```
>
```

Double Integrals

```
> restart:with(plots):with(plottools):
> setoptions3d(axes=NORMAL,labels=["x","y","z"],orientation=[20,
  70]);
```

For this worksheet we need the Student package.

```
> with(Student);
```

```
[Calculus1, LinearAlgebra, MultivariateCalculus, Precalculus, SetColors, VectorCalculus]
```

At first we need the Calculus1 subpackage.

```
> with(Calculus1);
```

```
[AntiderivativePlot, AntiderivativeTutor, ApproximateInt, ApproximateIntTutor, ArcLength,
    ArcLengthTutor, Asymptotes, Clear, CriticalPoints, CurveAnalysisTutor, DerivativePlot,
    DerivativeTutor, DiffTutor, ExtremePoints, FunctionAverage, FunctionAverageTutor,
    FunctionChart, FunctionPlot, GetMessage, GetNumProblems, GetProblem, Hint, InflectionPoints,
    IntTutor, Integrand, InversePlot, InverseTutor, LimitTutor, MeanValueTheorem,
    MeanValueTheoremTutor, NewtonQuotient, NewtonsMethod, NewtonsMethodTutor,
    PointInterpolation, RiemannSum, RollesTheorem, Roots, Rule, Show, ShowIncomplete, ShowSteps,
    Summand, SurfaceOfRevolution, SurfaceOfRevolutionTutor, Tangent, TangentSecantTutor,
    TangentTutor, TaylorApproximation, TaylorApproximationTutor, Understand, Undo,
    VolumeOfRevolution, VolumeOfRevolutionTutor, WhatProblem]
```

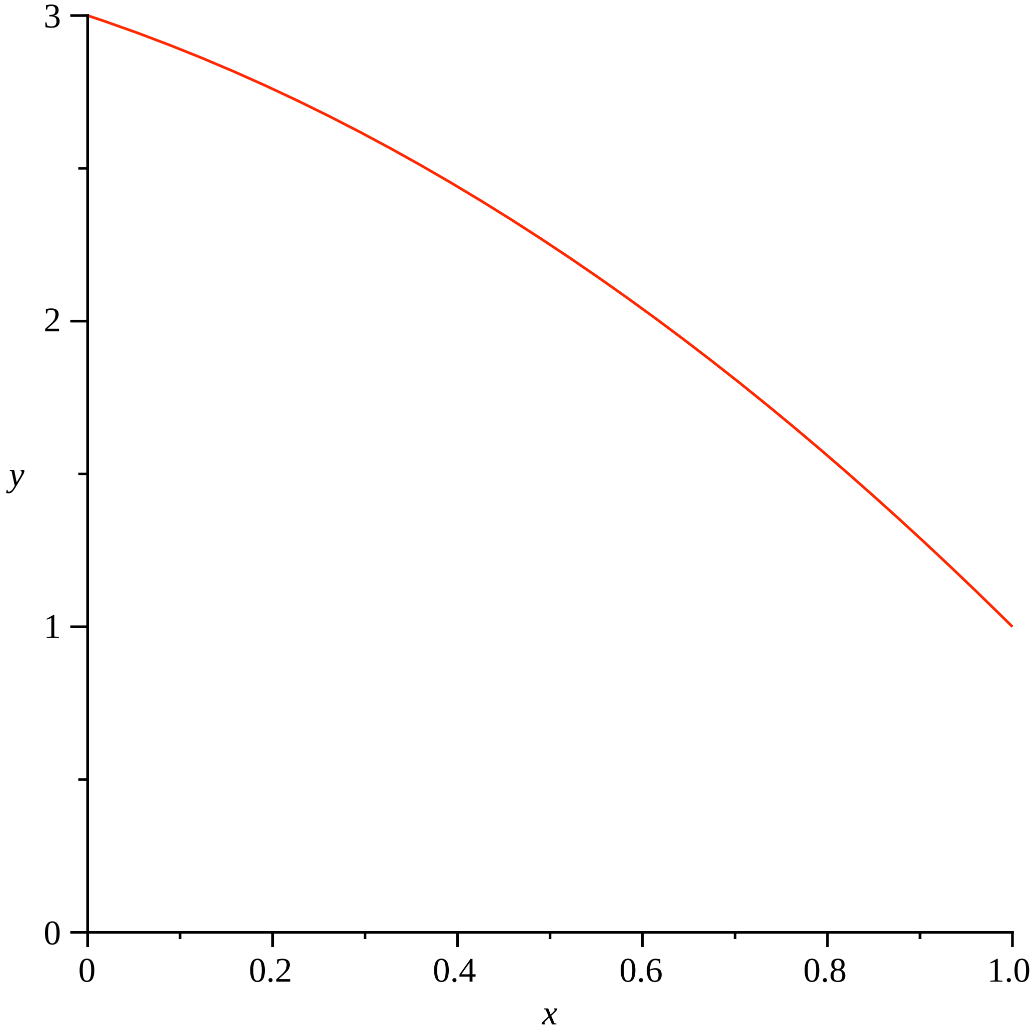
Integrals of Functions of One Variable.

To help us understand integrals of functions of two variables, we first reconsider functions of a single variable. We will use the decreasing function f(x) = $3 - x - x^2$ on the interval [0, 1]. We enter the function and plot its graph.

```
> f:=3-x-x^2;
```

