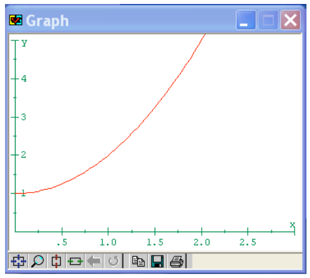
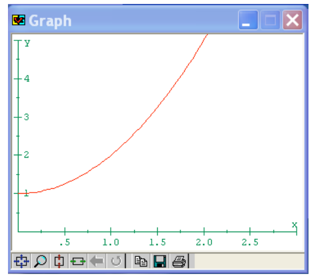
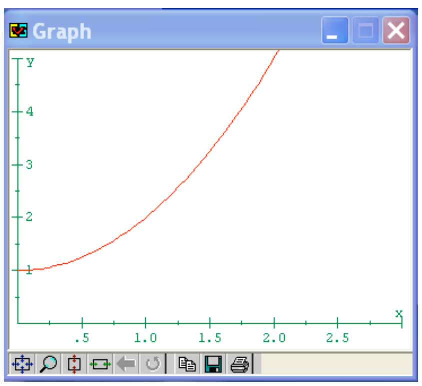
Example Area Problem – Introduction to Integration.

Find the area under the curve $f(x) = x^2+1$ over the interval [0, 2].

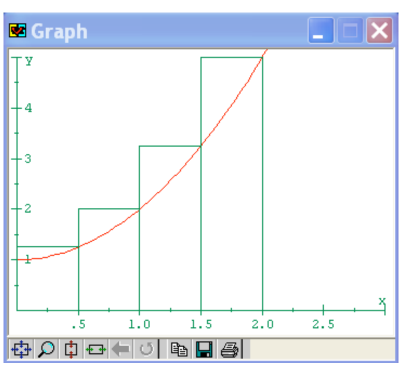




Suppose we cut the region into 4 vertical strips of equal width and create rectangles, and let us choose to draw the top of the rectangle in accordance with the functional value at the left endpoint of each subinterval. The area enclosed in the four rectangles L_4 is an approximation (an under estimate in this case) to the desired area under the curve.

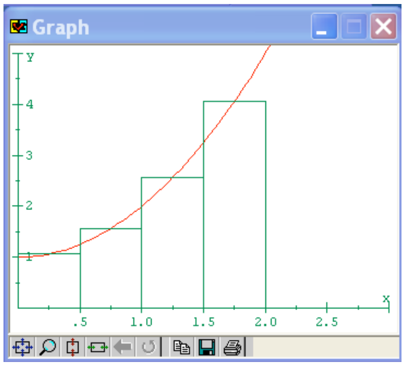


Suppose we choose to draw the top of the rectangle in accordance with the functional value at the right endpoint of each subinterval. The area enclosed in the four rectangles, R_4 is an approximation (an over estimate in this case) to the desired area under the curve.

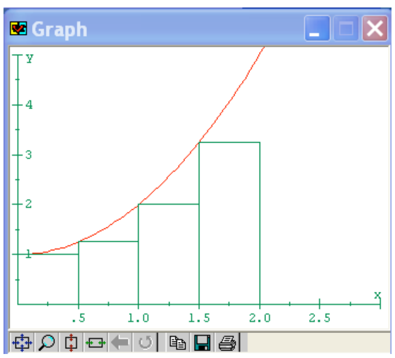


Using these two estimates, what can we say about the actual area? ______________________________

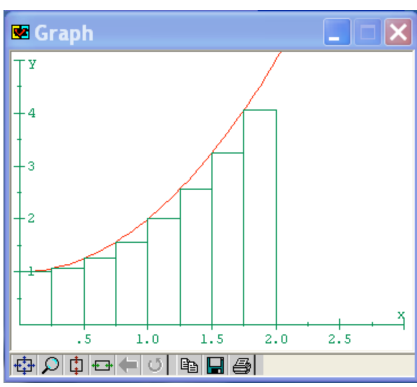
Suppose this time we choose to draw the top of the rectangle in accordance with the functional value at the midpoint of each subinterval. From the graph it appears that the area enclosed in the four rectangles, M_4 is a "good" approximation in this case.



