

Approximating Areas on the TI-83

To do these approximations efficiently, you need to know about creating and operating with lists, and how to use the "sum" command.

Creating, Storing, and Manipulating Lists

A list of numbers can be created and stored in one of the "list variables" L1, L2, . . . , L6 (these are found at "2nd L1", . . . , "2nd L6" on the TI-83). One way to do this is to enter the list in braces, separated by commas, and store it in one of the list variables. For example,

$\{1,2,3,4,5\}$ STO L1 (*stores the list {1 2 3 4 5} in the variable L1*)

Lists are operated on one term at a time. For example, $2*L1$ produces the new list $\{2\ 4\ 6\ 8\ 10\}$, and $(L1)/2$ produces the list $\{.5\ 1\ 1.5\ 2\ 2.5\}$. If you have defined a function in the "Y=" window, then you can also apply it to a list. For example, if $Y1=X^2$, then $Y1(L1)$ gives the new list $\{1\ 4\ 9\ 16\ 25\}$, obtained by squaring each term in the old list.

(Note: separation by commas is necessary when entering the list but when the TI-83 displays the list, the commas don't show up on the screen.)

Creating a long list of numbers by entering them one at a time can be tedious. If there's a pattern to the numbers, you may be able to use the "seq" command (found at 2nd LIST OPS on the TI-83) to create the list of numbers. For example, the command

$\text{seq}(2I, I, 1, 100)$

creates the list $\{2, 4, 6, \dots, 198, 200\}$. (*The use of the letter "I" is not important; any other letter could be used instead. Think of the command as reading "create a sequence of numbers using the formula 2I, as I takes on the values 1, 2, 3, ... up to 100"*). If you want to save this list, say in L1, then after executing the "seq" command, hit STO L1.

Finding the Sum of a List

The "sum" function adds the numbers in a list. (It is found at 2nd LIST MATH on the TI-83.) For example, if the list $\{1, 2, 3, 4, 5\}$ has been stored in L1, then the command

$\text{sum}(L1)$

gives the answer 15 ($= 1+2+3+4+5$)

The "sum" command can also be combined with the "seq" command to create the list and sum it at the same time. For example, to find the sum $2+4+6+\dots+198+200$, we could use the command

$\text{sum}(\text{seq}(2I, I, 1, 100))$ (which gives the answer 10100).

In "sigma notation", this sum could be written as $\sum_{i=1}^{100} 2i = 10100$.

Examples

1) Find the value of $(\sin .1)(.01) + (\sin .3)(.01) + (\sin .5)(.01) + \dots + (\sin 6.3)(.01)$

To find this sum, we will need to compute the sin of each number in the list $\{.1, .3, .5, \dots, 6.3\}$. So we start by inventing a formula which creates this list. For example,

`seq((2i+1)/10, I, 0, 31)` (*Test it! Could you find a different formula ?*)

Then we want to take the sine of each number in the list, multiply each result by .01, and finally add them all. We can do all this with the command

`sum(.01*sin(seq((2i+1)/10, I, 0, 31)))`

The TI-83 gives the answer $3.413226488E - 4$ or 0.0003413226488 .

Another way to get the answer would be with the command

`.01*(sum(sin(seq((2i+1)/10, I, 0, 31))))` (*why?*)

An even neater way would be with the following sequence of commands:

`seq((2i+1)/10, I, 0, 31)`
`STO L1`
`.01*sum(sin(L1))` (*explain*)

The sum we just found, written in sigma notation, is

$$\sum_{i=0}^{31} .01 \sin\left(\frac{2i+1}{10}\right) \simeq 0.0003413226488$$

2) (*Read this example along with the discussion and pictures in Example 1-2, p 346-9 of the text.*) We want to estimate the area under the parabola $y = f(x) = x^2$ and above the interval $[0, 1]$.

Suppose we divide the interval $[0, 1]$ into 100 equal subintervals, each of length $\frac{1-0}{100} = \frac{1}{100}$.

On each subinterval, we construct a rectangle whose height is the distance up to the graph at the right endpoint of the subinterval. The sum of the areas of all these rectangles approximates (in this example, overestimates) the exact area under the graph and above $[0, 1]$. (*See Figure 8, p. 349, which illustrates for 10, 30, and 50 rectangles.*)

To get the sum of these areas on the TI-83 :

`Y1=X^2` (*Enter in the "Y=" window; return to the main screen*)
`seq(I/100, I, 1, 100)` (*this makes a list of the right endpoints of the subintervals*)
`STO L1` (*store the list of right endpoints in L1*)
`sum(Y1(L1)*(1/100))` (*Y1(L1) is a list of the values of the function x^2 at the right endpoints, so $Y1(L1)*(1/100)$ is a list of the areas of the 100 rectangles, and the sum function adds them together.*)

The TI-83 gives the answer 0.33835. Notice that this is the value for R_{100} in the table on p. 347.

If, instead, we want to use the distance up to the graph at the left endpoints to construct the rectangles, we do the "same" thing, except that we start with a list of the left endpoints of the subintervals:

```
seq( (I - 1)/100, I, 1, 100 )  ( or you could use seq(I/100, I, 0, 99) to get the same list )
STO L2                          ( store the list in L2 )
sum( Y1(L2)*(1/100) )
```

(Figure 9, p. 349 illustrates for 10, 30, and 50 rectangles).

The TI-83 gives the answer 0.32835. Notice that this is the value for L_{100} in the table on p. 347.

3) Estimate the area under the graph of $y = f(x) = \frac{3}{1+x^2}$ and over the interval $[1, 3]$ using 20 "right endpoint" rectangles. (Make a sketch to be sure you know what the situation looks like!)

We can do this in the same way on the TI-83 once we get a list of the right endpoints. If we divide $[1, 3]$ into 20 equal subintervals, each one will be $\frac{3-1}{20} = \frac{1}{10} = 0.1$ long. Therefore the right endpoints of the subintervals will be 1.1, 1.2, 1.3, ..., 2.8, 2.9, 3. So we can generate a list of the right endpoints using the "seq" command and store it in L1 as follows:

```
seq(1 + .1*I, I, 1, 20)
STO L1
```

If we enter $Y1=3/(1+x^2)$ (in the "Y=" window),

then the sum of the areas of the 20 rectangles is

```
sum(.1*Y1(L1))  ( since .1 is the width of each rectangle and Y1(L1) is
                  the list of rectangle heights )
```

The TI-83 gives the answer 1.33204285 for the sum of the rectangular areas.