Elementary maths

This example is based on a similar example in the Python collection. Its purpose is to show that Cadabra is fluent in Python – which is not surprisng since the Cadabra language is based on Python (and a subset of LaTeX).

```
from sympy import *
x, y, z = symbols('x y z')
a, b, c = symbols('a b c')
ans = expand((a+b)**3)                                    # cdb (ans.101,ans)
ans = factor(-2*x+2*x+a*x-x**2+a*x**2-x**3)               # cdb (ans.102,ans)
ans = solve(x**2-4, x)                                    # cdb (ans.103,ans)
ans = solve([2*a-b - 3, a+b+c - 1,-b+c - 6],[a,b,c])      # cdb (ans.104,ans)
ans = N(pi,50)                                            # cdb (ans.105,ans)
ans = apart(1/((1 + x)*(5 + x)))                          # cdb (ans.106,ans)
ans = together((1/(1 + x) - 1/(5 + x))/4)                 # cdb (ans.107,ans)
ans = simplify(tanh(log(x)))                              # cdb (rhs.108,ans)
ans = simplify(tanh(I*x))                                 # cdb (rhs.109,ans)
ans = simplify(sinh(3*x) - 3*sinh(x) - 4*(sinh(x))**3)    # cdb (rhs.110,ans)
ans = tanh(log(x))                                        # cdb (lhs.108,ans)
ans = tanh(UnevaluatedExpr(I*x))                          # cdb (lhs.109,ans)
ans = sinh(3*x) - 3*sinh(x) - 4*(sinh(x))**3              # cdb (lhs.110,ans)
```

```
\begin{align*}
   &\cdb*{ans.101}\\
   &\cdb*{ans.102}\\
   &\cdb*{ans.103}\\
   &\cdb*{ans.104}\\
   &\cdb*{ans.105}\\
   &\cdb*{ans.106}\\
   &\cdb*{ans.107}\\
   \cdb{lhs.108} &= \Cdb{rhs.108}\\
   \cdb{lhs.109} &= \Cdb{rhs.109}\\
   \cdb{lhs.110} &= \Cdb{rhs.110}
\end{align*}
```

$$\texttt{ans.101} := a^3 + 3a^2b + 3ab^2 + b^3$$

$$\texttt{ans.102} := -x\,(-a+x)\,(x+1)$$

$$\texttt{ans.103} := [-2,\quad 2]$$

$$\texttt{ans.104} := \left\{a : \frac{1}{5},\quad b : -\frac{13}{5},\quad c : \frac{17}{5}\right\}$$

$$\texttt{ans.105} := 3.1415926535897932384626433832795028841971693993751$$

$$\texttt{ans.106} := -\frac{1}{4\,(x+5)} + \frac{1}{4\,(x+1)}$$

$$\texttt{ans.107} := \frac{1}{(x+1)\,(x+5)}$$

$$\tanh(\log(x)) = \tanh(\log(x)) \qquad (\texttt{rhs.108})$$

$$\tanh(ix) = i\tan(x) \qquad (\texttt{rhs.109})$$

$$-4\sinh^3(x) - 3\sinh(x) + \sinh(3x) = 0 \qquad (\texttt{rhs.110})$$

Linear Algebra

```
from sympy import linsolve
lamda = Symbol('lamda')
mat  = Matrix([[2,3], [5,4]])                     # cdb (ans.201,mat)
eig1 = mat.eigenvects()[0][0]                     # 1st eigenvalue
eig2 = mat.eigenvects()[1][0]                     # 2nd eigenvalue
v1   = mat.eigenvects()[0][2][0]                  # 1st eigenvector
v2   = mat.eigenvects()[1][2][0]                  # 2nd eigenvector
eig  = simplify(Matrix([eig1,eig2]))              # cdb (ans.202,eig)
vec  = simplify(5*Matrix([]).col_insert(0,v1)
                           .col_insert(1,v2))     # cdb (ans.203,vec)
det  = expand((mat - lamda * eye(2)).det())       # cdb (ans.204,det)
rhs  = Matrix([[3],[7]])                          # cdb (ans.205,rhs)
ans  = list(linsolve((mat,rhs),x,y))[0]           # cdb (ans.206,ans)
```

```
\begin{align*}
   &\cdb*{ans.201}\\
   &\cdb*{ans.202}\\
   &\cdb*{ans.203}\\
   &\cdb*{ans.204}\\
   &\cdb*{ans.205}\\
   &\cdb*{ans.206}
\end{align*}
```

$$
\begin{aligned}
\texttt{ans.201} &:= \begin{bmatrix} 2 & 3 \\ 5 & 4 \end{bmatrix} \\
\texttt{ans.202} &:= \begin{bmatrix} -1 \\ 7 \end{bmatrix} \\
\texttt{ans.203} &:= \begin{bmatrix} -5 & 3 \\ 5 & 5 \end{bmatrix} \\
\texttt{ans.204} &:= \lambda^2 - 6\lambda - 7 \\
\texttt{ans.205} &:= \begin{bmatrix} 3 \\ 7 \end{bmatrix} \\
\texttt{ans.206} &:= \left(\frac{9}{7}, \quad \frac{1}{7} \right)
\end{aligned}
$$

