Calculus Tutorial

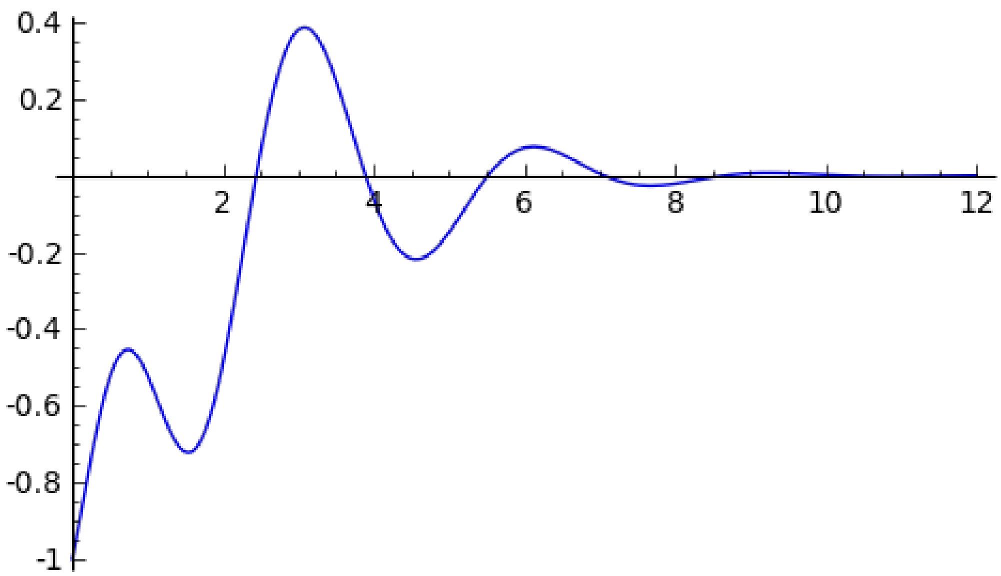
Limits

```
var("s")
f(s) = (exp(3*s)-1)/sin(4*s)
flim = limit(f(s), s=0); show(flim)
#g(s) = (s^3-3*s^2+6*s)/(7*s^3+9)
#glim = limit(g(s), s=infinity); show(glim)
```

$$\frac{3}{4}$$

Plot function of one variable

```
var("x")
f(x) = (x^2*cos(2*x)-1)*exp(-x)
g(x) = x^3/(1+x^6)
plotf = plot(f(x), (x,0,12), color='blue', linestyle='-', thickness=1)
show(plotf, figsize=[5,3])
# plotg = plot(g(x), (x,0,12), color='red', linestyle=':', thickness=1)
# plotall = plotf + plotg
# show(plotall, figsize=[5,3])
```



Differentiation

```
var("x")
f(x) = x^3*cos(4*exp(9*x))
fp(x) = diff(f(x),x); show(fp(x))
```

$$-36\,x^3e^{(9\,x)}\sin\left(4\,e^{(9\,x)}\right) + 3\,x^2\cos\left(4\,e^{(9\,x)}\right)$$

Partial differentiation

```
var("x,y")
f(x,y) = x^2*y^3 + exp(-3*x)
fpx(x,y) = diff(f(x,y),x); show(fpx(x,y))
fpy(x,y) = diff(f(x,y),y); show(fpy(x,y))
```

$$2\,xy^3 - 3\,e^{(-3\,x)}$$

$$3\,x^2y^2$$

Gradient

```
var("x,y,z")
g(x,y,z) = x^2 + y^2 - 4*exp(3*y*z^2)
gx = diff(g(x,y,z),x); gy = diff(g(x,y,z),y); gz = diff(g(x,y,z),z);
ggrad = vector([gx,gy,gz]); show(ggrad)
```

$$\left(2\,x,\, -12\,z^2e^{(3\,yz^2)} + 2\,y,\, -24\,yze^{(3\,yz^2)}\right)$$

Divergence

```
var("x,y,z")
g(x,y,z) = x^2 - 6*y^5 + cos(x*y*z)
gx(x,y,z) = diff(g(x,y,z),x); gy(x,y,z) = diff(g(x,y,z),y); gz(x,y,z) =
diff(g(x,y,z),z);
gdiv(x,y,z) = gx(x,y,z) + gy(x,y,z) + gz(x,y,z)
show(gdiv(x,y,z))
```

$$-30\,y^4 - xy\sin\left(xyz\right) - xz\sin\left(xyz\right) - yz\sin\left(xyz\right) + 2\,x$$

