for the normalization of the T_a generators in (see SUNT).

```
SUNF[a, b, c, Explicit → True]
```

$2\, i\, (\text{tr}(T_a\, T_c\, T_b) - \text{tr}(T_a\, T_b\, T_c))$

```
SUNSimplify[SUNF[a, b, c] SUNF[a, b, d]]
```

$C_A\, \delta_{c\,d}$

```
SUNSimplify[SUNF[a, b, c], Explicit → True]
```

$-2\, i\, (\text{tr}(T_a\, T_b\, T_c) - \text{tr}(T_b\, T_a\, T_c))$

```
SUNF[a, b, c] // InputForm
```

```
SUNF[a, b, c]
```

```
SUNF[a, b, c] // FCI // InputForm
```

```
SUNF[SUNIndex[a], SUNIndex[b], SUNIndex[c]]
```

```
SUNF[a, b, c] // FCI // FCE // InputForm
```

```
SUNF[a, b, c]
```

```
SUNF[b, a, c]
```

f_{bac}

```
SUNF[b, a, c] // FCI
```

$-f_{abc}$

■ SUNIndex

■ Description

SUNIndex is the head of SU(N) indices.

■ See also: SUNDelta, SUNF.

■ Examples

```
SUNIndex[i]
```

i

```
SUNDelta[i, j] // FCI // InputForm
```

```
SUNDelta[SUNIndex[i], SUNIndex[j]]
```

■ SUNSimplify

■ Description

SUNSimplify simplifies products of SUNT (and complex conjugated) matrices.
Basic renaming of dummy indices is done. If the option SUNTrace is set to False, then any SUNT–matrices are taken out of DiracTrace[...], otherwise a color–trace is taken (by SUNTrace) before taking the SUN–objects in front of DiracTrace[...].

```
Options[SUNSimplify]
```

```
{Expanding→ False, Explicit→ False, Factoring→ False, SUNFJacobi→ False, SUNNToCACF→ True, SUNTrace→ True}
```

■ Seealso: Trick.

■ Examples

```
t1 = SUNDelta[a, b] SUNDelta[b, c]
```

$\delta_{ab}\,\delta_{bc}$

```
SUNSimplify[t1]
```

δ_{ac}

```
t2 = SUNT[a].SUNT[a]
```

$T_a\,T_a$

```
SUNSimplify[t2]
```

C_F

```
SUNSimplify[t2, SUNNToCACF -> False]
```

$$\frac{N^2-1}{2\,N}$$

```
t3 = SUNF[a, r, s] SUNF[b, r, s]
```

$f_{ars}\,f_{brs}$

```
SUNSimplify[t3]
```

$C_A\,\delta_{ab}$

```
t4 = SUNF[a, b, c] . SUNF[a, b, c]
```

$f_{abc}\,f_{abc}$

```
SUNSimplify[t4]
```

$2\,C_A^2\,C_F$

```
t5 = SUNF[a, b, c] SUNF[d, b, c]
```

$f_{abc}\,f_{dbc}$

```
SUNSimplify[t5]
```

$C_A\,\delta_{ad}$

```
t6 = SUNF[a, b, c] SUND[d, b, c]
```

$d_{bcd}\,f_{abc}$

```
SUNSimplify[t6, Explicit -> True]
```

0

```
SUNSimplify[SUND[a, b, c] SUND[a, b, c], SUNNToCACF → False] // Factor2
```

$$\frac{(1-N^2)(4-N^2)}{N}$$

```
SUNSimplify[SUND[a, b, c] SUND[e, b, c], SUNNToCACF → False] // FullSimplify
```

$$\frac{(N^2-4)\,\delta_{ae}}{N}$$

```
SUNSimplify[SUNF[a, b, c], Explicit → True]
```

$$-2\,i\,(\text{tr}(T_a\,T_b\,T_c)-\text{tr}(T_b\,T_a\,T_c))$$

```
SUNSimplify[SUND[a, b, c], Explicit → True]
```

$$2\,(\text{tr}(T_a\,T_b\,T_c)+\text{tr}(T_b\,T_a\,T_c))$$

```
SUNSimplify[SUNF[a, b, c] SUNT[c, b, a]]
```

$$-\frac{1}{2}\,i\,C_A\,C_F$$

```
a1 = SUNF[a, b, e] SUNF[c, d, e] + SUNF[a, b, z] SUNF[c, d, z]
```

$$f_{abe}\,f_{cde}+f_{abz}\,f_{cdz}$$

```
SUNSimplify[a1, Explicit → False]
```

$$2\,f_{abe}\,f_{cde}$$

```
a2 = DiracTrace[SUNT[a, b].DiracMatrix[μ, ν]]
```

$$\text{tr}((T_a\,T_b)\,(\gamma^\mu\,\gamma^\nu))$$

```
SUNSimplify[a2]
```

$$\frac{1}{2}\,\text{tr}(\gamma^\mu\,\gamma^\nu)\,\delta_{ab}$$

```
SUNSimplify[a2, SUNTrace → False]
```

$$\text{tr}(\gamma^\mu\,\gamma^\nu)\,T_a\,T_b$$

```
SUNSimplify[a2, SUNTrace → False] // InputForm
```

```
DiracTrace[DiracGamma[LorentzIndex[μ]] .
    DiracGamma[LorentzIndex[ν]]] . SUNT[SUNIndex[a]] .
  SUNT[SUNIndex[b]]
```

■ SUNT

■ Description

SUNT[a] is the SU(N) T_a generator in the fundamental representation.

■ See also: CA, CF, SUND, SUNDelta, SUNF, SUNSimplify.

■ Examples

```
SUNT[a]
```

T_a

Since T_a is a noncommutative object, products have to separated by a dot (.).

```
SUNT[a] . SUNT[b] . SUNT[c]
```

$T_a\ T_b\ T_c$

```
SUNT[a, b, c, d]
```

$T_a\ T_b\ T_c\ T_d$

```
SUNSimplify[SUNT[a, b, a], SUNNToCACF → False]
```

$$-\frac{T_b}{2\,N}$$

```
SUNSimplify[SUNT[a, b, b, a]]
```

C_F^2

```
SUNSimplify[SUNT[a, b, a]]
```

$$-\frac{1}{2}\,(C_A - 2\,C_F)\,T_b$$

```
SUNSimplify[SUNT[a, b, a], SUNNToCACF → False]
```

$$-\frac{T_b}{2\,N}$$

The normalization of the generators is chosen in the standard way, therefore $\mathrm{tr}(T_a\ T_b) = 1/2\,\delta_{\mathrm{ab}}$.

```
SUNTrace[SUNT[a, b]]
```

$$\frac{\delta_{ab}}{2}$$

In case you want T_f, you need to include a factor 2Tf inside the trace.

```
SUNTrace[2 Tf SUNT[a, b]]
```

$T_f \delta_{ab}$

```
SUNTrace[SUNT[a, b]] // InputForm
```

```
SUNDelta[SUNIndex[a], SUNIndex[b]]/2
```

```
SUNT[a] // FCI // InputForm
```

```
SUNT[SUNIndex[a]]
```

```
SUNT[a] // FCI // FCE // InputForm
```

```
SUNT[a]
```

■ SUNTrace

■ Description

SUNTrace[expr] calculates the color–trace.

```
Options[SUNTrace]
```

{Explicit→ False}

■ See also: SUNSimplify, Tr.

■ Examples

```
SUNTrace[SUNT[a, b]]
```

$$\frac{\delta_{ab}}{2}$$

```
SUNTrace[SUNT[a, b, c]]
```

$\mathrm{tr}(T_a\, T_b\, T_c)$

```
SUNTrace[SUNT[a, b, c], Explicit → True]
```

$$\frac{d_{abc}}{4} + \frac{1}{4}\, i\, f_{abc}$$

```
SUNTrace[SUNT[a, b, c, d]]
```

$\mathrm{tr}(T_a\, T_b\, T_c\, T_d)$

```
t1 = SUNTrace[SUNT[a, b, c, d], Explicit → True]
```

$$\frac{1}{8} d_{ead}\, d_{ebc} - \frac{1}{8} i\, f_{ade}\, d_{ebc} - \frac{1}{8} d_{eac}\, d_{ebd} + \frac{1}{8} d_{eab}\, d_{ecd} + \frac{\delta_{ad}\,\delta_{bc}}{4\,N} - \frac{\delta_{ac}\,\delta_{bd}}{4\,N} + \frac{\delta_{ab}\,\delta_{cd}}{4\,N} + \frac{1}{8} i\, d_{ead}\, f_{bce}$$

```
SUNSimplify[t1, Explicit → True]
```

$\mathrm{tr}(T_a\, T_b\, T_c\, T_d)$

```
t2 = SUNTrace[SUNT[a, b, c, d, e], Explicit → True]
```

$$\frac{\delta_{ab}\left(\frac{d_{cde}}{4} + \frac{1}{4} i\, f_{cde}\right)}{2\,N} +$$
$$\frac{1}{2} d_{\text{c105}ab}\left(-\frac{1}{8} d_{\text{c106}ce}\, d_{\text{c106c105}d} + \frac{1}{8} d_{\text{c106}cd}\, d_{\text{c106c105}e} + \frac{1}{8} d_{\text{c106}c\text{c105}}\, d_{\text{c106}de} - \frac{\delta_{ce}\,\delta_{\text{c105}d}}{4\,N} + \frac{\delta_{cd}\,\delta_{\text{c105}e}}{4\,N} + \right.$$
$$\left. \frac{\delta_{c\text{c105}}\,\delta_{de}}{4\,N} + \frac{1}{8} i\, d_{\text{c106c105}e}\, f_{cd\text{c106}} - \frac{1}{8} i\, d_{\text{c106}cd}\, f_{\text{c105ec106}}\right) +$$
$$\frac{1}{2} i\, f_{ab\text{c105}}\left(-\frac{1}{8} d_{\text{c107}ce}\, d_{\text{c107c105}d} + \frac{1}{8} d_{\text{c107}cd}\, d_{\text{c107c105}e} + \frac{1}{8} d_{\text{c107}c\text{c105}}\, d_{\text{c107}de} - \frac{\delta_{ce}\,\delta_{\text{c105}d}}{4\,N} + \right.$$
$$\left. \frac{\delta_{cd}\,\delta_{\text{c105}e}}{4\,N} + \frac{\delta_{c\text{c105}}\,\delta_{de}}{4\,N} + \frac{1}{8} i\, d_{\text{c107c105}e}\, f_{cd\text{c107}} - \frac{1}{8} i\, d_{\text{c107}cd}\, f_{\text{c105ec107}}\right)$$

```
SUNSimplify[t2, Explicit → True]
```

$\mathrm{tr}(T_a\, T_b\, T_c\, T_d\, T_e)$

```
SUNSimplify[SUNF[a, b, c] SUND[d, b, c]]
```

0

```
SUNSimplify[SUNF[a, b, c] SUND[a, b, d]]
```

0

```
SUNSimplify[SUNF[a, b, c] SUND[a, d, c]]
```

0

```
SUNSimplify[SUND[a, b, c] SUND[d, b, c]]
```

$-(4 - C_A^2)\,(C_A - 2\,C_F)\,\delta_{ad}$

■ Symmetrize

■ Description

Symmetrize[expr, {a1, a2, ...}] symmetrizes expr w.r.t. to the variables a2,a2....