Limits

```
n, dx = symbols('n dx')
ans = limit(sin(4*x)/x,x,0)                 # cdb (ans.301,ans)
ans = limit(2**x/x,x,oo)                    # cdb (ans.302,ans)
ans = limit(((x+dx)**2 - x**2)/dx, dx,0)    # cdb (ans.303,ans)
ans = limit((4*n + 1)/(3*n - 1),n,oo)       # cdb (ans.304,ans)
ans = limit((1+(a/n))**n,n,oo)              # cdb (ans.305,ans)
```

```
\begin{align*}
   &\cdb*{ans.301}\\
   &\cdb*{ans.302}\\
   &\cdb*{ans.303}\\
   &\cdb*{ans.304}\\
   &\cdb*{ans.305}
\end{align*}
```

$$
\begin{aligned}
\texttt{ans.301} &:= 4\\
\texttt{ans.302} &:= \infty\\
\texttt{ans.303} &:= 2x\\
\texttt{ans.304} &:= \frac{4}{3}\\
\texttt{ans.305} &:= e^{a}
\end{aligned}
$$

Series

```
ans = series((1 + x)**(-2), x, 1, 6)        # cdb (ans.401,ans)
ans = series(exp(x), x, 0, 6)               # cdb (ans.402,ans)
ans = Sum(1/n**2, (n,1,50)).doit()          # cdb (ans.403,ans)
ans = Sum(1/n**4, (n,1,oo)).doit()          # cdb (ans.404,ans)
```

```
\begin{align*}
   &\cdb*{ans.401}\\
   &\cdb*{ans.402}\\
   &\cdb*{ans.403}\\
   &\cdb*{ans.404}
\end{align*}
```

$$
\begin{aligned}
\texttt{ans.401} &:= \frac{1}{2} + \frac{3\left(x-1\right)^{2}}{16} - \frac{\left(x-1\right)^{3}}{8} + \frac{5\left(x-1\right)^{4}}{64} - \frac{3\left(x-1\right)^{5}}{64} - \frac{x}{4} + O\left(\left(x-1\right)^{6}; x \to 1\right)\\
\texttt{ans.402} &:= 1 + x + \frac{x^{2}}{2} + \frac{x^{3}}{6} + \frac{x^{4}}{24} + \frac{x^{5}}{120} + O\left(x^{6}\right)\\
\texttt{ans.403} &:= \frac{3121579929551692678469635660835626209661709}{1920815367859463099600511526151929560192000}\\
\texttt{ans.404} &:= \frac{\pi^{4}}{90}
\end{aligned}
$$

Calculus

This example shows how `\Cdb` can be used to set the equation tag on the far right hand side.

```
ans = diff(x*sin(x),x)                                  # cdb (ans.501,ans)
ans = diff(x*sin(x),x).subs(x,pi/4)                     # cdb (ans.502,ans)
ans = integrate(2*sin(x)**2, (x,a,b))                   # cdb (ans.503,ans)
ans = Integral(2*exp(-x**2), (x,0,oo))                  # cdb (lhs.504,ans)
ans = ans.doit()                                        # cdb (ans.504,ans)
ans = Integral(Integral(x**2 + y**2, (y,0,x)), (x,0,1)) # cdb (lhs.505,ans)
ans = ans.doit()                                        # cdb (ans.505,ans)
```

```
\begin{align*}
   &\cdb*{ans.501}\\
   &\cdb*{ans.502}\\
   &\cdb*{ans.503}\\
    \cdb{lhs.504}&=\Cdb{ans.504}\\
    \cdb{lhs.505}&=\Cdb{ans.505}
\end{align*}
```