Curl

```
var("x,y,z")
g(x,y,z) = x^2 + y^3 + x*y*z
gx(x,y,z) = diff(g(x,y,z),x); gy(x,y,z) = diff(g(x,y,z),y); gz(x,y,z) =
diff(g(x,y,z),z);
```

```
gci = gz(x,y,z) - gy(x,y,z); gcj = gz(x,y,z) - gx(x,y,z); gck = gy(x,y,z)
- gx(x,y,z);
gcurl = vector([gci,-gcj,gck]); show(gcurl)
```

$$\left(xy - 3\,y^{2} - xz,\ -xy + yz + 2\,x,\ 3\,y^{2} + xz - yz - 2\,x\right)$$

Anti-differentiation

```
var("t")
a(t) = -exp(-3*t)*sin(2*t)
A(t) = integral(a(t),t); show(A(t))
# Alim1 = integral(a(t),t,0,4*pi); show (Alim1)
# Alim2 = integral(a(t),t,0,infinity); show (Alim2)
```

$$\frac{1}{13}\,(2\,\cos\left(2\,t\right) + 3\,\sin\left(2\,t\right))e^{\left(-3\,t\right)}$$

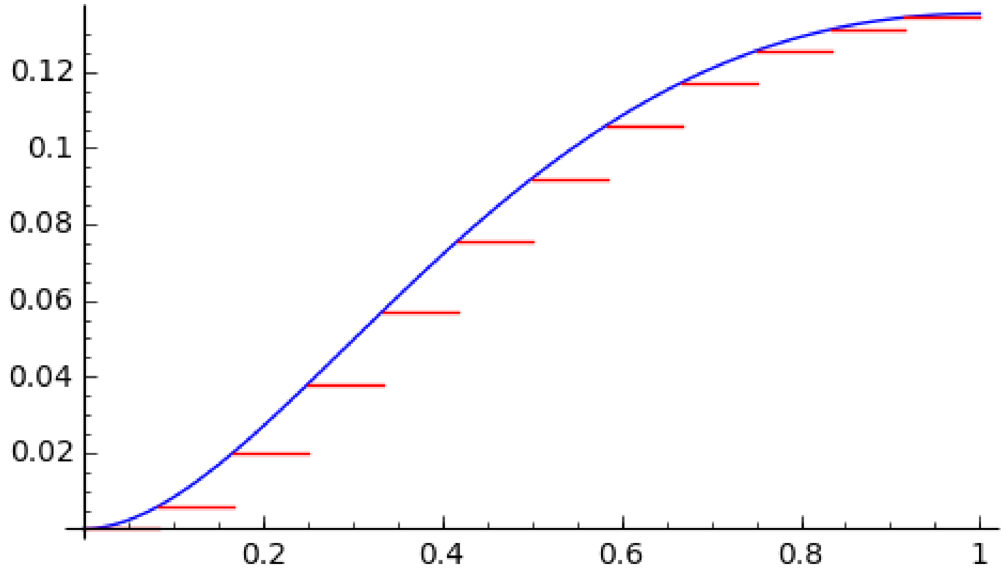
Numerical integration

Rectangle rules

```
var("x")
g(x) = x^2*exp(-2*x)                                    # function
a = 0; b = 1                                            # endpoints
n = 4                                                   # number of
subdivisions
h = (b-a)/n
grman = h*sum([g(a+i*h) for i in range(n)])             # left point
#grman = h*sum([g(a+(i+1)*h) for i in range(n)])         # right point
#grman = h*sum([g(a+(i+1/2)*h) for i in range(n)])       # midpoint
gexact = integral(g(x),x,a,b); error = abs(gexact-grman)
float(grman); float(error)
```

```
0.06384718540209798
0.016983710552136138
```

```
var("x")
g(x) = x^2*exp(-2*x)                             # function