■ Examples

```
Symmetrize[f[a, b], {a, b}]
```

$$\frac{1}{2}(f(a, b) + f(b, a))$$

```
Symmetrize[f[a, b, c, d], {a, b, c}]
```

$$\frac{1}{6}(f(a, b, c, d) + f(a, c, b, d) + f(b, a, c, d) + f(b, c, a, d) + f(c, a, b, d) + f(c, b, a, d))$$

■ Tdec

■ Description

Tdec[{q,mu},{p}]; Tdec[{{qi, mu}, {qj, nu}, ...}, {p1, p2, ...}] or Tdec[exp,{{qi, mu}, {qj, nu}, ...}, {p1, p2, ...}] calculates the tensorial decomposition formulas. The more common ones are saved in TIDL.

```
Options[Tdec]
```

{Dimension → *D*, Factoring → Factor2, FeynCalcExternal → True, List → True, NumberOfMetricTensors → ∞}

■ See also: TID, TIDL, OneLoopSimplify.

■ Examples

Check that $\int d\text{^}D\, f(p, q)\, q\text{^}\mu = p\text{^}\mu / p\text{^}2 \int d\text{^}D\, f(p, q)\, p\cdot q$

```
Tdec[{q, μ}, {p}]
```

$$\left\{\{\text{X1} \to p^2, \text{X2} \to p\cdot q\}, \frac{\text{X2}\ p^\mu}{\text{X1}}\right\}$$

```
%[[2]] /. %[[1]]
```

$$\frac{p^\mu\ p\cdot q}{p^2}$$

This calculates integral transformation for any $\int d^D q_1\, d^D q_2\, d^D q_3\ \ f(p, q_1, q_2, q_3)\, q_1^\mu\, q_2^\nu\, q_3^\rho$.

```
t = Tdec[{{q1, μ}, {q2, ν}, {q3, ρ}}, {p}];
```

```
t[[2]] /. t[[1]]
```

$$\frac{(1-D^2)\,p^\rho\,g^{\mu\nu}\,p\cdot q_3\,(p\cdot q_1\;p\cdot q_2-p^2\,q_1\cdot q_2)}{(1-D)^2\,(D+1)\,p^2{}^2}+\frac{(1-D^2)\,p^\nu\,g^{\mu\rho}\,p\cdot q_2\,(p\cdot q_1\;p\cdot q_3-p^2\,q_1\cdot q_3)}{(1-D)^2\,(D+1)\,p^2{}^2}+$$
$$\frac{(1-D^2)\,p^\mu\,g^{\nu\rho}\,p\cdot q_1\,(p\cdot q_2\;p\cdot q_3-p^2\,q_2\cdot q_3)}{(1-D)^2\,(D+1)\,p^2{}^2}-\frac{1}{(1-D)^2\,(D+1)\,p^2{}^3}$$
$$((1-D^2)\,p^\mu\,p^\nu\,p^\rho\,(D\,p\cdot q_1\;p\cdot q_2\;p\cdot q_3+2\,p\cdot q_1\;p\cdot q_2\;p\cdot q_3-p^2\,q_1\cdot q_2\;p\cdot q_3-p^2\,p\cdot q_2\;q_1\cdot q_3-p^2\,p\cdot q_1\;q_2\cdot q_3))$$

```
Contract[% FVD[p, μ] FVD[p, ν] FVD[p, ρ]] // Factor
```

$$p\cdot q_1\;p\cdot q_2\;p\cdot q_3$$

```
Clear[t];
```

■ Tensorfunction

■ Description

Tensorfunction[t,mu,nu,...] transformsintot[LorentzIndex[mu],LorentzIndex[nu],...], i.e., it canbeusedas an unspecifiedtensorielfunctiont.
A symmetrictensorcan be obtainedby Tensorfunction[{t,"S"}, mu, nu, ...], and an antisymmtericoneby Tensorfunction[{t,"A"}, mu,nu,...].

■ Seealso: Symmetrize.

■ Examples

```
Tensorfunction[t, μ, ν, τ]
```

$$t(\mu,\nu,\tau)$$

```
% // FullForm
```

t[LorentzIndex[Mu], LorentzIndex[Nu], LorentzIndex[Tau]]

```
Contract[FV[p, μ] %]
```

$$t(p,\nu,\tau)$$

```
% // InputForm
```

```
t[Momentum[p], LorentzIndex[ν], LorentzIndex[τ]]
```

```
Tensorfunction[{f, "S"}, α, β]
```

$$f(\alpha,\beta)$$

```
Tensorfunction[{f, "S"}, β, α]
```

$f(\alpha, \beta)$

```
% // InputForm
```

```
f[LorentzIndex[α], LorentzIndex[β]]
```

```
?? f
```

```
Global`f

Attributes[f] = {Orderless}
```

```
ClearAttributes[f, Orderless]
```

■ TFi

■ Description

TFi[d, pp, {{n1, m1}, {n2, m2}, {n3, m3}, {n4, m4}, {n5, m5}}] is the loop d-dimensional integral 1 / ((q1^2 - m1^2)^n1 (q2^2 - m2^2)^n2 ((q1 - p)^2 - m3^2)^n3 ((q2 - p)^2 - m4^2)^n4 ((q1 - q2)^2 - m5^2)^n5) .

TFi[d, pp, {x, y, z, v, w}, {{n1, m1}, {n2, m2}, {n3, m3}, {n4, m4}, {n5, m5}}] has as additional factors in the numerator (q1^2)^x * (q2^2)^y * (q1.p)^z * (q2.p)^v * (q1.q2)^w.

TFi[d, pp, dp, {a, b}, {{n1, m1}, {n2, m2}, {n3, m3}, {n4, m4}, {n5, m5}}] has as additional factors in the numerator (OPEDelta.q1)^a (OPEDelta.q2)^b, dp is (OPEDelta.p).

TFi is similar to TFI from the TARCER package, see hep-ph/9801383.

The function TarcerRecurse from the TARCER package recognize TFi (as well as TFI, which is defined in the HighEnergyPhysics`Tarcer` context).

■ See also: ToTFi, FromTFi.

■ Examples

```
TFi[D, M^2, {{2, m1}, {1, m2}, {3, m3}, {1, m4}, {1, m5}}]
```

$$\mathbf{F}^{(D)}_{\{2,m1\}\{1,m2\}\{3,m3\}\{1,m4\}\{1,m5\}}$$

```
TFi[D, M^2, {{2, M}, 1, 1, 1, 1}]
```

$$\mathbf{F}^{(D)}_{\{2,M\}\{1,0\}\{1,0\}\{1,0\}\{1,0\}}$$

■ TIDL

■ Description

TIDL is a database of tensorial reduction formalae.

■ See also: TID

■ Examples

■ B_μ – type

In any n–dimensional integral $\int d^n q^\mu \; f(q, p)$ the q^μ can be replaced by

```
TIDL[{q, μ}, {p}]
```

$$\frac{p^\mu \; p\cdot q}{p^2}$$

■ $B_{\mu\nu}$ – type

```
TIDL[{{q, μ}, {q, ν}}, {p}]
```

$$\frac{g^{\mu\,\nu} \; ((p\cdot q)^2 - p^2 \; q^2)}{(1 - D) \; p^2} - \frac{p^\mu \; p^\nu \; (D (p\cdot q)^2 - p^2 \; q^2)}{(1 - D) \; p^4}$$

■ C_μ – type

```
TIDL[{q, μ}, {p1, p2}]
```

$$\frac{\frac{1}{2} \; s \; p_2^\mu \; q\cdot p_1 - p_2^\mu \; q\cdot p_2 \; {p_1}^2}{\frac{s^2}{4} - {p_1}^2 \; {p_2}^2} + \frac{\frac{1}{2} \; s \; p_1^\mu \; q\cdot p_2 - p_1^\mu \; q\cdot p_1 \; {p_2}^2}{\frac{s^2}{4} - {p_1}^2 \; {p_2}^2}$$

■ $C_{\mu\nu}$ – type

```
TIDL[{{q, μ}, {q, ν}}, {p1, p2}]
```

$$\frac{g^{\mu\nu}\left(\frac{q^2 s^2}{4} - q\cdot p_1\, q\cdot p_2\, s + (q\cdot p_2)^2\, p_1^2 + (q\cdot p_1)^2\, p_2^2 - q^2\, p_1^2\, p_2^2\right)}{(D-2)\left(\frac{s^2}{4} - p_1^2\, p_2^2\right)} +$$
$$\left(p_2^\mu\, p_1^\nu\left(-\frac{1}{8}q^2 s^3 + \frac{1}{4}D q\cdot p_1\, q\cdot p_2\, s^2 - \frac{1}{2}D(q\cdot p_2)^2\, p_1^2\, s + \frac{1}{2}(q\cdot p_2)^2\, p_1^2\, s - \frac{1}{2}D(q\cdot p_1)^2\, p_2^2\, s + \frac{1}{2}(q\cdot p_1)^2\, p_2^2\, s + \right.\right.$$
$$\left.\left.\frac{1}{2}q^2\, p_1^2\, p_2^2\, s + D q\cdot p_1\, q\cdot p_2\, p_1^2\, p_2^2 - 2\, q\cdot p_1\, q\cdot p_2\, p_1^2\, p_2^2\right)\right)\Big/\left((D-2)\left(\frac{s^2}{4} - p_1^2\, p_2^2\right)^2\right) +$$
$$\left(p_1^\mu\, p_2^\nu\left(-\frac{1}{8}q^2 s^3 + \frac{1}{4}D q\cdot p_1\, q\cdot p_2\, s^2 - \frac{1}{2}D(q\cdot p_2)^2\, p_1^2\, s + \frac{1}{2}(q\cdot p_2)^2\, p_1^2\, s - \frac{1}{2}D(q\cdot p_1)^2\, p_2^2\, s + \right.\right.$$
$$\left.\left.\frac{1}{2}(q\cdot p_1)^2\, p_2^2\, s + \frac{1}{2}q^2\, p_1^2\, p_2^2\, s + D q\cdot p_1\, q\cdot p_2\, p_1^2\, p_2^2 - 2\, q\cdot p_1\, q\cdot p_2\, p_1^2\, p_2^2\right)\right)\Big/\left((D-2)\left(\frac{s^2}{4} - p_1^2\, p_2^2\right)^2\right) +$$
$$\left(p_2^\mu\, p_2^\nu\left(\frac{1}{4}D(q\cdot p_1)^2\, s^2 - \frac{1}{2}(q\cdot p_1)^2\, s^2 + \frac{1}{4}q^2\, p_1^2\, s^2 - D q\cdot p_1\, q\cdot p_2\, p_1^2\, s + q\cdot p_1\, q\cdot p_2\, p_1^2\, s + \right.\right.$$
$$\left.\left.D(q\cdot p_2)^2\, p_1^4 - (q\cdot p_2)^2\, p_1^4 - q^2\, p_1^4\, p_2^2 + (q\cdot p_1)^2\, p_1^2\, p_2^2\right)\right)\Big/\left((D-2)\left(\frac{s^2}{4} - p_1^2\, p_2^2\right)^2\right) +$$
$$\left(p_1^\mu\, p_1^\nu\left(\frac{1}{4}D(q\cdot p_2)^2\, s^2 - \frac{1}{2}(q\cdot p_2)^2\, s^2 + \frac{1}{4}q^2\, p_2^2\, s^2 - D q\cdot p_1\, q\cdot p_2\, p_2^2\, s + q\cdot p_1\, q\cdot p_2\, p_2^2\, s + \right.\right.$$
$$\left.\left.D(q\cdot p_1)^2\, p_2^4 - (q\cdot p_1)^2\, p_2^4 - q^2\, p_1^2\, p_2^4 + (q\cdot p_2)^2\, p_1^2\, p_2^2\right)\right)\Big/\left((D-2)\left(\frac{s^2}{4} - p_1^2\, p_2^2\right)^2\right)$$