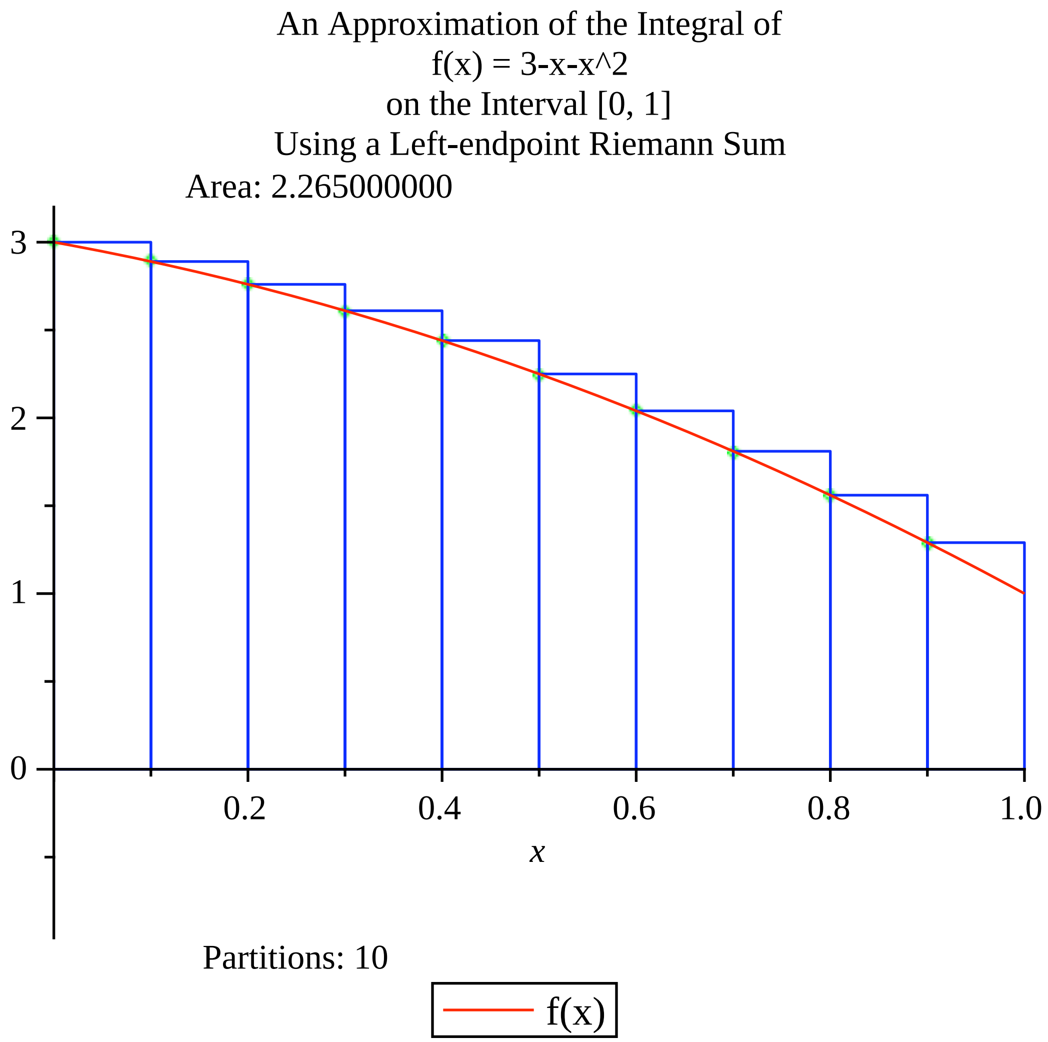
$$f := 3 - x - x^2$$

```
> plot(f,x=0..1,y=0..3);
```



Since this function is decreasing, the **left rule**, which uses the left hand endpoint of each interval to determine the height of each of n rectangles, gives us an upper sum that overestimates the integral. With $n = 10$, implemented by **partition=10**, we view this graphically by using the **method=left** option in ApproximateInt .

```
> ApproximateInt(f, x=0..1, method=left, partition=10, output=
  plot);
```



The integral is (over) approximated by the area of the 10 rectangles. The general formula is $\sum_{i=0}^{n-1} f(x_i)\ \Delta x$. In Maple, we implement this sum with the option **output=sum**. We then approximate the value of the integral and give it as decimal. Since the left rule gives an upper sum here, we can get the same result by using **method=upper**.

```
> left_sum_notation:=ApproximateInt(f, x=0..1, method=left,
  partition=10, output=sum);
  leftsum:=ApproximateInt(f, x=0..1, method=left, partition=10,
  output=value);
  leftsum:=evalf(leftsum);
  uppersum:=ApproximateInt(f, x=0..1, method=upper, partition=10,
  output=value);
```

$$\mathit{left\_sum\_notation} := \frac{1}{10}\sum_{i=0}^{9}\left(3 - \frac{1}{10}\,i - \frac{1}{100}\,i^2\right)$$