Returning to the situation where we computed L_4, how might we improve this estimate?



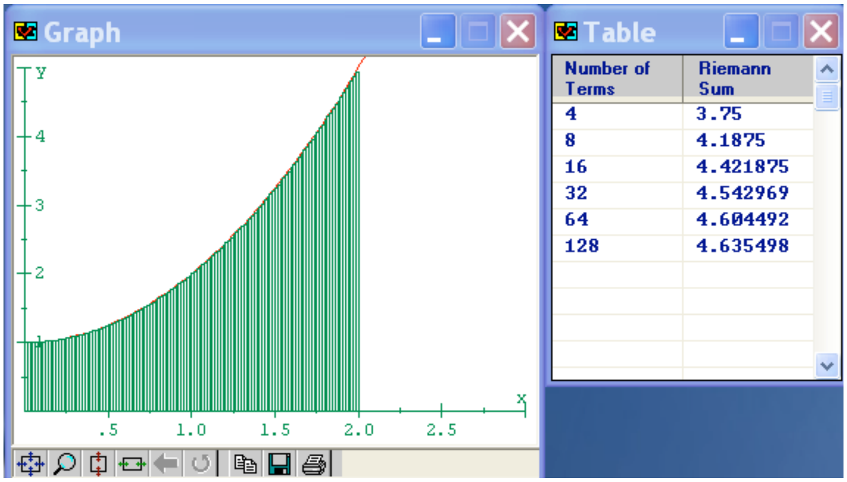
Suppose we use 8 rectangles with height determined by the left endpoint on each subinterval.



Table

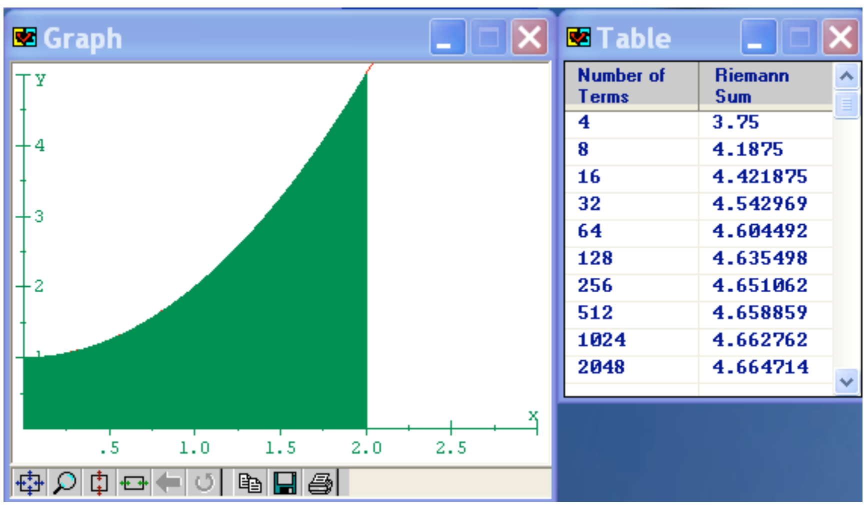
Number of Terms	Riemann Sum
4	3.75
8	4.1875