■ $C_{\mu\nu\rho}$ – type

```
TIDL[{{q, μ}, {q, ν}, {q, ρ}}, {p, k}];
```

■ $C_{\mu\nu\rho\sigma}$ – type

```
TIDL[{{q, μ}, {q, ν}, {q, ρ}, {q, σ}}, {p, k}];
```

■ D_μ – type

```
TIDL[{q, μ}, {p1, p2, p3}]
```

$$\left(4\, p_3^{\mu}\left(\frac{1}{4}\, q\cdot p_3\, s^2-\frac{1}{2}\, q\cdot p_2\; p_1\cdot p_3\, s-\frac{1}{2}\, q\cdot p_1\; p_2\cdot p_3\, s-q\cdot p_3\; p_1^{\,2}\; p_2^{\,2}+q\cdot p_1\; p_1\cdot p_3\; p_2^{\,2}+q\cdot p_2\; p_1^{\,2}\; p_2\cdot p_3\right)\right)\Big/$$
$$\left(p_3^{\,2}\, s^2-4\, p_1\cdot p_3\; p_2\cdot p_3\, s+4\, p_1^{\,2}\,(p_2\cdot p_3)^2+4\,(p_1\cdot p_3)^2\; p_2^{\,2}-4\, p_1^{\,2}\; p_2^{\,2}\; p_3^{\,2}\right)-$$
$$\left(4\, p_2^{\mu}\left(-(q\cdot p_2)\,(p_1\cdot p_3)^2+\frac{1}{2}\, s\, q\cdot p_3\; p_1\cdot p_3+q\cdot p_1\; p_2\cdot p_3\; p_1\cdot p_3-q\cdot p_3\; p_1^{\,2}\; p_2\cdot p_3-\frac{1}{2}\, s\, q\cdot p_1\; p_3^{\,2}+q\cdot p_2\; p_1^{\,2}\; p_3^{\,2}\right)\right)\Big/$$
$$\left(p_3^{\,2}\, s^2-4\, p_1\cdot p_3\; p_2\cdot p_3\, s+4\, p_1^{\,2}\,(p_2\cdot p_3)^2+4\,(p_1\cdot p_3)^2\; p_2^{\,2}-4\, p_1^{\,2}\; p_2^{\,2}\; p_3^{\,2}\right)+$$
$$\left(4\, p_1^{\mu}\left(q\cdot p_1\,(p_2\cdot p_3)^2-\frac{1}{2}\, s\, q\cdot p_3\; p_2\cdot p_3-q\cdot p_2\; p_1\cdot p_3\; p_2\cdot p_3+q\cdot p_3\; p_1\cdot p_3\; p_2^{\,2}+\frac{1}{2}\, s\, q\cdot p_2\; p_3^{\,2}-q\cdot p_1\; p_2^{\,2}\; p_3^{\,2}\right)\right)\Big/$$
$$\left(p_3^{\,2}\, s^2-4\, p_1\cdot p_3\; p_2\cdot p_3\, s+4\, p_1^{\,2}\,(p_2\cdot p_3)^2+4\,(p_1\cdot p_3)^2\; p_2^{\,2}-4\, p_1^{\,2}\; p_2^{\,2}\; p_3^{\,2}\right)$$

■ $D_{\mu\nu}$ – type

```
TIDL[{{q, μ}, {q, ν}}, {p, k, r}];
```

■ $D_{\mu\nu\rho}$ – type

```
TIDL[{{q, μ}, {q, ν}, {q, ρ}}, {p, k, r}];
```

■ $D_{\mu\nu\rho\sigma}$ – type

```
TIDL[{{q, μ}, {q, ν}, {q, ρ}, {q, σ}}, {p, k, r, s}];
```

```
{Length[%], LeafCount[%], ByteCount[%]}
```

{4, 29, 320}

■ TID

■ Description

TID[amp,q] performs a tensorial decomposition.

```
Options[TID]
```

{Collecting → True, Contract → False, Dimension → D, ChangeDimension → D,
DimensionalReduction → False, FeynAmpDenominatorCombine → True,
FeynAmpDenominatorSimplify → False, Isolate → False, ScalarProductCancel → True}

■ See also: OneLoopSimplify, TIDL.

■ Examples

```
FAD[k, k - p1, k - p2] FVD[k, μ] // FCI
```

$$\frac{k^{\mu}}{k^2\,(k-p_1)^2\,(k-p_2)^2}$$

```
Factor/@TID[%, k] // FCE
```

$$-\frac{\frac{1}{([k^2])([(k-p_1)^2])}\,(s\,p_1^{\mu}-2\,p_2^{\mu}\,p_1^{2})}{s^2-4\,p_1^{2}\,p_2^{2}}+\frac{\frac{1}{([k^2])([(k-p_1)^2])([(k-p_2)^2])}\,(s\,p_2^{\mu}\,p_1^{2}-2\,p_1^{\mu}\,p_2^{2}\,p_1^{2}-2\,p_2^{\mu}\,p_2^{2}\,p_1^{2}+s\,p_1^{\mu}\,p_2^{2})}{s^2-4\,p_1^{2}\,p_2^{2}}-$$
$$\frac{\frac{1}{([k^2])([(k-p_2)^2])}\,(s\,p_2^{\mu}-2\,p_1^{\mu}\,p_2^{2})}{s^2-4\,p_1^{2}\,p_2^{2}}+\frac{\frac{1}{([(k-p_1)^2])([(k-p_2)^2])}\,(s\,p_1^{\mu}-2\,p_2^{2}\,p_1^{\mu}+s\,p_2^{\mu}-2\,p_2^{\mu}\,p_1^{2})}{s^2-4\,p_1^{2}\,p_2^{2}}$$

```
FAD[k, k - p] FVD[k, μ] // FCI
```

$$\frac{k^{\mu}}{k^2\,(k-p)^2}$$

```
TID[%, k]
```

$$\frac{p^{\mu}}{2\,(k^2\,(k-p)^2)}-\frac{p^{\mu}}{2\,k^2\,p^2}+\frac{p^{\mu}}{2\,(k-p)^2\,p^2}$$

```
TID[%%, k, FeynAmpDenominatorSimplify → True]
```

$$\frac{p^{\mu}}{2\,(k^2\,(k-p)^2)}$$

■ ToDistribution

■ Description

ToDistribution[exp, x] replaces (1-x)^(a Epsilon-1) in exp by 1/(a Epsilon) DeltaFunction[1-x] - 1/(1-x)+ a Epsilon Log[1-x]/(1-x) - 1/2 a^2 Epsilon^2 Log[1-x]^2/(1-x) and (1-x)^(a Epsilon-2) in exp by -1/(a Epsilon) DeltaFunctionPrime[1-x] - 1/(1-x)^2 - (a Epsilon) Log[1-x]/(1-x)^2 - a^2 Epsilon^2/2 Log[1-x]^2/(1-x)^2 - a^3 Epsilon^3/6 Log[1-x]^3/(1-x)^2.

```
Options[ToDistribution]
```

{PlusDistribution → Identity}

■ See also: PlusDistribution.

■ Examples

```
SetOptions[ToDistribution, PlusDistribution → PlusDistribution]
```

{PlusDistribution→ PlusDistribution}

```
ToDistribution[(1 - x) ^ (Epsilon - 1), x]
```

$$\frac{1}{6}\left(\frac{\log^3(1-x)}{1-x}\right)_+ \varepsilon^3 + \frac{1}{2}\left(\frac{\log^2(1-x)}{1-x}\right)_+ \varepsilon^2 + \left(\frac{\log(1-x)}{1-x}\right)_+ \varepsilon + \left(\frac{1}{1-x}\right)_+ + \frac{\delta(1-x)}{\varepsilon}$$

```
SetOptions[ToDistribution, PlusDistribution → Identity]
```

{PlusDistribution→ Identity}

```
ToDistribution[(1 - x) ^ (Epsilon - 2), x]
```

$$\frac{\varepsilon^3 \log^3(1-x)}{6(1-x)^2} + \frac{\varepsilon^2 \log^2(1-x)}{2(1-x)^2} + \frac{\varepsilon \log(1-x)}{(1-x)^2} - \frac{\delta'(1-x)}{\varepsilon} + \frac{1}{(1-x)^2}$$

```
Series2[Integrate[(1 - x)^Epsilon-2, {x, 0, 1}, GenerateConditions → False], Epsilon, 3]
```

$$-\varepsilon^3 - \varepsilon^2 - \varepsilon - 1$$

```
Options@Integrate
```

{Assumptions→ {}, GenerateConditions→ Automatic, PrincipalValue→ False}

```
Integrate2[ToDistribution[(1 - x) ^ (Epsilon - 2), x], {x, 0, 1}]
```

$$-\varepsilon^3 - \varepsilon^2 - \varepsilon - 1$$

■ ToHypergeometric

■ Description

ToHypergeometric[t^b(1 − t)^c(1 + t z)^a, t] returns
Γ(b+1) Γ(c+1)/Γ(b+c+2)$_2$ $F_1(-a, b+1; b+c+2; -z)$.
Remember that Re b >0 and Re (c−b)> 0 should hold (need not be set in Mathematica).

■ **See also: HypergeometricAC, HypergeometricIR, HypergeometricSE.**

■ Examples

```
ToHypergeometric[t^b (1 - t)^c (1 + t z)^a, t]
```

$$\frac{\Gamma(b+1)\,\Gamma(c+1)\;{}_2F_1(-a,\,b+1;\,b+c+2;\,-z)}{\Gamma(b+c+2)}$$

```
ToHypergeometric[w t^(b - 1) (1 - t)^(c - b - 1) (1 - t z)^(-a), t]
```

$$\frac{w\,\Gamma(b)\,\Gamma(c-b)\;{}_2F_1(a,\,b,\,c;\,z)}{\Gamma(c)}$$

```
ToHypergeometric[t^b (1 - t)^c (u + t z)^a, t]
```

$$\frac{u^a\,\Gamma(b+1)\,\Gamma(c+1)\;{}_2F_1(-a,\,b+1;\,b+c+2;\,-\frac{z}{u})}{\Gamma(b+c+2)}$$

```
ToHypergeometric[w t^(b - 1) (1 - t)^(c - b - 1) (u - t z)^(-a), t]
```

$$\frac{u^{-a}\,w\,\Gamma(b)\,\Gamma(c-b)\;{}_2F_1(a,\,b,\,c;\,\frac{z}{u})}{\Gamma(c)}$$

■ ToLarin

■ Description

ToLarin[exp] translates all $\gamma^\mu\ \gamma^5$ in exp into $-i/6\,\varepsilon^{\mu\nu\lambda\sigma}\ \gamma^\nu\ \gamma^\lambda\ \gamma^\sigma$.

■ **See also: Eps.**

■ Examples

```
DiracMatrix[ν].DiracMatrix[μ].DiracMatrix[5]
```

$$\gamma^\nu\ \gamma^\mu\ \gamma^5$$

```
ToLarin[%]
```

$$-\frac{1}{6}\,i\,(\gamma^\nu\ \gamma^{\text{du1}}\ \gamma^{\text{du2}}\ \gamma^{\text{du3}})\,\epsilon^{\mu\,\text{du1du2du3}}$$

■ ToTFi

■ Description

ToTFi[expr,q1, q2, p] translates FeynCalc 2–loop self energy type integrals into the TFi notatation, which can be used as input for the function TarcerRecurse from the TARCER package. See TFi for details on the conventions.