$$\texttt{ans.501} := x\cos(x) + \sin(x)$$

$$\texttt{ans.502} := \frac{\sqrt{2}\pi}{8} + \frac{\sqrt{2}}{2}$$

$$\texttt{ans.503} := -a + b + \sin(a)\cos(a) - \sin(b)\cos(b)$$

$$\int_0^\infty 2e^{-x^2}\,dx = \sqrt{\pi} \tag{ans.504}$$

$$\int_0^1\int_0^x \left(x^2+y^2\right)\,dy\,dx = \frac{1}{3} \tag{ans.505}$$

Differential equations

```
y = Function('y')
C1, C2 = symbols('C1 C2')

ode = Eq(y(x).diff(x) + y(x), 2*a*sin(x))
sol = expand(dsolve(ode,y(x)).rhs)                          # cdb (ans.601,sol)
cst = solve([sol.subs(x,0)],dict=True)
sol = sol.subs(cst[0])                                      # cdb (ans.602,sol)

ode = Eq(y(x).diff(x,2) + y(x), 0)
sol = expand(dsolve(ode,y(x)).rhs)                          # cdb (ans.603,sol)
cst = solve([sol.subs(x,0),sol.diff(x).subs(x,0)-1],dict=True)
sol = sol.subs(cst[0])                                      # cdb (ans.604,sol)

ode = Eq(y(x).diff(x,2) + 5*y(x).diff(x) - 6*y(x), 0)
sol = expand(dsolve(ode,y(x)).rhs)                          # cdb (ans.605,sol)
sol = sol.subs({C1:2,C2:3})                                 # cdb (ans.606,sol)
```

```
\begin{align*}
   &\cdb*{ans.601}\\
   &\cdb*{ans.602}\\
   &\cdb*{ans.603}\\
   &\cdb*{ans.604}\\
   &\cdb*{ans.605}\\
   &\cdb*{ans.606}
\end{align*}
```

$$
\begin{aligned}
\texttt{ans.601} &:= C_1 e^{-x} + a\sin(x) - a\cos(x)\\
\texttt{ans.602} &:= a\sin(x) - a\cos(x) + ae^{-x}\\
\texttt{ans.603} &:= C_1\sin(x) + C_2\cos(x)\\
\texttt{ans.604} &:= \sin(x)\\
\texttt{ans.605} &:= C_1 e^{-6x} + C_2 e^{x}\\
\texttt{ans.606} &:= 3e^{x} + 2e^{-6x}
\end{aligned}
$$

Curvature of a 2-sphere

This examples uses standard methods to compute the scalar curvature of a 2-sphere.

```
{\theta, \varphi}::Coordinate.
{\alpha, \beta, \gamma, \delta, \rho, \sigma, \mu, \nu, \lambda}::Indices(values={\varphi, \theta}, position=independent).

\partial{#}::PartialDerivative.

g_{\alpha\beta}::Metric.
g^{\alpha\beta}::InverseMetric.

Chr := \Gamma^{\alpha}_{\mu\nu} -> 1/2 g^{\alpha\beta} (  \partial_{\nu}{g_{\beta\mu}}
                                                        + \partial_{\mu}{g_{\beta\nu}}
                                                        - \partial_{\beta}{g_{\mu\nu}} ).

Rabcd := R^{\rho}_{\sigma\mu\nu} ->   \partial_{\mu}{\Gamma^{\rho}_{\sigma\nu}}
                                    - \partial_{\nu}{\Gamma^{\rho}_{\sigma\mu}}
                                    + \Gamma^{\rho}_{\beta\mu} \Gamma^{\beta}_{\sigma\nu}
                                    - \Gamma^{\rho}_{\beta\nu} \Gamma^{\beta}_{\sigma\mu}.

Rab := R_{\sigma\nu} -> R^{\rho}_{\sigma\rho\nu}.

R := R -> R_{\sigma\nu} g^{\sigma\nu}.

gab:={ g_{\theta\theta}   = r**2,
       g_{\varphi\varphi} = r**2 \sin(\theta)**2 }.   # cdb(gab,gab)

complete   (gab, $g^{\alpha\beta}$)                   # cdb(iab,gab)

evaluate   (Chr, gab, rhsonly=True)                   # cdb(Chr,Chr)

substitute (Rabcd, Chr)
evaluate   (Rabcd, gab, rhsonly=True)                 # cdb(Rabcd,Rabcd)

substitute (Rab, Rabcd)
evaluate   (Rab, gab, rhsonly=True)                   # cdb(Rab,Rab)
```

```
substitute (R, Rab)
evaluate   (R, gab, rhsonly=True)                         # cdb(R,R)
```

$$\left[g_{\theta\theta} = r^{2},\; g_{\varphi\varphi} = r^{2}(\sin\theta)^{2},\; g^{\varphi\varphi} = \left(r^{2}(\sin\theta)^{2}\right)^{-1},\; g^{\theta\theta} = r^{-2}\right]$$

$$\Gamma^{\alpha}{}_{\mu\nu} \to \square_{\mu\nu}{}^{\alpha} \begin{cases} \square_{\varphi\theta}{}^{\varphi} = (\tan\theta)^{-1} \\ \square_{\theta\varphi}{}^{\varphi} = (\tan\theta)^{-1} \\ \square_{\varphi\varphi}{}^{\theta} = -\frac{1}{2}\sin 2\theta \end{cases}$$

$$R^{\rho}{}_{\sigma\mu\nu} \to \square_{\sigma\nu}{}^{\rho}{}_{\mu} \begin{cases} \square_{\varphi\varphi}{}^{\theta}{}_{\theta} = \frac{1}{2}\sin 2\theta(\tan\theta)^{-1} - \cos 2\theta \\ \square_{\theta\varphi}{}^{\varphi}{}_{\theta} = -1 \\ \square_{\varphi\theta}{}^{\theta}{}_{\varphi} = -\frac{1}{2}\sin 2\theta(\tan\theta)^{-1} + \cos 2\theta \\ \square_{\theta\theta}{}^{\varphi}{}_{\varphi} = 1 \end{cases}$$