More.. 128 rectangles L_{128}



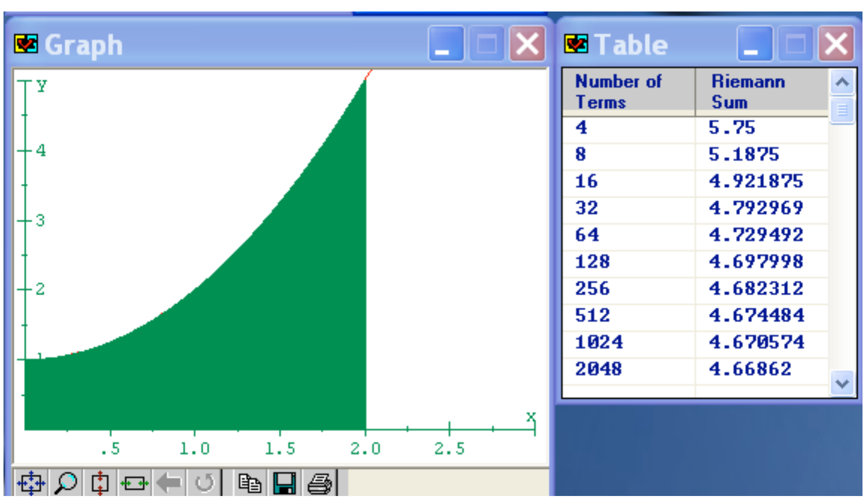
Number of Terms	Riemann Sum
4	3.75
8	4.1875
16	4.421875
32	4.542969
64	4.604492
128	4.635498

Still more.. 2048 rectangles !! L_{2048}



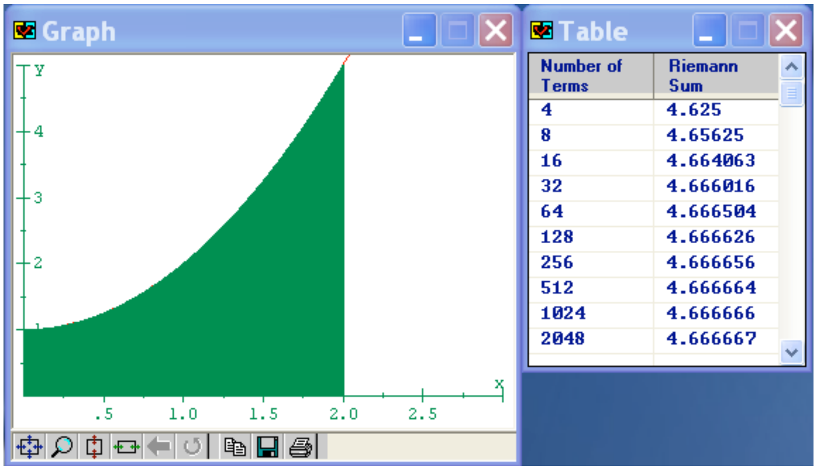
Number of Terms	Riemann Sum
4	3.75
8	4.1875
16	4.421875
32	4.542969
64	4.604492
128	4.635498
256	4.651062
512	4.658859
1024	4.662762
2048	4.664714

Let's do the same thing for the right endpoint, jumping right to 2048 rectangle. R_{2048}



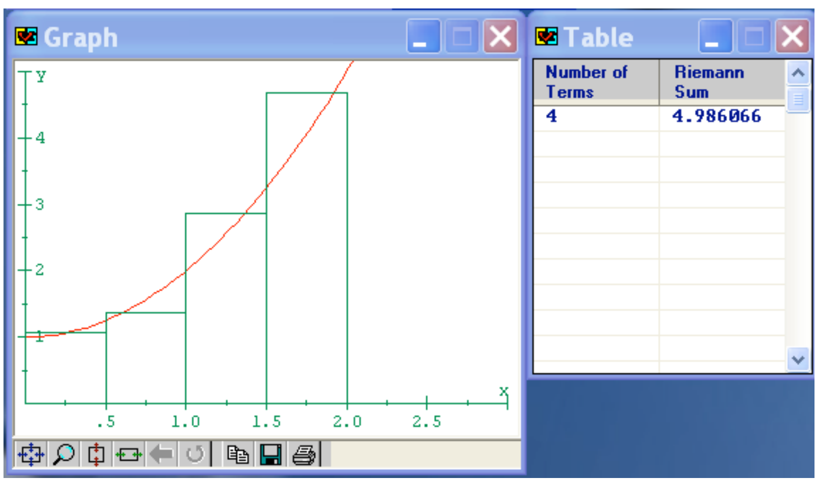
Number of Terms	Riemann Sum
4	5.75
8	5.1875
16	4.921875
32	4.792969
64	4.729492
128	4.697998
256	4.682312
512	4.674484
1024	4.670574
2048	4.66862