■ See also: TFi, FromTFi.

■ Examples

```
FAD[q1, q1 - p, {q2, M}, {q2 - p, m}, q1 - q2]
```

$$\frac{1}{([q_1^2])\,([(q_1-p)^2])\,([(q_1-q_2)^2])\,([q_2^2-M^2])\,([(q_2-p)^2-m^2])}$$

```
ToTFi[%, q1, q2, p]
```

$$\mathbf{F}^{(D)}_{\{1,0\}\{1,M\}\{1,0\}\{1,m\}\{1,0\}}$$

```
% // InputForm
```

```
TFi[D, SPD[p, p], {{1, 0}, {1, M}, {1, 0}, {1, m}, {1, 0}}]
```

```
SOD[q1] SOD[q2] FAD[q1, q1 - p, {q2, M}, {q2 - p, m}, q1 - q2] // FCI
```

$$\frac{\Delta\cdot q_1\,\Delta\cdot q_2}{q_1^2\,(q_1-p)^2\,(q_2^2-M^2)\,((q_2-p)^2-m^2)\,(q_1-q_2)^2}$$

```
ToTFi[%]
```

$$\mathbf{F}^{(D)\,1\,1}_{\{1,0\}\{1,M\}\{1,0\}\{1,m\}\{1,0\}}$$

```
% // InputForm
```

```
TFi[D, SPD[p, p], SOD[p], {1, 1},
  {{1, 0}, {1, M}, {1, 0}, {1, m}, {1, 0}}]
```

■ TrickMandelstam

■ Description

TrickMandelstam[expr, s, t, u, $m_1^2 + m_2^2 + m_3^2 + m_4^2$] simplifies all sums in expr so that one of the Mandelstam variables s, t or u is eliminated by the relation
$s + t + u = m_1^2 + m_2^2 + m_3^2 + m_4^2$.
The trick is that the resulting sum has the most short number of terms.

■ Examples

```
TrickMandelstam[(s + t - u) (2 M_W^2 - t - u), {s, t, u, 2 M_W^2}] // Factor2
```

$-2\,s\,(u - M_W^2)$

```
TrickMandelstam[M^2 s - s^2 + M^2 t - s t + M^2 u - s u, {s, t, u, 2 M^2}]
```

$2\,M^2\,(M^2 - s)$

■ Trick

■ Description

Trick[exp] performs several basic simplifications without expansion.
Trick[exp] uses ContractDotSimplify and SUNDeltaContract.

■ See also: Calc, Contract, DiracTrick, DotSimplify, DiracTrick.

■ Examples

This calculates g^μν γ_μ and g_ν^ν in D dimensions.

```
Trick[{GA[μ] MT[μ, ν], MTD[ν, ν]}]
```

$\{\gamma^\nu,\ D\}$

```
FV[p + r, μ] MT[μ, ν] FV[q - p, ν]
```

$((q - p)^\nu)\,((p + r)^\mu)\,g^{\mu\nu}$

```
Trick[%]
```

$-p^2 + p\cdot q - p\cdot r + q\cdot r$

```
Trick[c.b.a . GA[d].GA[e]]
```

$a\,b\,c\,(\gamma^d\,\gamma^e)$

```
% // FCE // InputForm
```

```
a*b*c*GA[d] . GA[e]
```

■ Tr

■ Description

Tr[exp] calculates the Dirac trace of exp. Depending on the setting of the option SUNTrace also a trace over SU(N) objects is performed. Tr is identical to DiracTrace, up to the default setting of DiracTraceEvaluate.

```
Options[Tr]
```

```
{DiracTraceEvaluate → True, Factoring → False, FeynCalcExternal → False, LeviCivitaSign → -1,
  Mandelstam → {}, PairCollect → False, Schouten → 442, SUNTrace → True, TraceOfOne → 4}
```

■ See also: DiracSimplify, DiracTrace, SUNTrace.

■ Examples

```
GA[μ, ν]
```

$\gamma^{\mu}\,\gamma^{\nu}$

```
Tr[%]
```

$4\,g^{\mu\nu}$

```
Tr[(GSD[p] + m).GAD[μ].(GSD[q] - m).GAD[ν]]
```

$4\,(-g^{\mu\nu}\,m^2 + q^{\mu}\,p^{\nu} + p^{\mu}\,q^{\nu} - g^{\mu\nu}\,p\cdot q)$

```
Tr[GA[μ, ν, ρ, σ, 5]]
```

$-4\,i\,\epsilon^{\mu\nu\rho\sigma}$

```
Tr[GAD[μ, ν, ρ, σ, τ, ξ, 5]]
```

$$4\,(i\,\epsilon^{\xi\rho\sigma\tau}\,g^{\mu\nu} - i\,\epsilon^{\nu\rho\sigma\tau}\,g^{\mu\xi} + i\,\epsilon^{\nu\xi\sigma\tau}\,g^{\mu\rho} - i\,\epsilon^{\nu\xi\rho\tau}\,g^{\mu\sigma} + i\,\epsilon^{\nu\xi\rho\sigma}\,g^{\mu\tau} + i\,\epsilon^{\mu\rho\sigma\tau}\,g^{\nu\xi} - i\,\epsilon^{\mu\xi\sigma\tau}\,g^{\nu\rho} + i\,\epsilon^{\mu\xi\rho\tau}\,g^{\nu\sigma} - i\,\epsilon^{\mu\xi\rho\sigma}\,g^{\nu\tau} - i\,\epsilon^{\mu\nu\sigma\tau}\,g^{\xi\rho} + i\,\epsilon^{\mu\nu\rho\tau}\,g^{\xi\sigma} - i\,\epsilon^{\mu\nu\rho\sigma}\,g^{\xi\tau} + i\,\epsilon^{\mu\nu\xi\tau}\,g^{\rho\sigma} - i\,\epsilon^{\mu\nu\xi\sigma}\,g^{\rho\tau} + i\,\epsilon^{\mu\nu\xi\rho}\,g^{\sigma\tau})$$

```
$West
```

True

```
? $West
```

If $West is set to True (which is the default), traces involving more than 4 Dirac matrices and gamma5 are calculated recursively according to formula (A.5) from Comp Phys Comm 77 (1993) 286–298, which is based on the Breiten-lohner-Maison gamma5 – scheme

$West=False should be set before any calculation is done (because intermediate results from trace calculations are cached).

```
$West = False
```

False

```
Tr[GAD[ν1, ν2, ν3, ν4, ν5, ν6].GA[5]]
```

$$4\,(-i\,\epsilon^{\nu_3\,\nu_4\,\nu_5\,\nu_6}\,g^{\nu_1\,\nu_2} + i\,\epsilon^{\nu_2\,\nu_4\,\nu_5\,\nu_6}\,g^{\nu_1\,\nu_3} - i\,\epsilon^{\nu_1\,\nu_4\,\nu_5\,\nu_6}\,g^{\nu_2\,\nu_3} - i\,\epsilon^{\nu_1\,\nu_2\,\nu_3\,\nu_6}\,g^{\nu_4\,\nu_5} + i\,\epsilon^{\nu_1\,\nu_2\,\nu_3\,\nu_5}\,g^{\nu_4\,\nu_6} - i\,\epsilon^{\nu_1\,\nu_2\,\nu_3\,\nu_4}\,g^{\nu_5\,\nu_6})$$

```
Tr[GS[p, q, r, s]]
```

$$4\,(p\cdot s\,q\cdot r - p\cdot r\,q\cdot s + p\cdot q\,r\cdot s)$$

```
Tr[(GS[p] + m).GA[μ].(GS[q] + m).GA[μ], Factoring → True]
```

$$8\,(2\,m^2 - p\cdot q)$$

```
Tr[GA[α, β], FCE → True]
```

$$4\,g^{\alpha\,\beta}$$

```
% // InputForm
```

```
4*MT[α, β]
```

■ Twist2GluonOperator

■ Description

Twist2GluonOperator[{p,mu,a}, {nu, b}] or Twist2GluonOperator[{p,mu, a}, {nu, b}] or Twist2GluonOperator[p,mu,a, nu,b] yields the 2–gluon operator (p is ingoing momentum corresponding to Lorentz index mu).
Twist2GluonOperator[{p,mu,a}, {q,nu,b}, {k,la,c}] or
Twist2GluonOperator[p,mu,a, q,nu,b, k,la,c] gives the 3–gluon operator.
Twist2GluonOperator[{p,mu,a}, {q,nu,b}, {k,la,c}, {s,si,d}] or
Twist2GluonOperator[p,mu,a, q,nu,b, k,la,c, s,si,d]
yields the 4–Gluon operator.
The dimension is determined by the option Dimension.
The setting of the option Polarization (unpolarized: 0; polarized: 1)
determines whether the unpolarized or polarized feynman rule is returned.
With the setting Explicit to False the color–structure and the (1+(–1)^OPEm)
(for polarized (1–(–1)^OPEm)) is extracted and the color indices are omitted in the arguments of Twist2GluonOperator.

■ See also: Twist2QuarkOperator.

■ Polarized case, zero–momentum insertion

```
SetOptions[Twist2GluonOperator, Polarization -> 1,
 Explicit -> False,  ZeroMomentumInsertion -> True]
```

{CouplingConstant → g_s, Dimension → D, Polarization → 1, Explicit → False, ZeroMomentumInsertion → True}

■ 2–gluonlegs

```
Twist2GluonOperator[p, {μ, a}, {ν, b}]
```

$i\,(1-(-1)^m)\,\delta_{ab}\,(O^{G2}_{\mu\nu}(p))$

```
Twist2GluonOperator[p, {μ, a}, {ν, b}, Explicit -> True]
```

$i\,\epsilon^{\mu\nu\Delta\,p}\,(1-(-1)^m)\,(\Delta\cdot p)^{m-1}\,\delta_{ab}$

```
Twist2GluonOperator[p, {μ}, {ν}]
```

$O^{G2}_{\mu\nu}(p)$

```
Twist2GluonOperator[p, {μ}, {ν}, Explicit -> True]
```

$\epsilon^{\mu\nu\Delta\,p}\,(\Delta\cdot p)^{m-1}$

■ 3–gluonlegs

```
Twist2GluonOperator[{p, μ, a}, {q, ν, b}, {r, ρ, c}]
```

$g_s\,(1-(-1)^m)\,f_{abc}\,(O^{G3}_{\nu\rho\mu}(q,\,r,\,p))$

```
Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}]
```