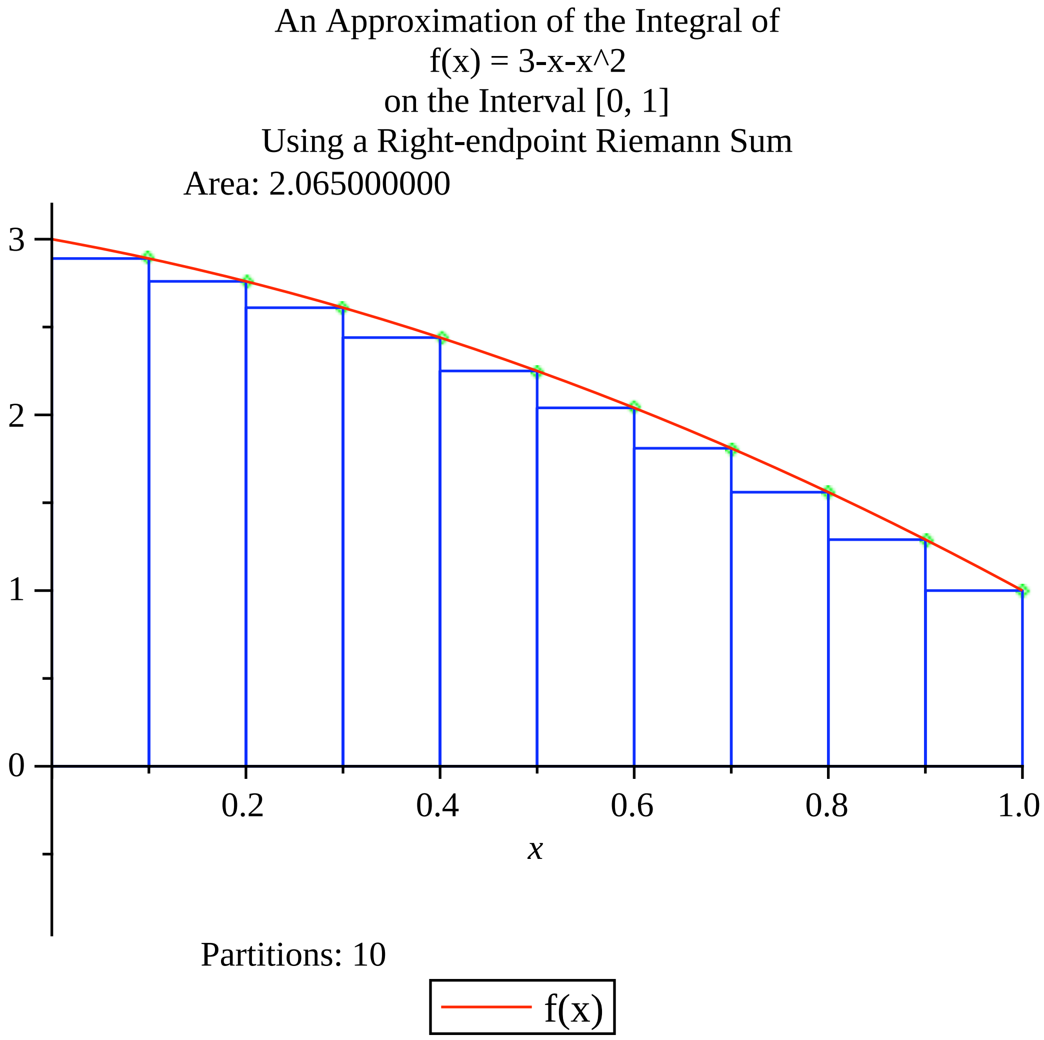
$$\mathit{leftsum} := \frac{453}{200}$$

$$\mathit{leftsum} := 2.265000000$$

$$\mathit{uppersum} := 2.265000000$$

Next we consider the **right rule**, where we use the right hand endpoint of each interval to determine the height of each of the n rectangles. Since we have a decreasing function, this should give us an underestimation. We again use $n = 10$, and use **method=right**.

```
> ApproximateInt(f, x=0..1, method=right, partition=10, output=
  plot);
```



The general formula is $\sum_{i=1}^{n} f(x_i)\,\Delta x$.. We again approximate this answer. Since the right rule gives an lower sum here, we can get the same result by using **method=lower**.

```
> right_sum_notation:=ApproximateInt(f, x=0..1, method=right,
  partition=10, output=sum);
  rightsum:=ApproximateInt(f, x=0..1, method=right, partition=10,
  output=value);
  rightsum:=evalf(rightsum);
  lowersum:=ApproximateInt(f, x=0..1, method=lower, partition=10,
  output=value);
```

$$right\_sum\_notation := \frac{1}{10} \sum_{i=1}^{10} \left(3 - \frac{1}{10}\, i - \frac{1}{100}\, i^2\right)$$

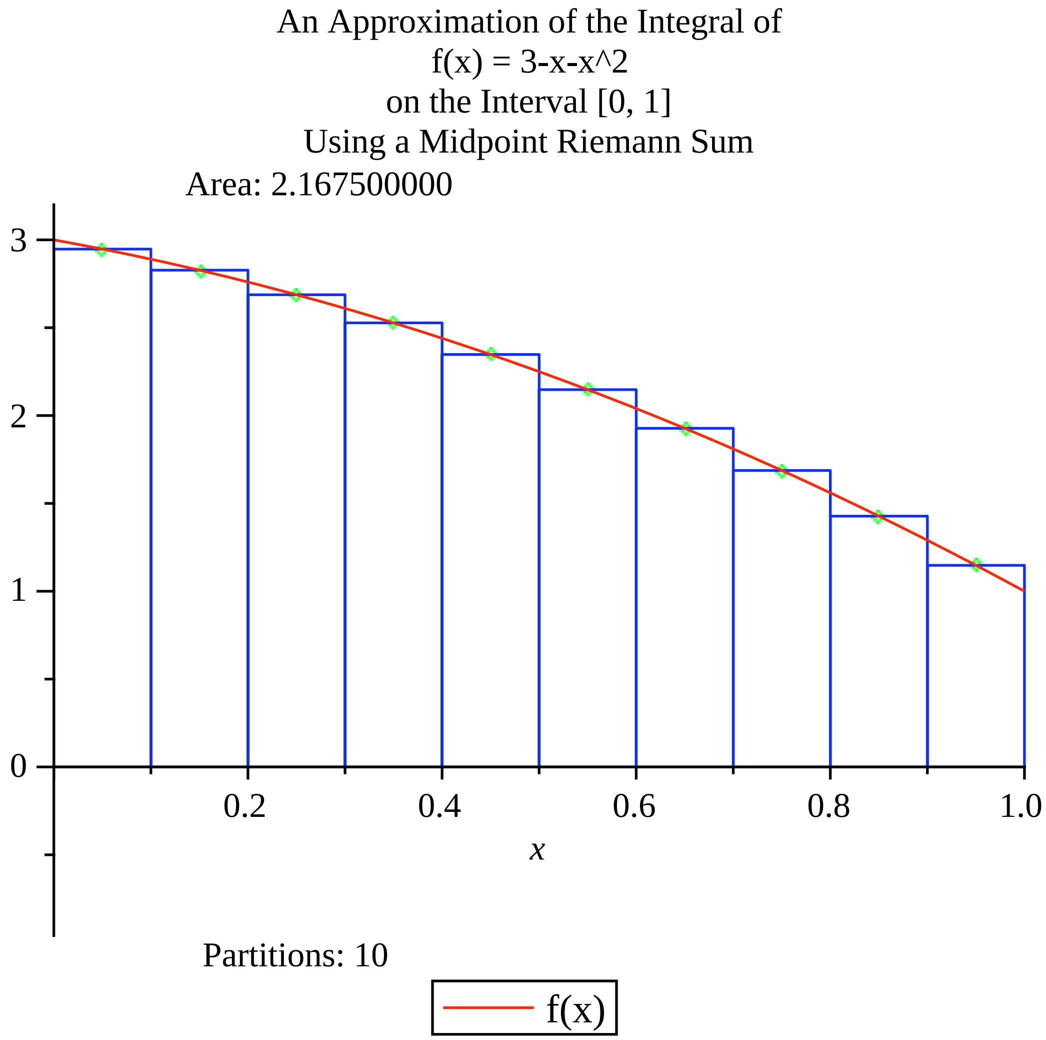
$$rightsum := \frac{413}{200}$$

$$rightsum := 2.065000000$$

$$\mathit{lowersum} := 2.065000000$$

At this point we know that $2.065 \leq \int_0^1 (3 - x - x^2)\,\mathrm{d}x$ and $\int_0^1 (3 - x - x^2)\,\mathrm{d}x \leq 2.265$. Increasing the number of rectangles or partition points along the x-axis will give an even closer approximation. In fact, we can choose any point from within each interval at which to compute the function, not just the left and right endpoints, and at the limit we will still get the integral. We now approximate the integral using the midpoint of each rectangle (**method=midpoint**).

```
> ApproximateInt(f, x=0..1, method=midpoint, partition=10, output=
  plot);
  midpointsum:=ApproximateInt(f, x=0..1, method=midpoint,
  partition=10, output=value);
  midpointsum:=evalf(midpointsum);
```