$$R_{\sigma\nu} \to \square_{\sigma\nu} \begin{cases} \square_{\varphi\varphi} = \frac{1}{2}\sin 2\theta(\tan\theta)^{-1} - \cos 2\theta \\ \square_{\theta\theta} = 1 \end{cases}$$

$$R \to 2r^{-2}$$

```
\begin{align*}
   &\cdb{iab}\\[10pt]
   &\cdb{Chr}\\[10pt]
   &\cdb{Rabcd}\\[10pt]
   &\cdb{Rab}\\[10pt]
   &\cdb{R}
\end{align*}
```

Symmetry of the Ricci tensor

This simple example shows that, for the metric connection, the Ricci tensor is symmetric, that is $R_{ab} = R_{ba}$.

```
{a,b,c,d,e,f,i,j,k,l,m,n,o,p,q,r,s,t,u#}::Indices(position=independent).

\partial{#}::PartialDerivative;

g_{a b}::Symmetric;
g^{a b}::Symmetric;

g_{a b}::Depends(\partial{#});
g^{a b}::Depends(\partial{#});

dgab := \partial_{c}{g^{a b}} -> - g^{a e} g^{b f} \partial_{c}{g_{e f}}. # cdb (dgab,dgab)

Gamma := \Gamma^{a}_{b c} ->
         (1/2) g^{a e} (   \partial_{b}{g_{e c}}
                         + \partial_{c}{g_{b e}}
                         - \partial_{e}{g_{b c}}).     # cdb (Chr,Gamma)

Rabcd := R^{a}_{b c d} ->
        \partial_{c}{\Gamma^{a}_{b d}} + \Gamma^{a}_{e c} \Gamma^{e}_{b d}
      - \partial_{d}{\Gamma^{a}_{b c}} - \Gamma^{a}_{e d} \Gamma^{e}_{b c}.
                                                       # cdb (Rabcd,Rabcd)

Rab := R_{a b} -> R^{c}_{a c b}.                       # cdb (Rab,Rab)

eqn := 2 (R_{a b} - R_{b a}).

substitute (eqn, Rab)
substitute (eqn, Rabcd)
substitute (eqn, Gamma)

distribute   (eqn)
product_rule (eqn)
canonicalise (eqn)                                     # cdb (final1,eqn)
```

```
substitute   (eqn,dgab)
canonicalise (eqn)                                    # cdb (final2,eqn)
```

Symmetry of the Ricci tensor

$$\begin{aligned}
g^{ab}{}_{,c} &:= \partial_{c}g^{ab} \rightarrow -g^{ae}g^{bf}\partial_{c}g_{ef}\\
\Gamma^{a}_{bc} &:= \Gamma^{a}{}_{bc} \rightarrow \frac{1}{2}g^{ae}\left(\partial_{b}g_{ec} + \partial_{c}g_{be} - \partial_{e}g_{bc}\right)\\
R^{a}{}_{bcd} &:= R^{a}{}_{bcd} \rightarrow \partial_{c}\Gamma^{a}{}_{bd} + \Gamma^{a}{}_{ec}\Gamma^{e}{}_{bd} - \partial_{d}\Gamma^{a}{}_{bc} - \Gamma^{a}{}_{ed}\Gamma^{e}{}_{bc}\\
R_{ab} &:= R_{ab} \rightarrow R^{c}{}_{acb}\\[10pt]
2(R_{ab} - R_{ba}) &= -\partial_{b}g^{ce}\partial_{a}g_{ce} + \partial_{a}g^{ce}\partial_{b}g_{ce}\\
&= 0
\end{aligned}$$

```
\begin{align*}
   g^{a b}{}_{,c}     &:=\cdb{dgab}\\
   \Gamma^{a}_{bc}    &:=\cdb{Chr}\\
   R^a{}_{bcd}        &:=\cdb{Rabcd}\\
   R_{ab}             &:=\cdb{Rab}\\[10pt]
   2(R_{ab}-R_{ba})   &= \cdb{final1}\\
                      &= \cdb{final2}
\end{align*}
```