$O^{G3}_{\nu\rho\mu}(q,\,r,\,p)$

```
Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}, Explicit -> True]
```

$$-(\epsilon^{\mu\rho\,p\Delta}\,\Delta^{\nu}-\epsilon^{\mu\nu\,p\Delta}\,\Delta^{\rho})\,(\Delta\cdot p)^{m-2}-(\epsilon^{\nu\rho\Delta\,q}\,\Delta^{\mu}+\epsilon^{\mu\nu\Delta\,q}\,\Delta^{\rho})\,(\Delta\cdot q)^{m-2}-$$
$$(\epsilon^{\nu\rho\Delta\,r}\,\Delta^{\mu}-\epsilon^{\mu\rho\Delta\,r}\,\Delta^{\nu})\,(\Delta\cdot r)^{m-2}+\left(\sum_{i=0}^{m-3}(-1)^i\,(\Delta\cdot p)^i\,(\Delta\cdot q)^{-i+m-3}\right)\Delta^{\rho}\,(\epsilon^{\nu\,p\Delta\,q}\,\Delta^{\mu}+\epsilon^{\mu\nu\Delta\,q}\,\Delta\cdot p)-$$
$$\left(\sum_{i=0}^{m-3}(-1)^i\,(\Delta\cdot p)^i\,(\Delta\cdot r)^{-i+m-3}\right)\Delta^{\nu}\,(\epsilon^{\rho\,p\Delta\,r}\,\Delta^{\mu}+\epsilon^{\mu\rho\Delta\,r}\,\Delta\cdot p)-\left(\sum_{i=0}^{m-3}(-1)^i\,(\Delta\cdot q)^i\,(\Delta\cdot r)^{-i+m-3}\right)\Delta^{\mu}\,(\epsilon^{\rho\Delta\,q\,r}\,\Delta^{\nu}-\epsilon^{\nu\rho\Delta\,r}\,\Delta\cdot q)$$

■ 4–gluonlegs

```
Twist2GluonOperator[{p, μ, a}, {q, ν, b}, {r, ρ, c}, {s, σ, d}]
```

$i\,(1-(-1)^m)\,g_s^2\,(-f_{abc116}\,f_{cc116d}\,(O^{G4}_{\mu\nu\rho\sigma}(p,\,q,\,r,\,s))-f_{acc116}\,f_{bc116d}\,(O^{G4}_{\mu\rho\nu\sigma}(p,\,r,\,q,\,s))+f_{ac116d}\,f_{bcc116}\,(O^{G4}_{\rho\nu\mu\sigma}(r,\,q,\,p,\,s)))$

```
Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}, {s, σ}]
```

$O^{G4}_{\mu\nu\rho\sigma}(p,\,q,\,r,\,s)$

```
Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}, {s, σ}, Explicit -> True]
```

$$(\epsilon^{\Delta\nu\rho\sigma}\,\Delta^{\mu}-\epsilon^{\Delta\mu\rho\sigma}\,\Delta^{\nu})\,(\Delta\cdot r+\Delta\cdot s)^{m-2}-\left(\sum_{i=0}^{m-3}(-(\Delta\cdot p)-\Delta\cdot q)^{i}\,(\Delta\cdot s)^{-i+m-3}\right)(\epsilon^{\nu\sigma\Delta s}\,\Delta^{\mu}-\epsilon^{\mu\sigma\Delta s}\,\Delta^{\nu})\,\Delta^{\rho}+$$
$$\left(\sum_{i=0}^{m-3}(\Delta\cdot p+\Delta\cdot q)^{-i+m-3}\,(-(\Delta\cdot r))^{i}\right)(\epsilon^{\rho\nu r\Delta}\,\Delta^{\mu}-\epsilon^{\rho\mu r\Delta}\,\Delta^{\nu})\,\Delta^{\sigma}-$$
$$\left(\sum_{i=0}^{m-3}(-(\Delta\cdot p))^{i}\,(\Delta\cdot r+\Delta\cdot s)^{-i+m-3}\right)\Delta^{\nu}\,(\epsilon^{\mu\sigma\Delta p}\,\Delta^{\rho}-\epsilon^{\mu\rho\Delta p}\,\Delta^{\sigma})+$$
$$\left(\sum_{i=0}^{m-3}(-(\Delta\cdot q))^{i}\,(\Delta\cdot r+\Delta\cdot s)^{-i+m-3}\right)\Delta^{\mu}\,(\epsilon^{\nu\sigma\Delta q}\,\Delta^{\rho}-\epsilon^{\nu\rho\Delta q}\,\Delta^{\sigma})-$$
$$\left(\sum_{j=0}^{m-4}\sum_{i=0}^{j}(\Delta\cdot p)^{-j+m-4}\,(-(\Delta\cdot r)-\Delta\cdot s)^{j-i}\,(-(\Delta\cdot s))^{i}\right)\Delta^{\nu}\,\Delta^{\rho}\,(-\epsilon^{\Delta\sigma ps}\,\Delta^{\mu}-\epsilon^{\mu\sigma\Delta s}\,\Delta\cdot p)+$$
$$\left(\sum_{j=0}^{m-4}\sum_{i=0}^{j}(\Delta\cdot q)^{-j+m-4}\,(-(\Delta\cdot r)-\Delta\cdot s)^{j-i}\,(-(\Delta\cdot s))^{i}\right)\Delta^{\mu}\,\Delta^{\rho}\,(-\epsilon^{\Delta\sigma qs}\,\Delta^{\nu}-\epsilon^{\nu\sigma\Delta s}\,\Delta\cdot q)+$$
$$\left(\sum_{j=0}^{m-4}\sum_{i=0}^{j}(\Delta\cdot p)^{-j+m-4}\,(-(\Delta\cdot r))^{i}\,(-(\Delta\cdot r)-\Delta\cdot s)^{j-i}\right)\Delta^{\nu}\,\Delta^{\sigma}\,(-\epsilon^{\Delta\mu pr}\,\Delta^{\rho}-\epsilon^{\mu\rho\Delta p}\,\Delta\cdot r)-$$
$$\left(\sum_{j=0}^{m-4}\sum_{i=0}^{j}(\Delta\cdot q)^{-j+m-4}\,(-(\Delta\cdot r))^{i}\,(-(\Delta\cdot r)-\Delta\cdot s)^{j-i}\right)\Delta^{\mu}\,\Delta^{\sigma}\,(-\epsilon^{\Delta\nu qr}\,\Delta^{\rho}-\epsilon^{\nu\rho\Delta q}\,\Delta\cdot r)$$

■ Unpolarized case, zero-momentum insertion

```
SetOptions[Twist2GluonOperator,
 Polarization -> 0,                ZeroMomentumInsertion -> True]
```

{CouplingConstant→ g_s, Dimension→ D, Polarization→ 0, Explicit→ False, ZeroMomentumInsertion→ True}

■ 2-gluon legs

```
Twist2GluonOperator[p, {μ, a}, {ν, b}]
```

$$\frac{1}{2}\,((-1)^{m}+1)\,\delta_{ab}\,(O^{G2}_{\mu\nu}(p))$$

```
Twist2GluonOperator[p, {μ, a}, {ν, b}, Explicit -> True]
```

$$\frac{1}{2}\,(g^{\mu\nu}\,(\Delta\cdot p)^{2}-(p^{\mu}\,\Delta^{\nu}+\Delta^{\mu}\,p^{\nu})\,\Delta\cdot p+\Delta^{\mu}\,\Delta^{\nu}\,p^{2})\,((-1)^{m}+1)\,(\Delta\cdot p)^{m-2}\,\delta_{ab}$$

```
Twist2GluonOperator[p, {μ}, {ν}]
```

$$O^{G2}_{\mu\nu}(p)$$

```
Twist2GluonOperator[p, {μ}, {ν}, Explicit -> True]
```

$$(g^{\mu\nu}\,(\Delta\cdot p)^{2}-(p^{\mu}\,\Delta^{\nu}+\Delta^{\mu}\,p^{\nu})\,\Delta\cdot p+\Delta^{\mu}\,\Delta^{\nu}\,p^{2})\,(\Delta\cdot p)^{m-2}$$

■ 3–gluon legs

```
Twist2GluonOperator[{p, μ, a}, {q, ν, b}, {r, ρ, c}]
```

$$-\frac{1}{2}\, i\, g_s\, ((-1)^m + 1)\, f_{abc}\, (O^{\text{G3}}_{\nu\rho\mu}(q,\, r,\, p))$$

```
Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}]
```

$$O^{\text{G3}}_{\nu\rho\mu}(q,\, r,\, p)$$

```
Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}, Explicit -> True]
```

$$\left(\sum_{i=0}^{m-3} (-(\Delta\cdot p))^i\, (\Delta\cdot q)^{-i+m-3}\right) \Delta^\rho\, (q^\mu\, \Delta^\nu\, \Delta\cdot p - g^{\mu\nu}\, \Delta\cdot q\, \Delta\cdot p + \Delta^\mu\, p^\nu\, \Delta\cdot q - \Delta^\mu\, \Delta^\nu\, p\cdot q) +$$
$$\left(\sum_{i=0}^{m-3} (\Delta\cdot p)^{-i+m-3}\, (-(\Delta\cdot r))^i\right) \Delta^\nu\, (r^\mu\, \Delta^\rho\, \Delta\cdot p - g^{\mu\rho}\, \Delta\cdot r\, \Delta\cdot p + \Delta^\mu\, p^\rho\, \Delta\cdot r - \Delta^\mu\, \Delta^\rho\, p\cdot r) +$$
$$\left(\sum_{i=0}^{m-3} (-(\Delta\cdot q))^i\, (\Delta\cdot r)^{-i+m-3}\right) \Delta^\mu\, (r^\nu\, \Delta^\rho\, \Delta\cdot q - g^{\nu\rho}\, \Delta\cdot r\, \Delta\cdot q + \Delta^\nu\, q^\rho\, \Delta\cdot r - \Delta^\nu\, \Delta^\rho\, q\cdot r) +$$
$$(\Delta^\mu\, (p^\nu\, \Delta^\rho - \Delta^\nu\, p^\rho) + (g^{\mu\rho}\, \Delta^\nu - g^{\mu\nu}\, \Delta^\rho)\, \Delta\cdot p)\, (\Delta\cdot p)^{m-2} +$$
$$(\Delta^\nu\, (\Delta^\mu\, q^\rho - q^\mu\, \Delta^\rho) + (g^{\mu\nu}\, \Delta^\rho - \Delta^\mu\, g^{\nu\rho})\, \Delta\cdot q)\, (\Delta\cdot q)^{m-2} + ((r^\mu\, \Delta^\nu - \Delta^\mu\, r^\nu)\, \Delta^\rho + (\Delta^\mu\, g^{\nu\rho} - g^{\mu\rho}\, \Delta^\nu)\, \Delta\cdot r)\, (\Delta\cdot r)^{m-2}$$