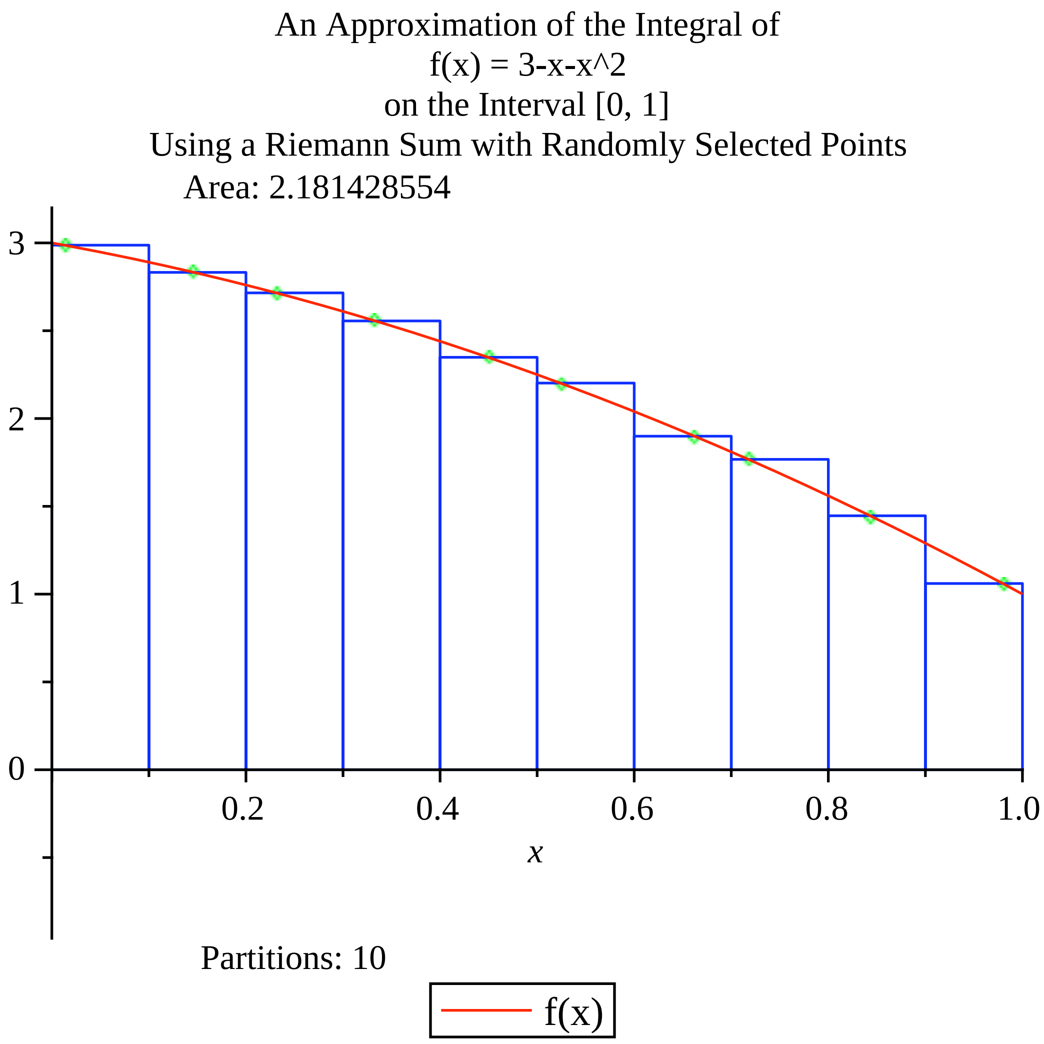
$$\mathit{midpointsum} := \frac{867}{400}$$

$$\mathit{midpointsum} := 2.167500000$$

Notice that this is not the average of the upper and lower approximations. We now approximate the integral using a random point of each rectangle (**method=random**).

```
> ApproximateInt(f, x=0..1, method=random, partition=10, output=
  plot);
```

```
randomsum:=ApproximateInt(f, x=0..1, method=random, partition=
10, output=value);
randomsum:=evalf(randomsum);
```



$$randomsum := \frac{868816203202461242111112971}{399999999990400000000005760}$$

$$randomsum := 2.172040508$$

Notice that we likely have two different answers here since two sets of random numbers were used. We now find the actual value of the integral.

```
> Int(3-x-x^2,x = 0 .. 1)=int(3-x-x^2,x = 0 .. 1);
```

$$\int_0^1 \left(3 - x - x^2\right) \mathrm{d}x = \frac{13}{6}$$

Using a decimal approximation, we get:

```
> Int(3-x-x^2,x = 0 .. 1)=evalf(int(3-x-x^2,x = 0 .. 1));
```

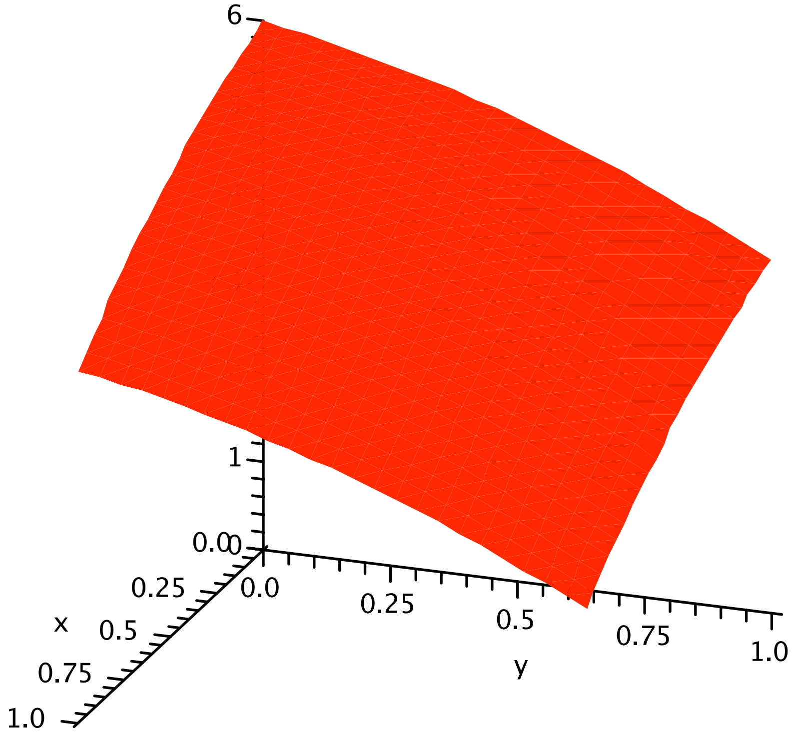
$$\int_0^1 \left(3 - x - x^2\right) \mathrm{d}x = 2.166666667$$

Integrals of Functions of Two Variables.