The metric connection

This is a very standard computation that shows if

$$\Gamma^{a}_{bc} = \frac{1}{2} g^{ad}(\partial_b g_{dc} + \partial_c g_{bd} - \partial_d g_{bc}) \tag{1}$$

then

$$g_{ab;c} = 0. \tag{2}$$

This example might well be regarded as the Cadabra counterpart to the familiar *Hello World* program of undergraduate programming classes.

```
{a,b,c,d,e,f,h,i,j,k,l,m,n,o,p,q,r,s,t,u#}::Indices.

g_{a b}::Metric.
g_{a}^{b}::KroneckerDelta.

\partial_{#}::PartialDerivative.

cderiv:=\partial_{c}{g_{a b}} - g_{a d}\Gamma^{d}_{b c}
                              - g_{d b}\Gamma^{d}_{a c}.             # cdb (term31,cderiv)

Gamma:=\Gamma^{a}_{b c} -> (1/2) g^{a d} (  \partial_{b}{g_{d c}}
                                          + \partial_{c}{g_{b d}}
                                          - \partial_{d}{g_{b c}} ). # cdb (term32,Gamma)

substitute          (cderiv,Gamma);     # cdb (term33,cderiv)
distribute          (cderiv)            # cdb (term34,cderiv)
eliminate_metric    (cderiv)            # cdb (term35,cderiv)
eliminate_kronecker (cderiv)            # cdb (term36,cderiv)
canonicalise        (cderiv)            # cdb (term37,cderiv)
```

The metric connection

$$\texttt{term31} := \partial_{c}g_{ab} - g_{ad}\Gamma^{d}\,_{bc} - g_{db}\Gamma^{d}\,_{ac}$$

$$\texttt{term32} := \Gamma^{a}\,_{bc} \rightarrow \frac{1}{2}g^{ad}\left(\partial_{b}g_{dc} + \partial_{c}g_{bd} - \partial_{d}g_{bc}\right)$$

$$\texttt{term33} := \partial_{c}g_{ab} - \frac{1}{2}g_{ad}g^{de}\left(\partial_{b}g_{ec} + \partial_{c}g_{be} - \partial_{e}g_{bc}\right) - \frac{1}{2}g_{db}g^{de}\left(\partial_{a}g_{ec} + \partial_{c}g_{ae} - \partial_{e}g_{ac}\right)$$

$$\texttt{term34} := \partial_{c}g_{ab} - \frac{1}{2}g_{ad}g^{de}\partial_{b}g_{ec} - \frac{1}{2}g_{ad}g^{de}\partial_{c}g_{be} + \frac{1}{2}g_{ad}g^{de}\partial_{e}g_{bc} - \frac{1}{2}g_{db}g^{de}\partial_{a}g_{ec} - \frac{1}{2}g_{db}g^{de}\partial_{c}g_{ae} + \frac{1}{2}g_{db}g^{de}\partial_{e}g_{ac}$$

$$\texttt{term35} := \partial_{c}g_{ab} - \frac{1}{2}g_{a}\,^{e}\partial_{b}g_{ec} - \frac{1}{2}g_{a}\,^{e}\partial_{c}g_{be} + \frac{1}{2}g_{a}\,^{e}\partial_{e}g_{bc} - \frac{1}{2}g_{b}\,^{e}\partial_{a}g_{ec} - \frac{1}{2}g_{b}\,^{e}\partial_{c}g_{ae} + \frac{1}{2}g_{b}\,^{e}\partial_{e}g_{ac}$$

$$\texttt{term36} := \frac{1}{2}\partial_{c}g_{ab} - \frac{1}{2}\partial_{c}g_{ba}$$

$$\texttt{term37} := 0$$

```
\begin{align*}
   &\cdb*{term31}\\
   &\cdb*{term32}\\
   &\cdb*{term33}\\
   &\cdb*{term34}\\
   &\cdb*{term35}\\
   &\cdb*{term36}\\
   &\cdb*{term37}
\end{align*}
```

The metric connection

This is a very standard computation that shows if

$$\Gamma^{a}_{bc} = \frac{1}{2} g^{ad}(\partial_b g_{dc} + \partial_c g_{bd} - \partial_d g_{bc}) \tag{1}$$

then

$$g_{ab;c} = 0. \tag{2}$$

This example might well be regarded as the Cadabra counterpart to the familiar *Hello World* program of undergraduate programming classes.

```
{a,b,c,d,e,f,h,i,j,k,l,m,n,o,p,q,r,s,t,u#}::Indices.

g_{a b}::Metric.
g_{a}^{b}::KroneckerDelta.

\partial_{#}::PartialDerivative.

cderiv:=\partial_{c}{g_{a b}} - g_{a d}\Gamma^{d}_{b c}
                              - g_{d b}\Gamma^{d}_{a c}.             # cdb (term31,cderiv)

Gamma:=\Gamma^{a}_{b c} -> (1/2) g^{a d} (  \partial_{b}{g_{d c}}
                                          + \partial_{c}{g_{b d}}
                                          - \partial_{d}{g_{b c}} ). # cdb (term32,Gamma)

substitute          (cderiv,Gamma);     # cdb (term33,cderiv)
distribute          (cderiv)            # cdb (term34,cderiv)
eliminate_metric    (cderiv)            # cdb (term35,cderiv)
eliminate_kronecker (cderiv)            # cdb (term36,cderiv)
canonicalise        (cderiv)            # cdb (term37,cderiv)
```