■ 4–gluon legs

```
Twist2GluonOperator[{p, μ, a}, {q, ν, b}, {r, ρ, c}, {s, σ, d}]
```

$$\frac{1}{2}\, (1 + (-1)^m)\, g_s^2\, (-f_{abc\text{117}}\, f_{cc\text{117}d}\, (O^{\text{G4}}_{\mu\nu\rho\sigma}(p,\, q,\, r,\, s)) - f_{ac\text{117}}\, f_{bc\text{117}d}\, (O^{\text{G4}}_{\mu\rho\nu\sigma}(p,\, r,\, q,\, s)) + f_{ac\text{117}d}\, f_{bcc\text{117}}\, (O^{\text{G4}}_{\rho\nu\mu\sigma}(r,\, q,\, p,\, s)))$$

```
Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}, {s, σ}]
```

$$O^{\text{G4}}_{\mu\nu\rho\sigma}(p,\, q,\, r,\, s)$$

```
Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}, {s, σ}, Explicit -> True]
```

$$(\Delta^\mu\, g^{\nu\sigma}\, \Delta^\rho - g^{\mu\sigma}\, \Delta^\nu\, \Delta^\rho - \Delta^\mu\, g^{\nu\rho}\, \Delta^\sigma + g^{\mu\rho}\, \Delta^\nu\, \Delta^\sigma)\,(\Delta\cdot r + \Delta\cdot s)^{m-2} +$$
$$\left(\sum_{i=0}^{m-3} (-(\Delta\cdot p))^i\,(\Delta\cdot r+\Delta\cdot s)^{-i+m-3}\right)\Delta^\nu\,(\Delta^\mu\, p^\rho\, \Delta^\sigma - g^{\mu\rho}\,\Delta\cdot p\Delta^\sigma - \Delta^\mu\,\Delta^\rho\, p^\sigma + g^{\mu\sigma}\,\Delta^\rho\,\Delta\cdot p) -$$
$$\left(\sum_{i=0}^{m-3} (-(\Delta\cdot q))^i\,(\Delta\cdot r+\Delta\cdot s)^{-i+m-3}\right)\Delta^\mu\,(\Delta^\nu\, q^\rho\, \Delta^\sigma - g^{\nu\rho}\,\Delta\cdot q\Delta^\sigma - \Delta^\nu\,\Delta^\rho\, q^\sigma + g^{\nu\sigma}\,\Delta^\rho\,\Delta\cdot q) +$$
$$\left(\sum_{i=0}^{m-3} (\Delta\cdot p+\Delta\cdot q)^{-i+m-3}\,(-(\Delta\cdot r))^i\right)\Delta^\sigma\,(r^\mu\,\Delta^\nu\,\Delta^\rho - \Delta^\mu\, r^\nu\,\Delta^\rho + \Delta^\mu\, g^{\nu\rho}\,\Delta\cdot r - g^{\mu\rho}\,\Delta^\nu\,\Delta\cdot r) +$$
$$\left(\sum_{i=0}^{m-3} (-(\Delta\cdot p)-\Delta\cdot q)^i\,(\Delta\cdot s)^{-i+m-3}\right)\Delta^\rho\,(s^\mu\,\Delta^\nu\,\Delta^\sigma - \Delta^\mu\, s^\nu\,\Delta^\sigma + \Delta^\mu\, g^{\nu\sigma}\,\Delta\cdot s - g^{\mu\sigma}\,\Delta^\nu\,\Delta\cdot s) -$$
$$\left(\sum_{j=0}^{m-4}\sum_{i=0}^{j} (\Delta\cdot p)^{-j+m-4}\,(-(\Delta\cdot r))^i\,(-(\Delta\cdot r)-\Delta\cdot s)^{j-i}\right)\Delta^\nu\,\Delta^\sigma\,(-r^\mu\,\Delta^\rho\,\Delta\cdot p + g^{\mu\rho}\,\Delta\cdot r\,\Delta\cdot p - \Delta^\mu\, p^\rho\,\Delta\cdot r + \Delta^\mu\,\Delta^\rho\, p\cdot r) +$$
$$\left(\sum_{j=0}^{m-4}\sum_{i=0}^{j} (\Delta\cdot p)^{-j+m-4}\,(-(\Delta\cdot r)-\Delta\cdot s)^{j-i}\,(-(\Delta\cdot s))^i\right)\Delta^\nu\,\Delta^\rho\,(-s^\mu\,\Delta^\sigma\,\Delta\cdot p + g^{\mu\sigma}\,\Delta\cdot s\,\Delta\cdot p - \Delta^\mu\, p^\sigma\,\Delta\cdot s + \Delta^\mu\,\Delta^\sigma\, p\cdot s) +$$
$$\left(\sum_{j=0}^{m-4}\sum_{i=0}^{j} (\Delta\cdot q)^{-j+m-4}\,(-(\Delta\cdot r))^i\,(-(\Delta\cdot r)-\Delta\cdot s)^{j-i}\right)\Delta^\mu\,\Delta^\sigma\,(-r^\nu\,\Delta^\rho\,\Delta\cdot q + g^{\nu\rho}\,\Delta\cdot r\,\Delta\cdot q - \Delta^\nu\, q^\rho\,\Delta\cdot r + \Delta^\nu\,\Delta^\rho\, q\cdot r) -$$
$$\left(\sum_{j=0}^{m-4}\sum_{i=0}^{j} (\Delta\cdot q)^{-j+m-4}\,(-(\Delta\cdot r)-\Delta\cdot s)^{j-i}\,(-(\Delta\cdot s))^i\right)\Delta^\mu\,\Delta^\rho\,(-s^\nu\,\Delta^\sigma\,\Delta\cdot q + g^{\nu\sigma}\,\Delta\cdot s\,\Delta\cdot q - \Delta^\nu\, q^\sigma\,\Delta\cdot s + \Delta^\nu\,\Delta^\sigma\, q\cdot s)$$

■ AdvancedExamples

ThesettingAll for Explicit performsthesums.

```
SetOptions[Twist2GluonOperator, Polarization -> 1, Explicit -> All];
```

```
Twist2GluonOperator[{p, μ, a}, {q, ν, b}, {r, ρ, c}];
```

■ Polarizedcase,non−zero−momentuminsertion

```
SetOptions[Twist2GluonOperator, Polarization -> 1, ZeroMomentumInsertion -> False]
```

{CouplingConstant→ g_s, Dimension→ D, Polarization→ 1, Explicit→ All, ZeroMomentumInsertionFalse}

■ 2−gluonlegs

```
Twist2GluonOperator[{p, q}, {μ, a}, {ν, b}]
```

$$\frac{1}{2}\,i\,(1-(-1)^m)\,((\epsilon^{\nu\Delta\,pq}\,\Delta^\mu - \epsilon^{\mu\nu\Delta\,q}\,\Delta\cdot p)\,(\Delta\cdot p)^{m-2} + (\epsilon^{\mu\nu\Delta\,p}\,\Delta\cdot q - \epsilon^{\mu\Delta\,pq}\,\Delta^\nu)\,(\Delta\cdot q)^{m-2})\,\delta_{ab}$$

■ 3–gluon legs

```
Twist2GluonOperator[{p, μ, a}, {q, ν, b}, {r, ρ, c}]
```

$$g_s\Bigg(-\epsilon^{\nu\rho\Delta p}\,\Delta^{\mu}\,(\Delta\cdot p)^{m-2}+\epsilon^{\mu\rho\Delta q}\,\Delta^{\nu}\,(\Delta\cdot q)^{m-2}-\epsilon^{\mu\nu\Delta r}\,\Delta^{\rho}\,(\Delta\cdot r)^{m-2}-$$

$$\epsilon^{\rho\Delta pr}\left(\sum_{i=0}^{m-3}(\Delta\cdot p)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot q)^{i}\right)\Delta^{\mu}\,\Delta^{\nu}+\epsilon^{\rho\Delta qr}\left(\sum_{i=0}^{m-3}(\Delta\cdot q)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot q)^{i}\right)\Delta^{\mu}\,\Delta^{\nu}+$$

$$\epsilon^{\nu\Delta pq}\left(\sum_{i=0}^{m-3}(\Delta\cdot p)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot r)^{i}\right)\Delta^{\mu}\,\Delta^{\rho}+\epsilon^{\nu\Delta qr}\left(\sum_{i=0}^{m-3}(\Delta\cdot r)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot r)^{i}\right)\Delta^{\mu}\,\Delta^{\rho}+$$

$$\epsilon^{\mu\Delta pq}\left(\sum_{i=0}^{m-3}(\Delta\cdot q)^{-i+m-3}\,(\Delta\cdot q+\Delta\cdot r)^{i}\right)\Delta^{\nu}\,\Delta^{\rho}-\epsilon^{\mu\Delta pr}\left(\sum_{i=0}^{m-3}(\Delta\cdot r)^{-i+m-3}\,(\Delta\cdot q+\Delta\cdot r)^{i}\right)\Delta^{\nu}\,\Delta^{\rho}+$$

$$\frac{1}{(\Delta\cdot p+\Delta\cdot q)^2}\Bigg(\epsilon^{\mu\rho\Delta r}\,\Delta^{\nu}\Bigg((\Delta\cdot p+\Delta\cdot q)^{m}+\left(\sum_{i=0}^{m-3}(\Delta\cdot p)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot q)^{i}\right)(\Delta\cdot p)^3+$$

$$\left(\sum_{i=0}^{m-3}(\Delta\cdot p)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot q)^{i}\right)\Delta\cdot p(\Delta\cdot q)^2+2\left(\sum_{i=0}^{m-3}(\Delta\cdot p)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot q)^{i}\right)(\Delta\cdot p)^2\,\Delta\cdot q\Bigg)\Bigg)-$$

$$\frac{1}{(\Delta\cdot p+\Delta\cdot q)^2}\Bigg(\epsilon^{\nu\rho\Delta r}\,\Delta^{\mu}\Bigg((\Delta\cdot p+\Delta\cdot q)^{m}+\left(\sum_{i=0}^{m-3}(\Delta\cdot q)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot q)^{i}\right)(\Delta\cdot q)^3+$$

$$2\left(\sum_{i=0}^{m-3}(\Delta\cdot q)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot q)^{i}\right)\Delta\cdot p(\Delta\cdot q)^2+\left(\sum_{i=0}^{m-3}(\Delta\cdot q)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot q)^{i}\right)(\Delta\cdot p)^2\,\Delta\cdot q\Bigg)\Bigg)-$$

$$\frac{1}{(\Delta\cdot p+\Delta\cdot r)^2}\Bigg(\epsilon^{\mu\nu\Delta q}\,\Delta^{\rho}\Bigg((\Delta\cdot p+\Delta\cdot r)^{m}+\left(\sum_{i=0}^{m-3}(\Delta\cdot p)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot r)^{i}\right)(\Delta\cdot p)^3+$$

$$\left(\sum_{i=0}^{m-3}(\Delta\cdot p)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot r)^{i}\right)\Delta\cdot p(\Delta\cdot r)^2+2\left(\sum_{i=0}^{m-3}(\Delta\cdot p)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot r)^{i}\right)(\Delta\cdot p)^2\,\Delta\cdot r\Bigg)\Bigg)-$$

$$\frac{1}{(\Delta\cdot p+\Delta\cdot r)^2}\Bigg(\epsilon^{\nu\rho\Delta q}\,\Delta^{\mu}\Bigg((\Delta\cdot p+\Delta\cdot r)^{m}+\left(\sum_{i=0}^{m-3}(\Delta\cdot r)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot r)^{i}\right)(\Delta\cdot r)^3+$$

$$2\left(\sum_{i=0}^{m-3}(\Delta\cdot r)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot r)^{i}\right)\Delta\cdot p(\Delta\cdot r)^2+\left(\sum_{i=0}^{m-3}(\Delta\cdot r)^{-i+m-3}\,(\Delta\cdot p+\Delta\cdot r)^{i}\right)(\Delta\cdot p)^2\,\Delta\cdot r\Bigg)\Bigg)-$$

$$\frac{1}{(\Delta\cdot q+\Delta\cdot r)^2}\Bigg(\epsilon^{\mu\nu\Delta p}\,\Delta^{\rho}\Bigg((\Delta\cdot q+\Delta\cdot r)^{m}+\left(\sum_{i=0}^{m-3}(\Delta\cdot q)^{-i+m-3}\,(\Delta\cdot q+\Delta\cdot r)^{i}\right)(\Delta\cdot q)^3+$$

$$\left(\sum_{i=0}^{m-3}(\Delta\cdot q)^{-i+m-3}\,(\Delta\cdot q+\Delta\cdot r)^{i}\right)\Delta\cdot q(\Delta\cdot r)^2+2\left(\sum_{i=0}^{m-3}(\Delta\cdot q)^{-i+m-3}\,(\Delta\cdot q+\Delta\cdot r)^{i}\right)(\Delta\cdot q)^2\,\Delta\cdot r\Bigg)\Bigg)+$$

$$\frac{1}{(\Delta\cdot q+\Delta\cdot r)^2}\Bigg(\epsilon^{\mu\rho\Delta p}\,\Delta^{\nu}\Bigg((\Delta\cdot q+\Delta\cdot r)^{m}+\left(\sum_{i=0}^{m-3}(\Delta\cdot r)^{-i+m-3}\,(\Delta\cdot q+\Delta\cdot r)^{i}\right)(\Delta\cdot r)^3+$$

$$2\left(\sum_{i=0}^{m-3}(\Delta\cdot r)^{-i+m-3}\,(\Delta\cdot q+\Delta\cdot r)^{i}\right)\Delta\cdot q(\Delta\cdot r)^2+\left(\sum_{i=0}^{m-3}(\Delta\cdot r)^{-i+m-3}\,(\Delta\cdot q+\Delta\cdot r)^{i}\right)(\Delta\cdot q)^2\,\Delta\cdot r\Bigg)\Bigg)-$$

$$\frac{\epsilon^{\mu\nu\rho\Delta}\,(\Delta\cdot q\Delta\cdot r(\Delta\cdot p)^m+\Delta\cdot q(\Delta\cdot r)^m\Delta\cdot p+(\Delta\cdot q)^m\Delta\cdot r\Delta\cdot p)}{\Delta\cdot q\Delta\cdot r\Delta\cdot p}\Bigg)\,f_{abc}$$