Next we consider the function $f(x, y) = 3 - x - x^2 - y - y^2$ of two variables and attempt to estimate its integral over the region R=[0, 1]x[0, 1]. This is the volume of the space below the surface and above the x-y plane. We first enter and graph the function.

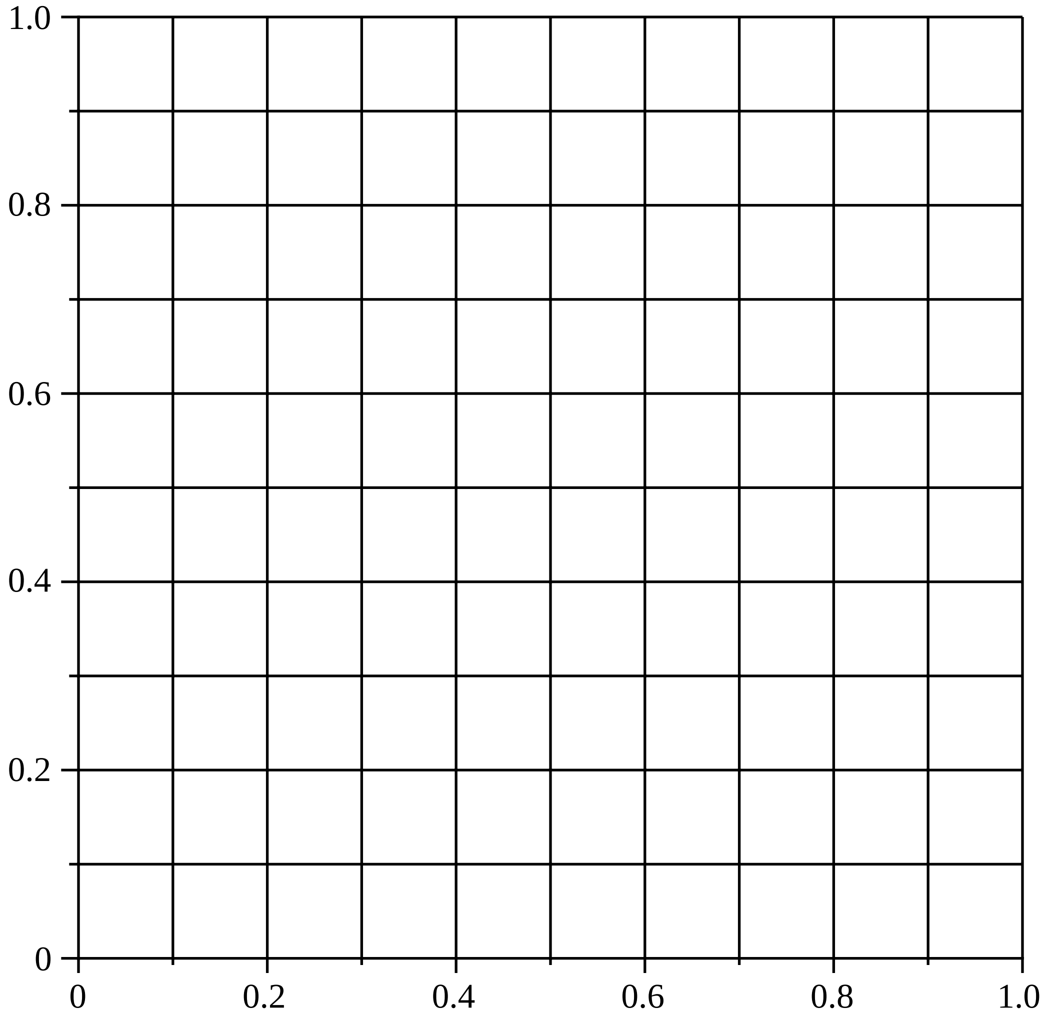
```
> z:=6-x-x^2-y-y^2;
> p0:=plot3d(z,x=0..1,y=0..1,view=0..6,style=patchnogrid,color=
  red):
  display(p0);
```

$$z := 6 - x - x^2 - y - y^2$$



Just like we partitioned the x-axis into n (10) equal intervals for the one-dimensional case, we partition the region R = [0, 1]x[0, 1] into n^2 ($10^2 = 100$ here) equal squares, as is shown below.

```
> for i from 1 to 10 do
  p[i]:=pointplot({[i*.1,0],[i*.1,1]},style=line)
  end do:
  for i from 1 to 10 do
  p[i+10]:=pointplot({[0,i*.1],[1,i*.1]},style=line)
  end do:
  display(seq(p[i],i=1..20),scaling=constrained);
```



To illustrate and compute approximations of the integral, we need to load the **MultivariateCalculus** subpackage of the **Student** package. which redefines some commands from the **Calculus1** package.

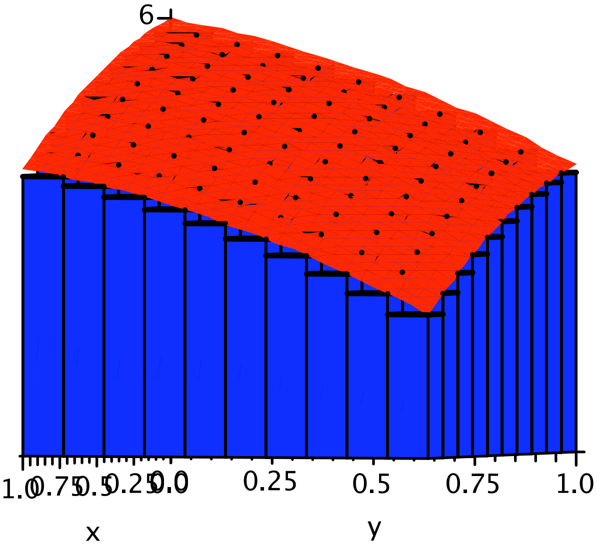
```
> with(MultivariateCalculus);
```

[ApproximateInt, ApproximateIntTutor, CenterOfMass, ChangeOfVariables, CrossSection, CrossSectionTutor, DirectionalDerivative, DirectionalDerivativeTutor, FunctionAverage, Gradient, GradientTutor, Jacobian, LagrangeMultipliers, MultiInt, Revert, SecondDerivativeTest, SurfaceArea, TaylorApproximation, TaylorApproximationTutor]