The metric connection

$$\texttt{term31} := \partial_{c}g_{ab} - g_{ad}\Gamma^{d}{}_{bc} - g_{db}\Gamma^{d}{}_{ac}$$

$$\texttt{term32} := \Gamma^{a}{}_{bc} \rightarrow \frac{1}{2}g^{ad}\left(\partial_{b}g_{dc} + \partial_{c}g_{bd} - \partial_{d}g_{bc}\right)$$

$$\texttt{term33} := \partial_{c}g_{ab} - \frac{1}{2}g_{ad}g^{de}\left(\partial_{b}g_{ec} + \partial_{c}g_{be} - \partial_{e}g_{bc}\right) - \frac{1}{2}g_{db}g^{de}\left(\partial_{a}g_{ec} + \partial_{c}g_{ae} - \partial_{e}g_{ac}\right)$$

$$\texttt{term34} := \partial_{c}g_{ab} - \frac{1}{2}g_{ad}g^{de}\partial_{b}g_{ec} - \frac{1}{2}g_{ad}g^{de}\partial_{c}g_{be} + \frac{1}{2}g_{ad}g^{de}\partial_{e}g_{bc} - \frac{1}{2}g_{db}g^{de}\partial_{a}g_{ec} - \frac{1}{2}g_{db}g^{de}\partial_{c}g_{ae} + \frac{1}{2}g_{db}g^{de}\partial_{e}g_{ac}$$

$$\texttt{term35} := \partial_{c}g_{ab} - \frac{1}{2}g_{a}{}^{e}\partial_{b}g_{ec} - \frac{1}{2}g_{a}{}^{e}\partial_{c}g_{be} + \frac{1}{2}g_{a}{}^{e}\partial_{e}g_{bc} - \frac{1}{2}g_{b}{}^{e}\partial_{a}g_{ec} - \frac{1}{2}g_{b}{}^{e}\partial_{c}g_{ae} + \frac{1}{2}g_{b}{}^{e}\partial_{e}g_{ac}$$

$$\texttt{term36} := \frac{1}{2}\partial_{c}g_{ab} - \frac{1}{2}\partial_{c}g_{ba}$$

$$\texttt{term37} := 0$$

```
\begin{align*}
   &\cdb*{term31}\\
   &\cdb*{term32}\\
   &\cdb*{term33}\\
   &\cdb*{term34}\\
   &\cdb*{term35}\\
   &\cdb*{term36}\\
   &\cdb*{term37}
\end{align*}
```

The metric connection in Riemann normal coordinates

In local Riemann normal coordinates, the metric components can always be expanded as a power series in the Riemann curvatures and its derivatives (provided the curvatives are finite at the expansion point). In particular

$$
\begin{aligned}
g_{ab}(x) &= g_{ab} - \frac{1}{3}x^c x^d R_{acbd} - \frac{1}{6}x^c x^d x^e \nabla_c R_{adbe} + \cdots \\
g^{ab}(x) &= g^{ab} + \frac{1}{3}x^c x^d g^{ae} g^{bf} R_{cedf} + \frac{1}{6}x^c x^d x^e g^{af} g^{bg} \nabla_c R_{dfeg} + \cdots
\end{aligned}
$$

where g_{ab} and g^{ab} are independent of the coordinates x^a and where ∇ is the metric compatable derivative operator (i.e., $\nabla(g) = 0$). In applications in General Relativity the g_{ab} are often chosen to be $g_{ab} = \mathrm{diag}(-1, 1, 1, 1)$.

Here we will use the standard metric compatible connection

$$
\Gamma^{d}_{ab}(x) = \frac{1}{2} g^{dc} \left(g_{cb,a} + g_{ac,b} - g_{ab,c} \right)
$$

to compute $\Gamma^{d}_{ab}(x)$ to terms linear in R_{abcd} and $\nabla_e R_{abcd}$.

```
{a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w#}::Indices(position=independent).

D{#}::PartialDerivative.
\nabla{#}::Derivative.

g_{a b}::Metric.
g^{a b}::InverseMetric.

\delta{#}::KroneckerDelta.

R_{a b c d}::RiemannTensor.

x^{a}::Depends(D{#}).
x^{a}::Weight(label=num,value=1).

R_{a b c d}::Depends(\nabla{#}).

DxaDxb := D_{a}{x^{b}}->\delta^{b}_{a}.