■ 4–gluon legs

```
Twist2GluonOperator[{p, μ, a}, {q, ν, b}, {r, ρ, c}, {s, σ, d},      Explicit -> False]
```

$$i\,(1-(-1)^m)\,g_s^2\,(-f_{abc118}\,f_{cc118d}\,(O^{G4}_{\mu\nu\rho\sigma}(p,q,r,s))-f_{acc118}\,f_{bc118d}\,(O^{G4}_{\mu\rho\nu\sigma}(p,r,q,s))+f_{ac118d}\,f_{bcc118}\,(O^{G4}_{\rho\nu\mu\sigma}(r,q,p,s)))$$

■ Unpolarized case, non–zero–momentum insertion

```
SetOptions[Twist2GluonOperator, Polarization -> 0, ZeroMomentumInsertion -> False]
```

{CouplingConstant → g_s, Dimension → D, Polarization → 0, Explicit → All, ZeroMomentumInsertion → False}

■ 2–gluon legs

```
Twist2GluonOperator[{p, q}, {μ, a}, {ν, b}]
```

$$-\frac{1}{2}\,(1+(-1)^m)\,(q^\mu\,\Delta^\nu\,\Delta\cdot p-g^{\mu\nu}\,\Delta\cdot q\,\Delta\cdot p+\Delta^\mu\,(p^\nu\,\Delta\cdot q-\Delta^\nu\,p\cdot q))\,((\Delta\cdot p)^{m-2}+(\Delta\cdot q)^{m-2})\,\delta_{ab}$$

■ 3–gluon legs

```
Twist2GluonOperator[{p, μ, a}, {q, ν, b}, {r, ρ, c}];
```

```
Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}]
```

$$-\frac{1}{2}\,(1+(-1)^m)\,(\mu^q\,\Delta^r\,\Delta\cdot p-g^{qr}\,\Delta\cdot\mu\,\Delta\cdot p+\Delta^q\,(p^r\,\Delta\cdot\mu-\Delta^r\,p\cdot\mu))\,((\Delta\cdot p)^{m-2}+(\Delta\cdot\mu)^{m-2})\,\delta_{\nu\rho}$$

```
Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}, Explicit -> True]
```

$$-\frac{1}{2}\,(1+(-1)^m)\,(\mu^q\,\Delta^r\,\Delta\cdot p-g^{qr}\,\Delta\cdot\mu\,\Delta\cdot p+\Delta^q\,(p^r\,\Delta\cdot\mu-\Delta^r\,p\cdot\mu))\,((\Delta\cdot p)^{m-2}+(\Delta\cdot\mu)^{m-2})\,\delta_{\nu\rho}$$

■ 4–gluon legs

```
Twist2GluonOperator[{p, μ, a}, {q, ν, b}, {r, ρ, c}, {s, σ, d},      Explicit -> False]
```

$$\frac{1}{2}\,(1+(-1)^m)\,g_s^2\,(-f_{abc119}\,f_{cc119d}\,(O^{G4}_{\mu\nu\rho\sigma}(p,q,r,s))-f_{acc119}\,f_{bc119d}\,(O^{G4}_{\mu\rho\nu\sigma}(p,r,q,s))+f_{ac119d}\,f_{bcc119}\,(O^{G4}_{\rho\nu\mu\sigma}(r,q,p,s)))$$

Suppress the lengthy output.

```
Short[Twist2GluonOperator[{p, μ}, {q, ν}, {r, ρ}, {s, σ}, Explicit -> True], 6]
```

$$(\Delta^\mu\,g^{\nu\sigma}\,\Delta^\rho-g^{\mu\sigma}\,\Delta^\nu\,\Delta^\rho-\Delta^\mu\,g^{\nu\rho}\,\Delta^\sigma+g^{\mu\rho}\,\Delta^\nu\,\Delta^\sigma)\,(\Delta\cdot r+\Delta\cdot s)^{m-2}+\text{«8»}+\left(\sum_{j=0}^{m-4}\sum_{i=0}^{j}\text{«1»}\right)\text{«2»}\,(\text{«1»})-$$

$$\left(\sum_{j=0}^{m-4}\sum_{i=0}^{j}(\Delta\cdot q)^{-j+m-4}\,(-(\Delta\cdot r)-\Delta\cdot s)^{j-i}\,(-(\Delta\cdot s))^i\right)\Delta^\mu\,\Delta^\rho\,(-s^\nu\,\Delta^\sigma\,\Delta\cdot q+g^{\nu\sigma}\,\Delta\cdot s\,\Delta\cdot q-\Delta^\nu\,q^\sigma\,\Delta\cdot s+\Delta^\nu\,\Delta^\sigma\,q\cdot s)$$

The setting Explicit–>All performs the sums.

```
SetOptions[Twist2GluonOperator, Polarization -> 1, Explicit -> All];
```

```
Short[Twist2GluonOperator[{p, μ, a}, {q, ν, b}, {r, ρ, c}], 5]
```

$$g_s\left(-\epsilon^{\nu\rho\Delta p}\,\Delta^{\mu}\,(\Delta\cdot p)^{m-2}+\epsilon^{\mu\rho\Delta q}\,\Delta^{\nu}\,(\Delta\cdot q)^{m-2}-\epsilon^{\mu\nu\Delta r}\,\Delta^{\rho}\,(\Delta\cdot r)^{m-2}-\epsilon^{\rho\Delta pr}\left(\sum_{i=0}^{m-3}(\Delta\cdot p)^{-i+m-3}\,(\ll 1\gg)^{i}\right)\ll 1\gg^{\ll 1\gg}\Delta^{\nu}+\right.$$
$$\left.\ll 15\gg+\frac{\epsilon^{\mu\rho\Delta p}\,\Delta^{\nu}\,(\ll 1\gg)}{(\Delta\cdot q+\Delta\cdot r)^2}-\frac{\epsilon^{\mu\nu\rho\Delta}\,(\Delta\cdot q\Delta\cdot r(\Delta\cdot p)^m+\Delta\cdot q(\Delta\cdot r)^m\Delta\cdot p+(\Delta\cdot q)^m\Delta\cdot r\Delta\cdot p)}{\Delta\cdot q\Delta\cdot r\Delta\cdot p}\right)f_{abc}$$

■ Twist2QuarkOperator

■ Description

Twist2QuarkOperator[p] or Twist2QuarkOperator[p,_,y] yields the quark–antiquark operator (p is momentum in the direction of the incoming quark).
Twist2QuarkOperator[{p,q}] yields the 2–quark operator for non–zero momentum insertion (p is momentum in the direction of the incoming quark).

Twist2QuarkOperator[{p1,___},{p2,___}, {p3, mu,a}] or
Twist2QuarkOperator[p1,_,p2,_,_, p3,mu,a] is the quark–antiquark–gluon
operator, where p1 is the incoming quark, p2 the incoming antiquark and p3 denotes the incoming gluon momentum.

Twist2QuarkOperator[{p1,p2}, {p3, mu,a}, {p4, nu,b}] or
Twist2QuarkOperator[{p1,___},{p2,___}, {p3, mu,a}, {p4, nu,b}] or Twist2QuarkOperator[p1,_,p2,_,_, p3,mu,ap4, nu,b] gives the Quark–Quark–Gluon–Gluon–operator.

The setting of the option Polarization (unpolarized: 0; polarized: 1) determines whether the unpolarized or polarized operator is returned

■ See also: Twist2GluonOperator.

■ Examples

■ Polarized case, zero–momentum insertion

```
SetOptions[Twist2QuarkOperator, Polarization → 1, ZeroMomentumInsertion → True]
```

```
{CouplingConstant → g_s, Dimension → D, Explicit → True, Polarization → 1, ZeroMomentumInsertion → True}
```

Quark–antiquark operator.

```
t1 = Twist2QuarkOperator[p]
```

$$-(\gamma\cdot\Delta\,\gamma^5)\,(\Delta\cdot p)^{m-1}$$

Quark–antiquark–gluon operator.

```
t2 = Twist2QuarkOperator[{p}, {q}, {k, μ, a}]
```

$$-g_s\,(\gamma\cdot\Delta\,\gamma^5\,T_a)\left(\sum_{i=0}^{m-2}(-1)^i\,(\Delta\cdot p)^{-i+m-2}\,(\Delta\cdot q)^i\right)\Delta^\mu$$

Quark–antiquark–gluon–gluon operator.

```
t3 = Twist2QuarkOperator[{p}, {q}, {k, μ, a}, {r, ν, b}]
```

$$-g_s^2\left(\gamma\cdot\Delta\,\gamma^5\left((T_b\,T_a)\left(\sum_{j=0}^{m-3}\sum_{i=0}^{j}(-1)^j\,(\Delta\cdot p)^i\,(\Delta\cdot q)^{-j+m-3}\,(\Delta\cdot p+\Delta\cdot r)^{j-i}\right)-\right.\right.$$
$$\left.\left.(-1)^m\,(T_a\,T_b)\left(\sum_{j=0}^{m-3}\sum_{i=0}^{j}(-1)^j\,(\Delta\cdot p)^{-j+m-3}\,(\Delta\cdot q)^i\,(\Delta\cdot q+\Delta\cdot r)^{j-i}\right)\right)\right)\Delta^\mu\,\Delta^\nu$$

■ Unpolarized case, zero–momentum insertion

```
SetOptions[Twist2QuarkOperator, Polarization -> 0, ZeroMomentumInsertion -> True]
```

{CouplingConstant→ g_s, Dimension→ D, Explicit→ True, Polarization→ 0, ZeroMomentumInsertion→ True}

Quark–antiquark operator.

```
t4 = Twist2QuarkOperator[p]
```

$$\gamma\cdot\Delta\,(\Delta\cdot p)^{m-1}$$

Quark–antiquark–gluon operator.

```
t5 = Twist2QuarkOperator[{p}, {q}, {k, μ, a}]
```

$$g_s\,(\gamma\cdot\Delta\,T_a)\left(\sum_{i=0}^{m-2}(-1)^i\,(\Delta\cdot p)^{-i+m-2}\,(\Delta\cdot q)^i\right)\Delta^\mu$$