a = 0; b = 1                                     # endpoints
n = 12                                           # number of subdivisions
gp = Piecewise([[(a,b),g]])
gpt = gp.riemann_sum(n)                          # left
#gpt = gp.riemann_sum(n, mode="right")            # right, midpoint
plotg = plot(g, (x,a,b), color='blue', linestyle='-', thickness=1)
plotgpt = gpt.plot(color='red', linestyle='-', thickness=1)
plotall = plotg + plotgpt
show(plotall, figsize=[5,3])
```

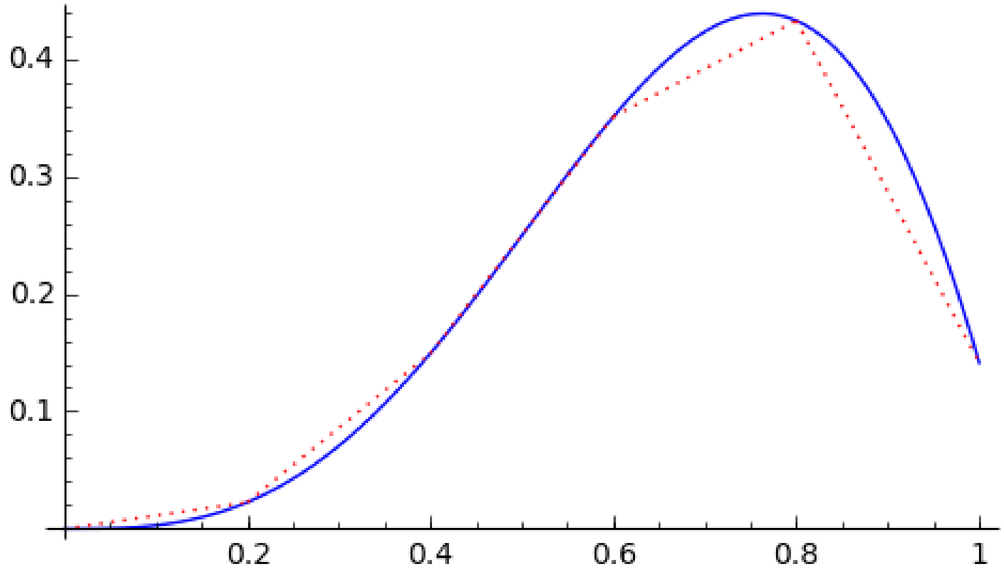


Trapezoid rule

```
var("x")
g(x) = x^2*sin(2*x)            # function
a = 0; b = 1                   # endpoints
n = 3                          # number of subdivisions
gp = Piecewise([[(a,b),g]])
gpt = gp.trapezoid(n)
gtrap = integral(gpt,x,a,b)
gexact = integral(g(x),x,a,b); error = abs(gexact-gtrap)
float(gtrap); float(error)
```

```
0.318442956638465
0.009757534088838588
```

```
var("x")
g(x) = x^2*sin(3*x)                                # function
a = 0; b = 1                                       # endpoints
n = 5                                              # number of subdivisions
gp = Piecewise([[(a,b),g]])
gpt = gp.trapezoid(n)
plotg = plot(g, (x,a,b), color='blue', linestyle='-', thickness=1)
plotgpt = gpt.plot(color='red', linestyle=':', thickness=1)
plotall = plotg + plotgpt
show(plotall, figsize=[5,3])
```