And using the midpoint, jumping right to 2048 rectangle. M_{2048}



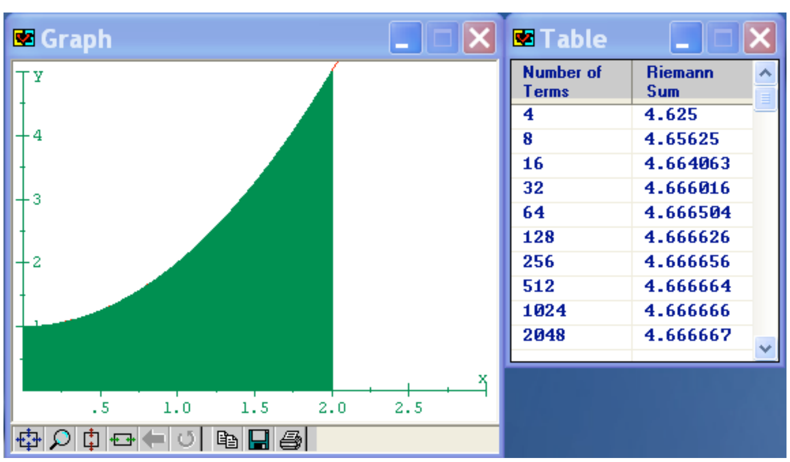
Number of Terms	Riemann Sum
4	4.625
8	4.65625
16	4.664063
32	4.666016
64	4.666504
128	4.666626
256	4.666656
512	4.666664
1024	4.666666
2048	4.666667

It appears that if we take enough rectangles, it doesn't matter which sample point we use to determine the height of the rectangle. Suppose we compute an estimate for 4 rectangles where the height of the rectangle is determined by a randomly chosen point in each subinterval.



Number of Terms	Riemann Sum
4	4.986066

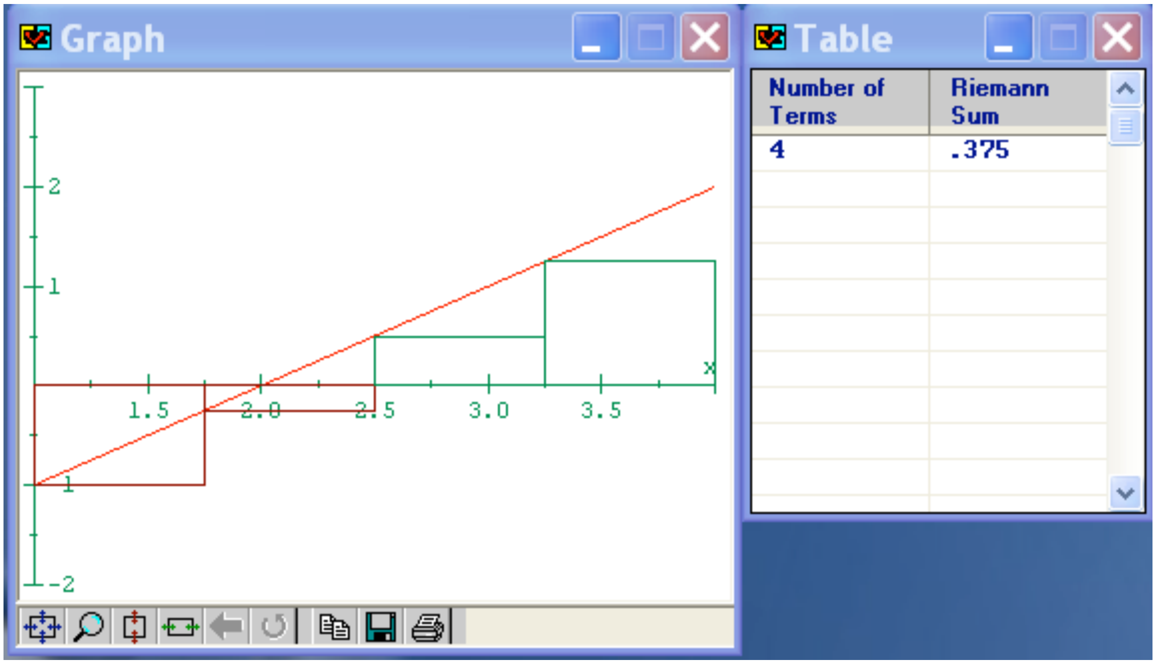
Now 2048 randomly chosen.