We approximate the integral over the region R by using ApproximateInt from the **MultivariateCalculus** package to compute a **lower Riemann sum**. Due to the nature of this function, the function is computed at the corner of each rectangle furthest from the origin.

```
> ApproximateInt(z, x=0..1, y=0..1, method=lower, partition=[10,
  10], output=plot);
```

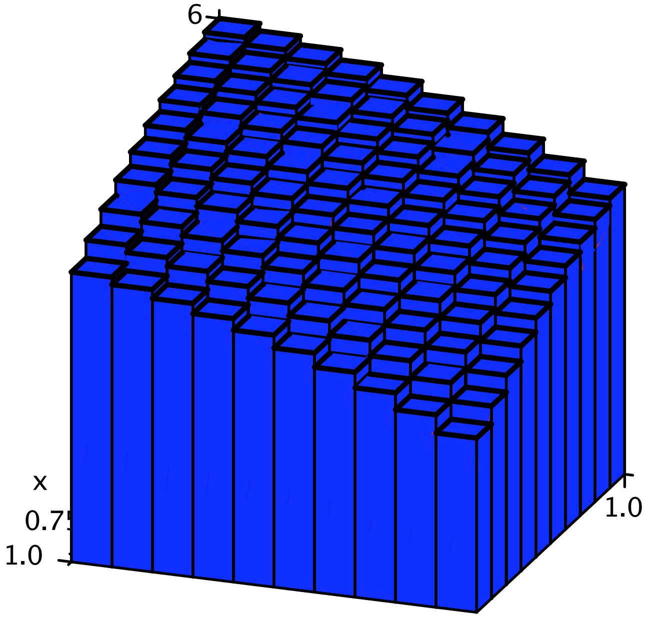
An approximation of the integral of
f(x, y) = 6−x−x^2−y−y^2
over the region [0 .. 1, 0 .. 1]
using a lower Riemann sum
Actual value: 4.3333
Approximate value: 4.1300



Next we approximate the integral with an **upper Riemann sum**. Now the function is computed at the corner of each rectangle closest from the origin.

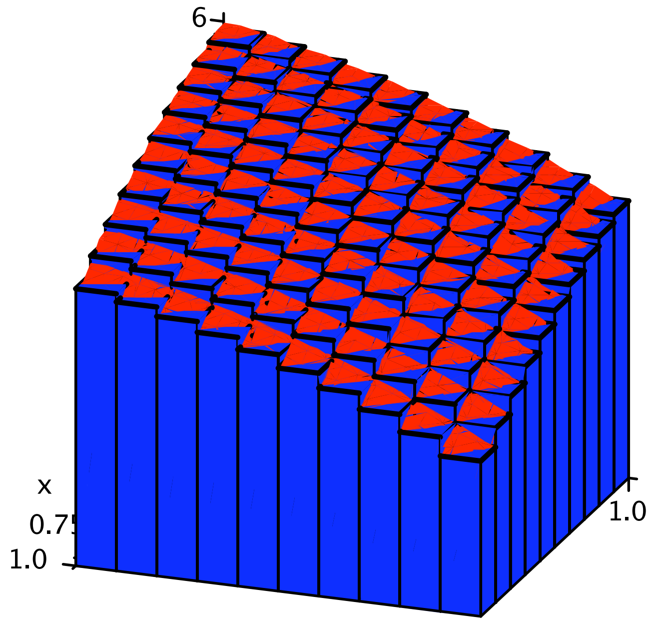
```
> ApproximateInt(z, x=0..1, y=0..1, method=upper, partition=[10,
  10], output=plot);
```

An approximation of the integral of
f(x, y) = 6−x−x^2−y−y^2
over the region [0 .. 1, 0 .. 1]
using a upper Riemann sum
Actual value: 4.3333
Approximate value: 4.5300



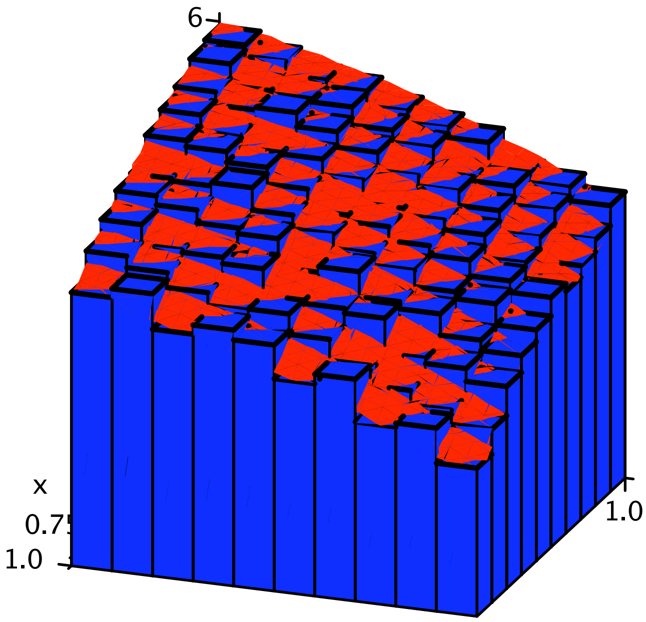
Now we approximate the integral with an **midpoint Riemann sum**. The function is computed at the midpoint of each rectangle.

```
> ApproximateInt(z, x=0..1, y=0..1, method=midpoint, partition=
  [10,10], output=plot);
```


An approximation of the integral of
f(x, y) = 6-x-x^2-y-y^2
over the region [0 .. 1, 0 .. 1]
using a midpoint Riemann sum
Actual value: 4.3333
Approximate value: 4.3350

Finally, we approximate the integral with a **random Riemann sum**. Now the function is computed at a random point in each rectangle.

```
> ApproximateInt(z, x=0..1, y=0..1, method=random, partition=[10,
  10], output=plot);
```



An approximation of the integral of
f(x, y) = 6−x−x^2−y−y^2
over the region [0 .. 1, 0 .. 1]
using a random Riemann sum
Actual value: 4.3333
Approximate value: 4.3314