Simpson's rule

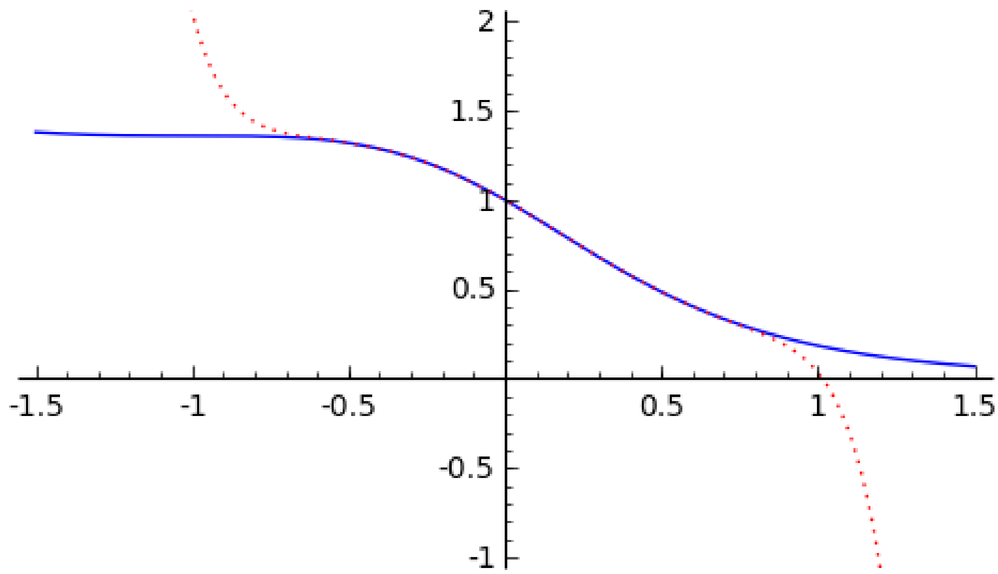
```
f(x) = 1/x                                                # function
a = 1; b = exp(1)                                         # endpoints
n = 3                                                     # number of subdivisions
is 2*n
h = (b-a)/(2*n)
vals = [f(a+i*h) for i in range(0,2*n+1)]
coeffs = [3-(-1)^i for i in range(0,2*n+1)]
coeffs[0] = 1; coeffs[2*n] = 1;
fsimp = h/3*vector(coeffs).dot_product(vector(vals))
fexact = integral(f(x),x,a,b); ferror = abs(fexact-fsimp)
float(fsimp); float(ferror)
```

```
1.000186876741404
0.00018687674140394073
```

Taylor polynomial

```
var("t")
f(t) = exp(-t)/(1+t^2)
napp = 9; center = 0
tend = 1.5                                                           # size of
interval to be plotted
ftay(t) = taylor(f(t),t,center,napp); show(ftay(t))                  # degree
napp, centered at t=center
plotf = plot(f(t), (t,-tend,tend), color='blue', linestyle='-',
thickness=1)
plotftay = plot(ftay(t), (t,-tend,tend), color='red', linestyle=':',
thickness=1)
plotall = plotf + plotftay
show(plotall, ymin=-1, ymax=2, figsize=[5,3])
```

$$-\frac{305353}{362880}\,t^{9}+\frac{4357}{8064}\,t^{8}+\frac{4241}{5040}\,t^{7}-\frac{389}{720}\,t^{6}-\frac{101}{120}\,t^{5}+\frac{13}{24}\,t^{4}+\frac{5}{6}\,t^{3}-\frac{1}{2}\,t^{2}-t+1$$