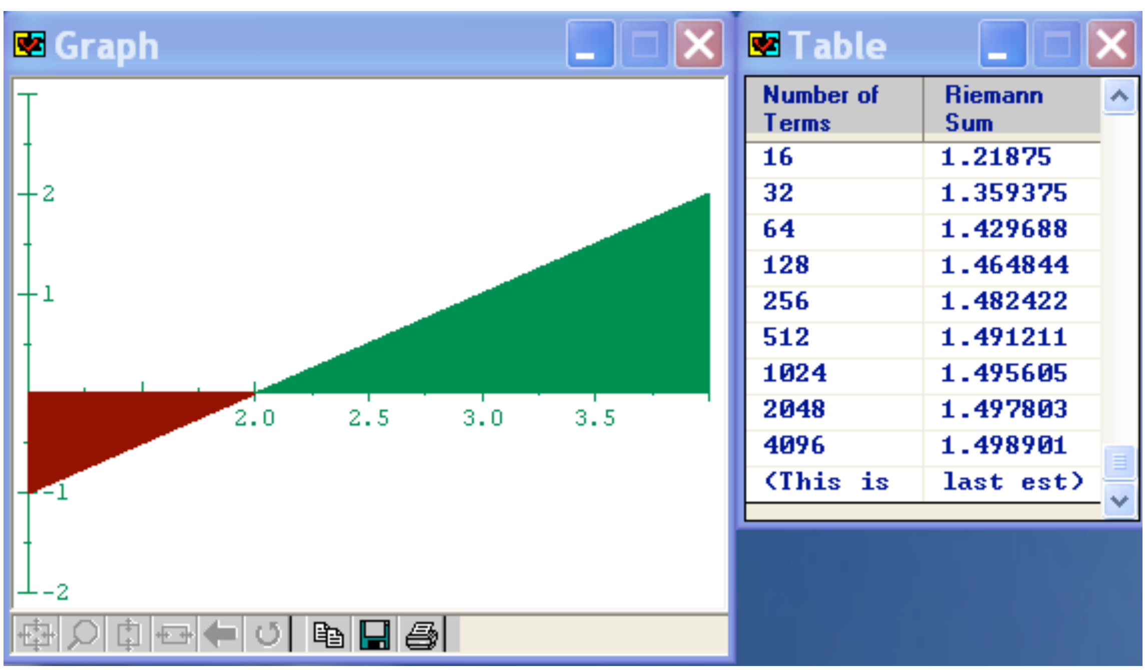


Number of Terms	Riemann Sum
4	4.625
8	4.65625
16	4.664063
32	4.666016
64	4.666504
128	4.666626
256	4.666656
512	4.666664
1024	4.666666
2048	4.666667

What if we apply this process to a function that above the x axis, that is f(x)<0?

Example: f(x)=x-2 on [1,4].





So we can see that this process does not yeild area in the case f(x) is not greater than or equal to zero. In this case it is the "net-signed-area", that it, area above – area below.