

x	y
0.0	2.000
0.1	1.901
0.2	1.808
0.3	1.727
0.4	1.664
0.5	1.625

x	y
0.6	1.616
0.7	1.643
0.8	1.712
0.9	1.829
1.0	2.000

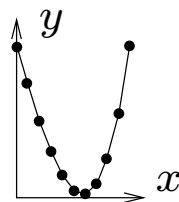


Figure 1. A table of xy -values for $y = x^3 - x + 2$. The graphic represents the table's rows as dots, which are joined to make the connect-the-dots graphic.

Table 1. Three numerical integration methods.

Rect	$Y = y_0 + hF(x_0)$
Trap	$Y = y_0 + \frac{h}{2}(F(x_0) + F(x_0 + h))$
Simp	$Y = y_0 + \frac{h}{6}(F(x_0) + 4F(x_0 + h/2) + F(x_0 + h))$

```

# Rectangular algorithm
# Group 1, initialize.
F:=x->evalf(cos(x) + 2*x):
x0:=0:y0:=0:h:=0.1*Pi:
Dots1:=[x0,y0]:

# Group 2, repeat 10 times
Y:=y0+h*F(x0):
x0:=x0+h:y0:=evalf(Y):
Dots1:=Dots1,[x0,y0];

# Group 3, plot.
plot([Dots1]);

```

```

# Trapezoidal algorithm
# Group 1, initialize.
F:=x->evalf(cos(x) + 2*x):
x0:=0:y0:=0:h:=0.1*Pi:
Dots2:=[x0,y0]:

# Group 2, repeat 10 times
Y:=y0+h*(F(x0)+F(x0+h))/2:
x0:=x0+h:y0:=evalf(Y):
Dots2:=Dots2,[x0,y0];

# Group 3, plot.
plot([Dots2]);

```

```

# Rectangular algorithm
# Group 1, initialize.
F:=x->evalf(exp(-x*x)):
x0:=0:y0:=0:h:=0.1:
Dots1:=[x0,y0]:

# Group 2, repeat 10 times
Y:=evalf(y0+h*F(x0)):
x0:=x0+h:y0:=Y:
Dots1:=Dots1,[x0,y0];

# Group 3, plot.
plot([Dots1]);

```

```

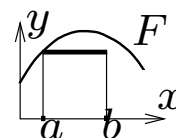
# Simpson algorithm
# Group 1, initialize.
F:=x->evalf(exp(-x*x)):
x0:=0:y0:=0:h:=0.1:
Dots3:=[x0,y0]:

# Group 2, repeat 10 times
Y:=evalf(y0+h*(F(x0)+
      4*F(x0+h/2)+F(x0+h))/6):
x0:=x0+h:y0:=Y:
Dots3:=Dots3,[x0,y0];

# Group 3, plot.
plot([Dots3]);

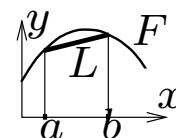
```

Rectangular Rule. The approximation uses Euler's idea of replacing the integrand by a constant. The value of the integral is approximately the area of a rectangle of width $b - a$ and height $F(a)$.



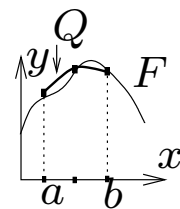
$$\int_a^b F(x)dx \approx (b - a)F(a)$$

Trapezoidal Rule. The rule replaces the integrand $F(x)$ by a linear function $L(x)$ which connects the planar points $(a, F(a))$, $(b, F(b))$. The value of the integral is approximately the area under the curve L , which is the area of a trapezoid.



$$\int_a^b F(x)dx \approx \frac{b - a}{2} (F(a) + F(b))$$

Simpson's Rule. The rule replaces the integrand $F(x)$ by a quadratic polynomial $Q(x)$ which connects the planar points $(a, F(a))$, $((a + b)/2, F((a + b)/2))$, $(b, F(b))$. Then the integral of F is approximately the area under the quadratic curve Q .



$$\int_a^b F(x)dx \approx (b - a) \left(\frac{F(a) + 4F\left(\frac{a+b}{2}\right) + F(b)}{6} \right)$$

```

# Euler algorithm
# Group 1, initialize.
f:=(x,y)->-y+1-x:
x0:=0:y0:=3:h:=0.1:L:=[x0,y0]:

# Group 2, repeat 10 times
Y:=y0+h*f(x0,y0):
x0:=x0+h:y0:=Y:L:=L,[x0,y0];

# Group 3, plot.
plot([L]);

```

```

# Heun algorithm
# Group 1, initialize.
f:=(x,y)->-y+1-x:
x0:=0:y0:=3:h:=0.1:L:=[x0,y0]:

# Group 2, repeat 10 times
Y:=y0+h*f(x0,y0):
Y:=y0+h*(f(x0,y0)+f(x0+h,Y))/2:
x0:=x0+h:y0:=Y:L:=L,[x0,y0];

# Group 3, plot.
plot([L]);

```

```

# Heun algorithm
# Group 1, initialize.
f:=(x,y)->-y+1-x:
x0:=0:y0:=3:h:=0.1:L:=[x0,y0]:

# Group 2, repeat 10 times
Y:=y0+h*f(x0,y0):
Y:=y0+h*(f(x0,y0)+f(x0+h,Y))/2:
x0:=x0+h:y0:=Y:L:=L,[x0,y0];

# Group 3, plot.
plot([L]);

```

```

# RK4 algorithm
# Group 1, initialize.
f:=(x,y)->-y+1-x:
x0:=0:y0:=3:h:=0.1:L:=[x0,y0]:

# Group 2, repeat 10
times.
k1:=h*f(x0,y0):
k2:=h*f(x0+h/2,y0+k1/2):
k3:=h*f(x0+h/2,y0+k2/2):
k4:=h*f(x0+h,y0+k3):
Y:=y0+(k1+2*k2+2*k3+k4)/6:
x0:=x0+h:y0:=Y:L:=L,[x0,y0];

# Group 3, plot.
plot([L]);

```