Double integral

```
var("x,y")
f(x,y) = x^4*y^8
Fx(x,y) = integral(f(x,y),x)
Fxy(x,y) = integral(Fx(x,y),y); show(Fxy(x,y))
```

$$\frac{1}{45}\,x^{5}y^{9}$$

Double integral with limits

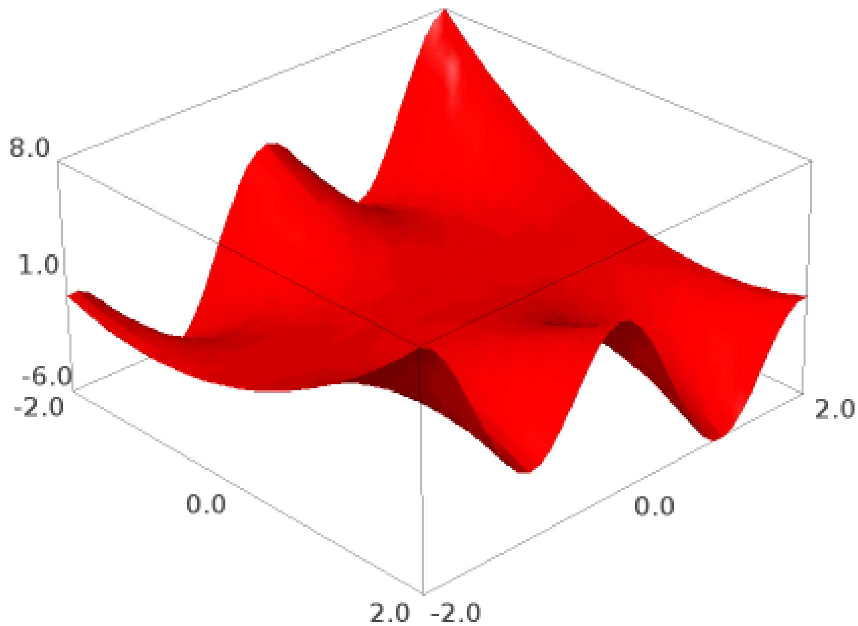
```
var("x,y")
f(x,y) = x^3*y^2
Fx(x,y) = integral(f(x,y),x,-y,4*y)
Fxy = integral(Fx(x,y),y,1,2); show(Fxy)
```

$$\frac{32385}{28}$$

Plot function of two variables

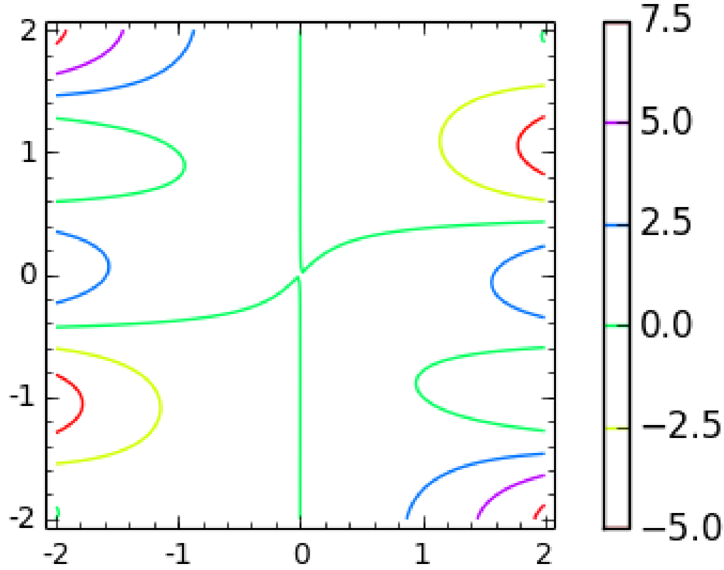
```
var("x,y")
```

```
f(x,y) = x^2*cos(pi*y) - x*y
plotf = plot3d(f(x,y), (x,-2,2), (y,-2,2), color='red')
show(plotf, figsize=[5,3])
```



Contour plot

```
var("x,y")
f(x,y) = x^2*cos(pi*y) - x*y
plotf = contour_plot(f(x,y), (x,-2,2), (y,-2,2), contours=6,
linestyles='-', linewidths=1, fill=False, cmap='hsv', colorbar=True)
show(plotf, figsize=[5,3])
```


2
1
0
-1
-2
-2
-1
0
1
2
7.5
5.0
2.5
0.0
−2.5
−5.0