Quark–antiquark–gluon–gluon operator.

```
t6 = Twist2QuarkOperator[{p}, {q}, {k, μ, a}, {r, ν, b}]
```

$$-(-1)^m\,g_s^2\left(\gamma\cdot\Delta\left((T_a\,T_b)\left(\sum_{i=0}^{m-3}\sum_{j=0}^{i}(-(\Delta\cdot p))^{-i+m-3}\,(\Delta\cdot q)^j\,(k\cdot\Delta+\Delta\cdot q)^{i-j}\right)+(T_b\,T_a)\left(\sum_{i=0}^{m-3}\sum_{j=0}^{i}(-(\Delta\cdot p))^{-i+m-3}\,(\Delta\cdot q)^j\,(\Delta\cdot q+\Delta\cdot r)^{i-j}\right)\right)\right)\Delta^\mu\,\Delta^\nu$$

This shows the FeynCalc External (FCE) form.

```
Twist2QuarkOperator[p] // FCE // InputForm
```

```
GSD[OPEDelta]*SOD[p]^(-1 + OPEm)
```

```
Twist2QuarkOperator[{p}, {q}, {k, μ, a}, Explicit → All, Polarization → 0]
```

$$-g_s\,(\gamma\cdot\Delta\;T_a)\,\Delta^{\mu}\left(\frac{(\Delta\cdot p)^{m-1}}{\Delta\cdot p+\Delta\cdot q}+\frac{(-1)^m\,(\Delta\cdot q)^{m-1}}{\Delta\cdot p+\Delta\cdot q}\right)$$

```
Twist2QuarkOperator[{p}, {q}, {k, μ, a}, {r, ν, b}, Explicit → All, Polarization → 0]
```

$$-g_s^2\left(\gamma\cdot\Delta\left((T_a\;T_b)\left(-\frac{(-1)^m\,(\Delta\cdot p)^{m-1}}{(\Delta\cdot p+\Delta\cdot q)\,(k\cdot\Delta+\Delta\cdot p+\Delta\cdot q)}-\frac{(\Delta\cdot q)^{m-1}}{k\cdot\Delta\,(\Delta\cdot p+\Delta\cdot q)}+\frac{(k\cdot\Delta+\Delta\cdot q)^{m-1}}{k\cdot\Delta\,(k\cdot\Delta+\Delta\cdot p+\Delta\cdot q)}\right)+\right.\right.$$
$$\left.\left.(T_b\;T_a)\left(-\frac{(-1)^m\,(\Delta\cdot p)^{m-1}}{(\Delta\cdot p+\Delta\cdot q)\,(\Delta\cdot p+\Delta\cdot q+\Delta\cdot r)}-\frac{(\Delta\cdot q)^{m-1}}{(\Delta\cdot p+\Delta\cdot q)\,\Delta\cdot r}+\frac{(\Delta\cdot q+\Delta\cdot r)^{m-1}}{\Delta\cdot r\,(\Delta\cdot p+\Delta\cdot q+\Delta\cdot r)}\right)\right)\right)\Delta^{\mu}\,\Delta^{\nu}\,(-1)^m$$

■ Polarized case, non–zero momentum insertion

```
SetOptions[Twist2QuarkOperator, Polarization → 1, ZeroMomentumInsertion → False]
```

{CouplingConstant → g_s, Dimension → D, Explicit → True, Polarization → 1, ZeroMomentumInsertion → False}

With the setting ZeroMomentumInsertion → False a non–zero momentum is assumed to flow into the operator vertex.

This is the Feynman rule associated with the quark–antiquark operator where p is the momentum of the incoming quark and q the momentum of the antiquark. The momentum flowing into the operator vertex is thus –p–q.

```
t1 = Twist2QuarkOperator[{p, q}]
```

$$2^{1-m}\,(\gamma^5\,\gamma\cdot\Delta)\,(\Delta\cdot p-\Delta\cdot q)^{m-1}$$

This is the quark–antiquark–gluon operator vertex.

```
t2 = Twist2QuarkOperator[{p}, {q}, {k, μ, a}]
```

$$2^{2-m}\,g_s\,(T_a\,\gamma^5\,\gamma\cdot\Delta)\left(\sum_{i=0}^{m-2}(-(k\cdot\Delta)+\Delta\cdot p-\Delta\cdot q)^{-i+m-2}\,(k\cdot\Delta+\Delta\cdot p-\Delta\cdot q)^i\right)\Delta^{\mu}$$

This shows the FeynCalcExternal form.

```
t2//FCE//InputForm
```

```
2^(2 - OPEm)*Gstrong*SUNT[a] . GA[5] . GSD[OPEDelta]*
  FVD[OPEDelta, μ]*OPESum[(-SOD[k] + SOD[p] - SOD[q])^
     (-2 - OPEi + OPEm)*(SOD[k] + SOD[p] - SOD[q])^OPEi,
   {OPEi, 0, -2 + OPEm}]
```

■ Unpolarized case, non–zero momentum insertion

```
SetOptions[Twist2QuarkOperator, Polarization → 1, ZeroMomentumInsertion → False]
```

{CouplingConstant → g_s, Dimension → D, Explicit → True, Polarization → 1, ZeroMomentumInsertion → False}

Quark–antiquark operator.

```
t4 = Twist2QuarkOperator[{p, q}]
```

$$2^{1-m} (\gamma^5 \gamma\cdot\Delta) (\Delta\cdot p - \Delta\cdot q)^{m-1}$$

Quark–antiquark–gluon operator.

```
t5 = Twist2QuarkOperator[{p}, {q}, {k, μ, a}]
```

$$2^{2-m} g_s (T_a \gamma^5 \gamma\cdot\Delta) \left(\sum_{i=0}^{m-2} (-(k\cdot\Delta) + \Delta\cdot p - \Delta\cdot q)^{-i+m-2} (k\cdot\Delta + \Delta\cdot p - \Delta\cdot q)^i\right) \Delta^\mu$$

```
Clear[t1, t2, t3, t4, t5];
```

■ Uncontract

■ Description

Uncontract[exp, q1, q2, ...] uncontracts Eps and DiracGamma. Uncontract[exp, q1, q2, Pair -> {p}] uncontracts also p.q1 and p.q2; the option Pair -> All uncontracts all momenta except OPEDelta.

```
Options[Uncontract]
```

{Dimension → *D*, DimensionalReduction → False, Pair → {}, Unique → True}

■ See also: Contract.

■ Examples

```
t1 = LeviCivita[μ, ν][p, q]
```

$$\epsilon^{\mu\nu pq}$$

```
Uncontract[t1, p]
```

$$\epsilon^{\mu\nu \$AL\$3219(1)\, q}\, p^{\$AL\$3219(1)}$$

```
t2 = DiracSlash[p]
```

$$\gamma\cdot p$$

```
Uncontract[t2, p]
```

$$\gamma^{\$AL\$3220(1)}\, p^{\$AL\$3220(1)}$$

```
Uncontract[t1, p, q]
```

$$\epsilon^{\mu\nu \$AL\$3222(1)\, \$AL\$3221(1)}\, q^{\$AL\$3221(1)}\, p^{\$AL\$3222(1)}$$

By default scalar products are not uncontracted.

```
Uncontract[ScalarProduct[p, q], q]
```

$p \cdot q$

With the option Pair→All they are "uncontracted".

```
Uncontract[ScalarProduct[p, q], q, Pair → All]
```

$p^{\$AL\$3224(1)}\ q^{\$AL\$3224(1)}$

```
Uncontract[ScalarProduct[p, q]^2, q, Pair → All]
```

$p^{\$AL\$3225(1)}\ q^{\$AL\$3225(1)}\ p^{\$AL\$3225(2)}\ q^{\$AL\$3225(2)}$

```
Clear[t1, t2]
```

■ UnDeclareNonCommutative

■ Description

UnDeclareNonCommutative[a,b,..] undeclares a,b,... to be noncommutative, i.e., DataType[a,b,.., NonCommutative] is set to False.

■ See also: DataType, DeclareNonCommutative.

■ Examples

```
DeclareNonCommutative[x]
```

As a side–effect of DeclareNonCommutative, x is declared to be of DataType NonCommutative.

```
DataType[x,NonCommutative]
```

True

The inverse operation is UnDeclareNonCommutative.

```
UnDeclareNonCommutative[x]
```

```
DataType[x, NonCommutative]
```

False

```
DeclareNonCommutative[y, z]
```

```
DataType[y, z, NonCommutative]
```

{True, True}

```
UnDeclareNonCommutative[y, z]
```

```
DataType[y, z, NonCommutative]
```

{False, False}

■ Write2

■ Description

Write2[file,val1 = expr1, val2 = expr2,...] writes the settings val1 = expr1, val2 = expr2 in sequence followed by a newline, to the specified output file. Setting the option FormatType of Write2 to FortranForm results in FORTRAN syntax output.

```
Options[Write2]
```

{FinalSubstitutions→ {}, FormatType→ InputForm, D0Convention→ 0,
PageWidth→ 65, PreFortranFile→ , PostFortranFile→ , StringReplace→ {}}

```
Attributes[Write2]
```

{HoldRest, ReadProtected}

■ See also: Isolate, PaVeReduce.

■ Examples

```
t = Collect[(a - b - c)^3, a, Factor]
```

$a^3 - 3(b+c)a^2 + 3(b+c)^2 a - (b+c)^3$

```
tempfilename = ToString[$SessionID] <> ".s"
```

2040234398224732236 0.s

```
Write2[tempfilename, r = t];
```

```
TableForm[ReadList[If[$OperatingSystem === "MacOS", ":", ""] <> tempfilename, String]]
```

r = a^3 − 3∗a^2∗(b + c) + 3∗a∗(b + c)^2 − (b + c)^3

```
DeleteFile[If[$OperatingSystem === "MacOS", ":", ""] <> tempfilename]
```

```
tpol = x + Isolate[t, a]
```

$a^3 - 3\,\mathrm{KK}(10)\,a^2 + 3\,(\mathrm{KK}(10))^2\,a - (\mathrm{KK}(10))^3 + x$

```
KK[1]
```

$-s + {p_1}^2 + {p_2}^2$

```
Write2[tempfilename, r = tpol];
```

```
TableForm[ReadList[If[$OperatingSystem === "MacOS", ":", ""] <> tempfilename, String]]
```

```
KK[10] = (b + c
    );
r = ( a^3 + x - 3*a^2*HoldForm[KK[10]] + 3*a*HoldForm[KK[10]]^2 -
 HoldForm[KK[10]]^3
    );
```

```
DeleteFile[If[$OperatingSystem === "MacOS", ":", ""] <> tempfilename]
```

```
Write2[tempfilename, r = tpol, FormatType -> FortranForm];
```

```
TableForm[ReadList[If[$OperatingSystem === "MacOS", ":", ""] <> tempfilename, String]]
```

```
    KK(10) = b + c
    r = a**3 + x - 3*a**2*KK(10) + 3*a*KK(10)**2 - KK(10)**3
```

```
DeleteFile[If[$OperatingSystem === "MacOS", ":", ""] <> tempfilename]
```

■ Evaluation time and memory usage

```
AbsoluteTime[] - absolutestarttime
```

683.00000

```
MemoryInUse[]
```

10871956

```
MaxMemoryUsed[]
```

14925712