We now know that $4.1300 \le \int_0^1\int_0^1 (6 - x - x^2 - y - y^2)\, \mathrm{d}x\, \mathrm{d}y$ and $\int_0^1\int_0^1 (6 - x - x^2 - y - y^2)\, \mathrm{d}x\, \mathrm{d}y \le 4.5300$. Increasing the number of bars or partition points along both axes will give an even closer approximation. In fact, we can choose any point from within each rectangle on the grid at which to compute the function, not just certain corner points, and at the limit we will still get the integral. We evaluate the integral

```
> Int(Int(6-x-x^2-y-y^2,x = 0 .. 1),y = 0 .. 1)=int(int(6-x-x^2-y-
  y^2,x = 0 .. 1),y = 0 .. 1);
>
```

$$\int_0^1\int_0^1 \left(6 - x - x^2 - y - y^2\right) \mathrm{d}x\, \mathrm{d}y = \frac{13}{3}$$

A decimal approximation is:

```
> Int(Int(6-x-x^2-y-y^2,x = 0 .. 1),y = 0 .. 1)=evalf(int(int(6-x-
  x^2-y-y^2,x = 0 .. 1),y = 0 .. 1));
```

$$\int_0^1\int_0^1 \left(6 - x - x^2 - y - y^2\right) \mathrm{d}x\, \mathrm{d}y = 4.333333333$$

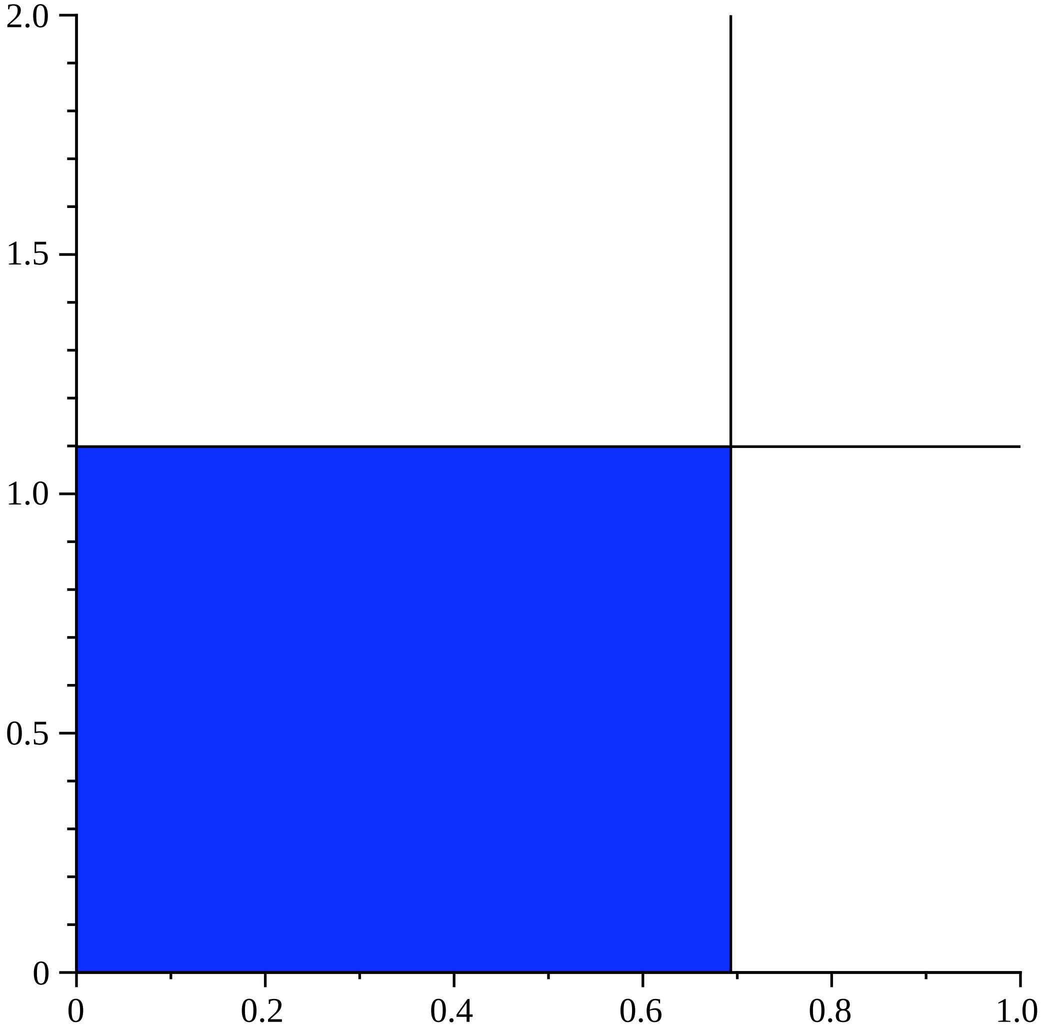
```
>
```

Evaluating an Iterated Integral over a Rectangle.

```
> restart:with(plots):
> setoptions3d(axes=NORMAL,labels=["x","y","z"],orientation=[20,
  70]);
```

We wish to integrate the function $f(x, y) = e^{x+y}$ over the rectangle $[0, \ln(2)]\text{x}[0, \ln(3)]$. We first view our rectangle.

```
> inequal({x>=0,x<=ln(2),y>=0,y<=ln(3)},x=0..1,y=0..2,
  optionsfeasible=(color=blue),optionsexcluded=(color=white));
```



Next we integrate in both orders.

```
> integral_1:=Int(Int(exp(x+y),x=0..ln(2)),y=0..ln(3))=int(int(exp
  (x+y),x=0..ln(2)),y=0..ln(3));
```

$$integral\_1 := \int_0^{\ln(3)} \int_0^{\ln(2)} e^{x+y}\, dx\, dy = 2$$

```
> integral_2:=Int(Int(exp(x+y),y=0..ln(3)),x=0..ln(2))=int(int(exp
  (x+y),y=0..ln(3)),x=0..ln(2));
```

$$integral\_2 := \int_0^{\ln(2)} \int_0^{\ln(3)} e^{x+y}\, dy\, dx = 2$$

We see that the two iterated integrals are equal, as is expected.

```
>
```

Evaluating an Iterated Integral over a Nonrectangular region.

We wish to integrate the function $f(x, y) = x\,y$ over the region R bounded by the parabola $x = y^2$ and the line $3\,x + 2\,y = 8$. We enter the equations that define our region.

```
> line:=3*x+2*y=8;parabola:=x=y^2;
```

$$line := 3\,x + 2\,y = 8$$

$$parabola := x = y^2$$

For graphing, we solve each of the equations for y.

