1 The Midpoint Approximation

We first partition the interval $[a, b]$ so that $a = x_0 < x_1 < x_2 < \cdots < x_{n-1} < x_n = b$, then use the midpoint of each resulting subinterval to calculate the height of a rectangle, as shown in **Figure 1**.

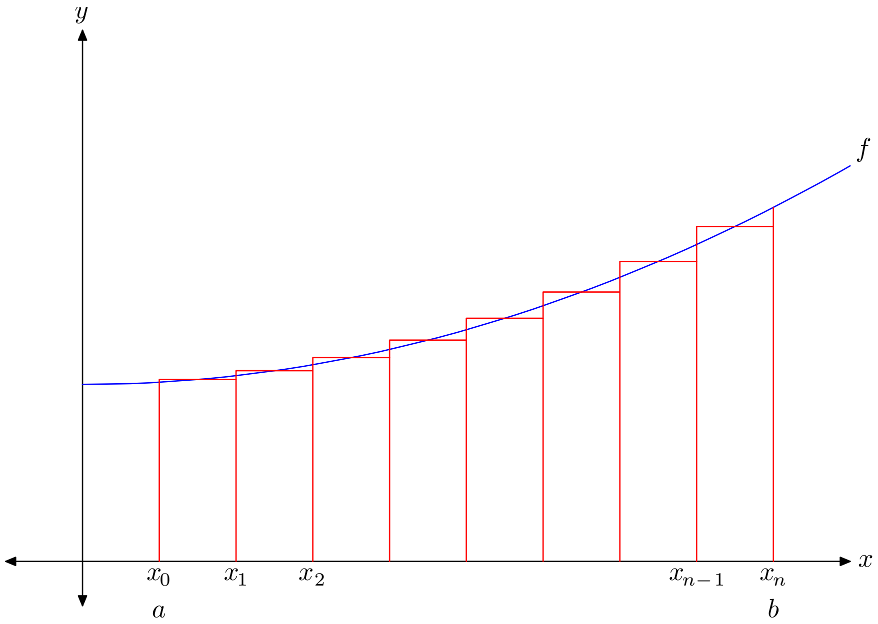

Figure 1 Using the midpoint to calculate the height of each rectangle.

We sum the areas of the rectangles as follows,

$$S = f\left(\frac{x_0 + x_1}{2}\right)\Delta x + f\left(\frac{x_1 + x_2}{2}\right)\Delta x + \cdots + f\left(\frac{x_{n-1} + (x_n}{2}\right)\Delta x, \tag{1}$$

where Δx is the width of each rectangle and is calculated by dividing the length of the interval $[a, b]$ by n.

$$\Delta x = \frac{b - a}{n} \tag{2}$$

The width of each rectangle, Δx, is a common factor, so we can write

$$S = \left[f\left(\frac{x_0 + x_1}{2}\right) + f\left(\frac{x_1 + x_2}{2}\right) + \cdots + f\left(\frac{x_{n-1} + x_n}{2}\right)\right]\Delta x \tag{3}$$

This will simplify the computation somewhat and ease the construction of the programming task.

We begin by providing the calculator with some initial data. We assume that the integrand of

$$\int_a^b f(x)dx \tag{4}$$

is currently loaded in the `Y=` Menu. For example if we wish to integrate

$$\int_1^2 \frac{1}{x}dx, \tag{5}$$

then we load $y = 1/x$ in `Y1`.

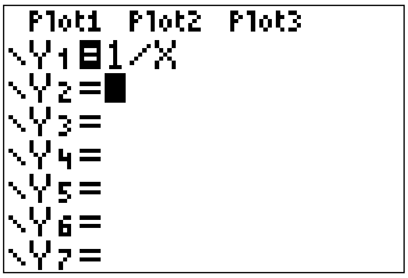

Figure 2 Entering the integrand in the `Y=` Menu.

The calculator will need to know the limits of integration and the number of subintervals in the partition. We also clear the home screen before prompting the user for this data.

```
Clr Home
Prompt A
Prompt B
Prompt N
```

We immediately calculate the width Δx of each subinterval and store it in the variable `H`.

```
(B-A)/N->H
```

Because we use the midpoint to calculate the height of each rectangle, the first function evaluation occurs at $x = a + h/2$. Therefore, we store the contents of the variable `A+H/2` in `X`.

```
A+H/2->X
```

We will store the sum in the variable `S`. It is important that we initialize this variable to zero before we begin.

```
0->S
```

We now add the areas of the `N` rectangles to the sum `S`. This is accomplished by proceeding one step at a time, moving from left to right. This calls for the use of a `For` loop.

```
For(K,1,N,1)
  S+Y1(X)->S
  X+H->X
End
```

This loop construct warrants some further explanation. The command `For(K,1,N,1)` initializes `K` to 1, then executes the loop. It increments the value of `K` by 1, then checks to see if `K` is less than or equal to `N`. If `K` is less than or equal to `N`, the loop is executed again. If `K` is greater than `N`, execution of the loop is halted and the program continues by executing the line following the `End` command, which terminates the loop. In the body of the loop, two things happen. First, the integrand, which was stored in `Y1`, is evaluated at `X`, added to the current value of `S`, then `S` is updated to contain this current sum. This is accomplished with the command `S+Y1(X)->S`. Next, `X` is incremented by `H` by executing the command `X+H->X`. This ensures that the next time through the loop, evaluation of the integrand takes place at the midpoint of the *next* subinterval.

Once the loop terminates, `S` contains the sum $f((x_0+x_1)/2)+f((x_1+x_2)/2)+\cdots+f((x_{n-1}+x_n)/2)$. By **Equation 3**, we must still multiply this sum byΔx. The command

```
S*H->S
```

multiplies the contents of `S` by `H`, then updates `S` to contain this product. The variable `S` now contains the sum of all `N` rectangles pictured in **Figure 1**. All that's left to do is display the answer for the user of the program. We indicate the integration method being used.

```
Disp "METHOD: MIDPOINT"
```

Then we display the sum.

```
Disp "SUM:"
Disp S
```

A complete listing of the program follows in **Appendix 2**.

We need to test our program. Here is some data you can use to determine whether or not your program is running properly.

No. of Rectangles	Sum
5	0.691908
10	0.692835
20	0.693069

Table 1 Integrating $1/x$ on the interval $[1, 2]$.

Enter the integrand of

$$\int_1^2 \frac{1}{x} dx$$

in the `Y=` Menu, as shown in **Figure 2**. Execute the program. Enter the upper and lower bounds of the integral when prompted. Enter $n = 5$ for the number of rectangles or subintervals.

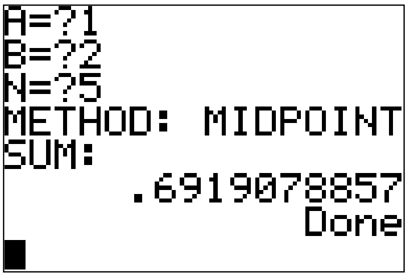

Figure 3 Entering the interval $[1, 2]$ and the number of subintervals ($n = 5$).

Note that this result agrees with the first row of **Table 1**. Use the program to verify the results in the next two rows of **Table 1**.

2 Appendix

Here is a complete program listing.

```
Clr Home
Prompt A
Prompt B
Prompt N
(B-A)/N->H
A+H/2->X
0->S
For(K,1,N,1)
  S+Y1(X)->S
```

```
  X+H->X
End
S*H->S
Disp "METHOD: MIDPOINT"
Disp "SUM:"
Disp S
```