```

```
# can chose lower order approximations by truncating the following pair

gab := g_{a b} - 1/3 x^{c} x^{d} R_{a c b d}
               - 1/6 x^{c} x^{d} x^{e} \nabla_{c}{R_{a d b e}}.                   # cdb(gab.000,gab)

iab := g^{a b} + 1/3 x^{c} x^{d} g^{a e} g^{b f} R_{c e d f}
               + 1/6 x^{c} x^{d} x^{e} g^{a f} g^{b g} \nabla_{c}{R_{d f e g}}.    # cdb(iab.000,iab)

gab := g_{a b} -> @(gab).
iab := g^{a b} -> @(iab).

gam := 1/2 g^{d c} (D_{a}{g_{c b}} + D_{b}{g_{a c}} - D_{c}{g_{a b}}).             # cdb(gam.001,gam)

substitute   (gam,gab)
substitute   (gam,iab)
distribute   (gam)              # cdb(gam.002,gam)
unwrap       (gam)              # cdb(gam.003,gam)
product_rule (gam)              # cdb(gam.004,gam)
distribute   (gam)              # cdb(gam.005,gam)
substitute   (gam,DxaDxb)       # cdb(gam.006,gam)
eliminate_kronecker (gam)       # cdb(gam.007,gam)
sort_product   (gam)            # cdb(gam.008,gam)
rename_dummies (gam)            # cdb(gam.009,gam)
canonicalise   (gam)            # cdb(gam.010,gam)

def truncate (obj,n):

    ans = Ex(0)  # create a Cadabra object with value zero

    for i in range (0,n+1):
       foo := @(obj).
       bah  = Ex("num = " + str(i))
       distribute  (foo)
       keep_weight (foo, bah)
       ans = ans + foo

    return ans

```

```
gam = truncate (gam,2)   # cdb (gam.101,gam)  # allow up to 2nd order in x^a

# =========================================================================
# the remaining code is just for pretty printing

{x^{a},g^{a b},R_{a b c d},\nabla_{e}{R_{a b c d}}}::SortOrder.

def get_term (obj,n):

    foo := @(obj).
    bah  = Ex("num = " + str(n))
    distribute  (foo)
    keep_weight (foo, bah)

    return foo

def reformat (obj,scale):

   foo  = Ex(str(scale))
   bah := @(foo) @(obj).

   distribute     (bah)
   sort_product   (bah)
   rename_dummies (bah)
   canonicalise   (bah)
   factor_out     (bah,$x^{a?},g^{b? c?}$)
   ans := @(bah) / @(foo).

   return ans

gam1 = get_term (gam,1)       # cdb (gam1.301,gam1)
gam2 = get_term (gam,2)       # cdb (gam2.301,gam2)

gam1 = reformat (gam1,  3)    # cdb (gam1.301,gam1)
gam2 = reformat (gam2, 12)    # cdb (gam2.301,gam2)

Gamma  := @(gam1) + @(gam2).  # cdb (Gamma.301,Gamma)
Scaled := 12 @(Gamma).        # cdb (Scaled.301,Scaled)
```

The metric connection in Riemann normal coordinates

$$\Gamma^{d}_{ab} = \frac{1}{3}x^{c}g^{de}\left(R_{aebc} + R_{acbe}\right) + \frac{1}{12}x^{c}x^{e}g^{df}\left(\nabla_{a}R_{bcef} + 2\nabla_{c}R_{afbe} + \nabla_{b}R_{acef} + 2\nabla_{c}R_{aebf} + \nabla_{f}R_{acbe}\right)$$

$$12\Gamma^{d}_{ab} = 4x^{c}g^{de}\left(R_{aebc} + R_{acbe}\right) + x^{c}x^{e}g^{df}\left(\nabla_{a}R_{bcef} + 2\nabla_{c}R_{afbe} + \nabla_{b}R_{acef} + 2\nabla_{c}R_{aebf} + \nabla_{f}R_{acbe}\right)$$

```
\begin{dgroup*}
   \begin{dmath*} \Gamma^{d}_{a b} = \cdb{Gamma.301} \end{dmath*}
\end{dgroup*}

\begin{dgroup*}
   \begin{dmath*} 12 \Gamma^{d}_{a b} = \cdb{Scaled.301} \end{dmath*}
\end{dgroup*}
```

The metric connection in Riemann normal coordinates

In local Riemann normal coordinates, the metric components can always be expanded as a power series in the Riemann curvatures and its derivatives (provided the curvatives are finite at the expansion point). In particular

$$
\begin{aligned}
g_{ab}(x) &= g_{ab} - \frac{1}{3}x^c x^d R_{acbd} - \frac{1}{6}x^c x^d x^e \nabla_c R_{adbe} + \cdots \\
g^{ab}(x) &= g^{ab} + \frac{1}{3}x^c x^d g^{ae} g^{bf} R_{cedf} + \frac{1}{6}x^c x^d x^e g^{af} g^{bg} \nabla_c R_{dfeg} + \cdots
\end{aligned}
$$

where g_{ab} and g^{ab} are independent of the coordinates x^a and where ∇ is the metric compatable derivative operator (i.e., $\nabla(g) = 0$). In applications in General Relativity the g_{ab} are often chosen to be $g_{ab} = \mathrm{diag}(-1,1,1,1)$.