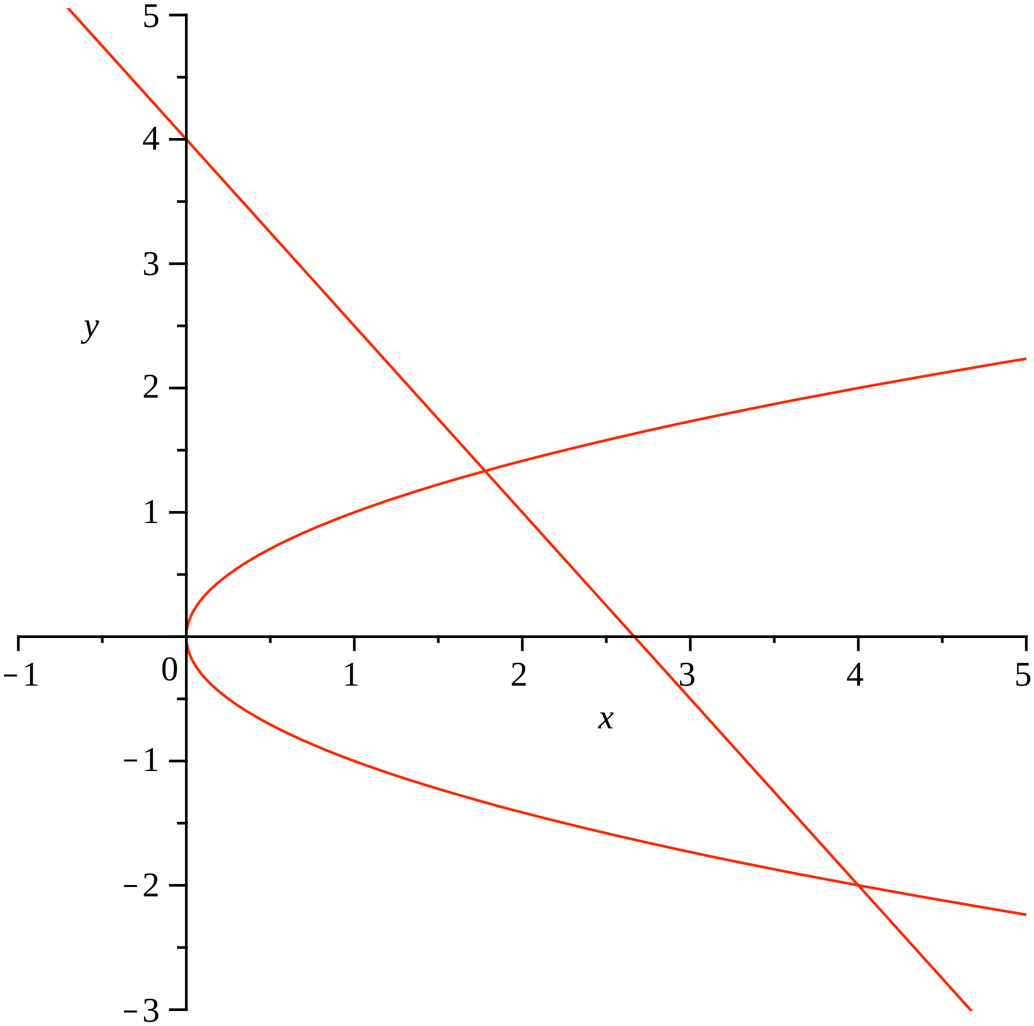
```
> y_line:=solve(line,y);
  y_parabola:=solve(parabola,y);
```

$$y\_line := -\frac{3}{2}\,x + 4$$

$$y\_parabola := \sqrt{x},\ -\sqrt{x}$$

We notice that $y\_parabola$ has two solutions. We plot our graphs.

```
> plot({y_line,y_parabola},x=-1..5,y=-3..5,color=red);
```



We solve our defining equations to find the points where the line intersects the parabola.

```
> solve({line,parabola},{x,y});
```

$$\left\{x = \frac{16}{9},\ y = \frac{4}{3}\right\},\ \{x = 4,\ y = -2\}$$

Since our region R is both horizontally and vertically simple, we compute the iterated integrals in both orders.

```
> integral_1:=Int(Int(x*y,x=y^2..-(2/3)*y+(8/3)),y=-2..(4/3))=int
  (int(x*y,x=y^2..-(2/3)*y+(8/3)),y=-2..(4/3));
```

$$integral\_1 := \int_{-2}^{\frac{4}{3}} \int_{y^2}^{-\frac{2}{3}y+\frac{8}{3}} x\,y\,\mathrm{d}x\,\mathrm{d}y = -\frac{13000}{2187}$$

```
> integral_2:=Int(Int(x*y,y=-sqrt(x)..sqrt(x)),x=0..16/9)+Int(Int
  (x*y,y=-sqrt(x)..-(3/2)*x+4),x=16/9..4)=int(int(x*y,y=-sqrt(x)..
  sqrt(x)),x=0..16/9)+int(int(x*y,y=-sqrt(x)..-(3/2)*x+4),x=16/9.
  .4);
```

$$integral\_2 := \int_{0}^{\frac{16}{9}} \int_{-\sqrt{x}}^{\sqrt{x}} x\,y\,\mathrm{d}y\,\mathrm{d}x + \int_{\frac{16}{9}}^{4} \int_{-\sqrt{x}}^{-\frac{3}{2}x+4} x\,y\,\mathrm{d}y\,\mathrm{d}x = -\frac{13000}{2187}$$

We see they are again equal.

```
>
```

A Necessary Changing of Order of Integration.

We now wish to evaluate the integral $\int_{0}^{3}\int_{y^2}^{9} y \sin(x^2)\,\mathrm{d}x\,\mathrm{d}y$.

```
> integral_1:=Int(Int(y*sin(x^2),x=y^2..9),y=0..3)=int(int(y*sin
  (x^2),x=y^2..9),y=0..3);
```

That is some solution. The problem is that $y \sin(x^2)$ does not have any elementary antiderivative with respect to x. So we need to try reversing the order of integration. We need to plot the functions $x = 9$ and $x = y^2$ between $y = 0$ and $y = 3$.

```
> p1:=pointplot({[9,0],[9,3]},style=LINE,color=red):
  p2:=plot(sqrt(x),x=0..10):
  display(p1,p2);
```

Since our region R is both horizontally and vertically simple, we can change the order of integration.

```
> integral_1:=Int(Int(y*sin(x^2),y=0..sqrt(x)),x=0..9)=int(int(y*
  sin(x^2),y=0..sqrt(x)),x=0..9);
```

Ah, we now get something we can understand.

```
>
>
>
```