Here we will use the standard metric compatible connection

$$\Gamma^d_{ab}(x) = \frac{1}{2} g^{dc} \left(g_{cb,a} + g_{ac,b} - g_{ab,c} \right)$$

to compute $\Gamma^d_{ab}(x)$ to terms linear in R_{abcd} and $\nabla_e R_{abcd}$.

```
{a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w#}::Indices(position=independent).

D{#}::PartialDerivative.
\nabla{#}::Derivative.

g_{a b}::Metric.
g^{a b}::InverseMetric.

\delta{#}::KroneckerDelta.

R_{a b c d}::RiemannTensor.

x^{a}::Depends(D{#}).
x^{a}::Weight(label=num,value=1).

R_{a b c d}::Depends(\nabla{#}).

DxaDxb := D_{a}{x^{b}}->\delta^{b}_{a}.

```

```
# can chose lower order approximations by truncating the following pair

gab := g_{a b} - 1/3 x^{c} x^{d} R_{a c b d}
               - 1/6 x^{c} x^{d} x^{e} \nabla_{c}{R_{a d b e}}.                    # cdb(gab.000,gab)

iab := g^{a b} + 1/3 x^{c} x^{d} g^{a e} g^{b f} R_{c e d f}
               + 1/6 x^{c} x^{d} x^{e} g^{a f} g^{b g} \nabla_{c}{R_{d f e g}}.    # cdb(iab.000,iab)

gab := g_{a b} -> @(gab).
iab := g^{a b} -> @(iab).

gam := 1/2 g^{d c} (D_{a}{g_{c b}} + D_{b}{g_{a c}} - D_{c}{g_{a b}}).             # cdb(gam.001,gam)

substitute   (gam,gab)
substitute   (gam,iab)
distribute   (gam)             # cdb(gam.002,gam)
unwrap       (gam)             # cdb(gam.003,gam)
product_rule (gam)             # cdb(gam.004,gam)
distribute   (gam)             # cdb(gam.005,gam)
substitute   (gam,DxaDxb)      # cdb(gam.006,gam)
eliminate_kronecker (gam)      # cdb(gam.007,gam)
sort_product   (gam)           # cdb(gam.008,gam)
rename_dummies (gam)           # cdb(gam.009,gam)
canonicalise   (gam)           # cdb(gam.010,gam)

def truncate (obj,n):

    ans = Ex(0)  # create a Cadabra object with value zero

    for i in range (0,n+1):
       foo := @(obj).
       bah  = Ex("num = " + str(i))
       distribute  (foo)
       keep_weight (foo, bah)
       ans = ans + foo

    return ans

```

```
gam = truncate (gam,2)   # cdb (gam.101,gam)  # allow up to 2nd order in x^a

# =========================================================================
# the remaining code is just for pretty printing

{x^{a},g^{a b},R_{a b c d},\nabla_{e}{R_{a b c d}}}::SortOrder.

def get_term (obj,n):

    foo := @(obj).
    bah  = Ex("num = " + str(n))
    distribute  (foo)
    keep_weight (foo, bah)

    return foo

def reformat (obj,scale):

   foo  = Ex(str(scale))
   bah := @(foo) @(obj).

   distribute     (bah)
   sort_product   (bah)
   rename_dummies (bah)
   canonicalise   (bah)
   factor_out     (bah,$x^{a?},g^{b? c?}$)
   ans := @(bah) / @(foo).

   return ans

gam1 = get_term (gam,1)       # cdb (gam1.301,gam1)
gam2 = get_term (gam,2)       # cdb (gam2.301,gam2)

gam1 = reformat (gam1,  3)    # cdb (gam1.301,gam1)
gam2 = reformat (gam2, 12)    # cdb (gam2.301,gam2)

Gamma  := @(gam1) + @(gam2).  # cdb (Gamma.301,Gamma)
Scaled := 12 @(Gamma).        # cdb (Scaled.301,Scaled)
```

The metric connection in Riemann normal coordinates

$$\Gamma^{d}_{ab} = \frac{1}{3}x^{c}g^{de}\left(R_{aebc} + R_{acbe}\right) + \frac{1}{12}x^{c}x^{e}g^{df}\left(\nabla_{a}R_{bcef} + 2\nabla_{c}R_{afbe} + \nabla_{b}R_{acef} + 2\nabla_{c}R_{aebf} + \nabla_{f}R_{acbe}\right)$$

$$12\Gamma^{d}_{ab} = 4x^{c}g^{de}\left(R_{aebc} + R_{acbe}\right) + x^{c}x^{e}g^{df}\left(\nabla_{a}R_{bcef} + 2\nabla_{c}R_{afbe} + \nabla_{b}R_{acef} + 2\nabla_{c}R_{aebf} + \nabla_{f}R_{acbe}\right)$$

```
\begin{dgroup*}
   \begin{dmath*} \Gamma^{d}_{a b} = \cdb{Gamma.301} \end{dmath*}
\end{dgroup*}

\begin{dgroup*}
   \begin{dmath*} 12 \Gamma^{d}_{a b} = \cdb{Scaled.301} \end{dmath*}
\end{dgroup*}
```