In this handout we discuss how to compute left- and right- Riemann sums using *Mathematica*. Ultimately, to do a Riemann sum you need to execute three commands found on page 3; the first two pages are devoted to explaining where these commands come from.

1. Defining the function f

First, recall that to define a function you use an underscore. For example, the following command defines f to be the function $f(x) = x^2$:

```
f[x_] = x^2
```

2. Defining the partition $\mathcal{P}$

Defining a partition in *Mathematica* is easy. Just use braces, and list the numbers from smallest to largest. For example, to define the partition $\mathcal{P} = \{0, 1, \frac{5}{2}, 4, 7\}$, just execute

```
P = {0, 1, 5/2, 4, 7}
```

We often use partitions which divide $[a, b]$ into n equal-length subintervals. To create such a partition in *Mathematica*, use the `Range` command. For example, to define a partition of $[0, 2]$ into 10 equal-length subintervals, execute the following:

```
P = Range[0, 2, (2-0)/10]
```

The `0` is a, the `2` is b, and the last number `2-0/10` is $\frac{b-a}{n}$, the width of each subinterval. In general, to split $[a, b]$ into n equal-length subintervals, execute

```
P = Range[a,b,(b-a)/n]
```

3. How to get to the individual numbers in a partition $\mathcal{P}$

Suppose you have defined a partition $\mathcal{P} = \{x_0, x_1, ..., x_n\}$ in *Mathematica*. To call one of the elements of $\mathcal{P}$, use double brackets as shown below. **There is a catch:** in handwritten math notation, we write our partitions starting with index 0. But *Mathematica* starts its partitions with index 1. So if $\mathcal{P} = \{0, 1, 5/2, 4, 7\}$ has been defined in *Mathematica*, executing

```
P[[3]]
```

generates the output $\frac{5}{2}$, which we think of as x_2, not x_3.

In general, once you have typed in a partition $\mathcal{P}$,

- execute `P[[j]]` to get the $(j-1)^{th}$ term x_{j-1}, and
- execute `P[[j+1]]` to get the j^{th} term x_j.

4. How to do sums (not necessarily Riemann sums) in *Mathematica*

Suppose you want to compute some sum which is written in $\Sigma-$notation. To do this, open the Basic Math Assistant pallette and click the $\left[d \int \sum\right]$ button (located under the phrase "Basic Commands"). In the first column of buttons, you will see a $\sum$ which you can click on to put a $\sum$ in your cell. You will get boxes to type all the pieces of the sum in.

5. An explanation of how to generate a Riemann sum for a function

First, remember that in any Riemann sum, $\Delta x_j = x_j - x_{j-1}$. From section 3 of this handout, in *Mathematica* this expression is `P[[j+1]] - P[[j]]`.

Next, suppose we are doing a left-hand sum. Then the test points c_j satisfy

$$\begin{aligned} c_j &= \text{ left endpoint of the } j^{th} \text{ subinterval} \\ &= \text{ left endpoint of } [x_{j-1}, x_j] \\ &= x_{j-1}. \end{aligned}$$

Therefore, $c_j = x_{j-1}$ should be `P[[j]]` in *Mathematica* code, and $f(c_j)$ is `f[ P[[j]] ]`.

Putting this together, the right *Mathematica* code for a left-hand Riemann sum (assuming you have defined your function `f` and your partition `P`) is

$$\sum_{\texttt{j = 1}}^{\texttt{n}} \texttt{f[P[[j]]] (P[[j + 1]] - P[[j]])}$$

6. The final commands for left- and right-hand Riemann sums

From the previous page, we came up with the following sequence of commands for computing a left-hand Riemann sum:

Syntax to compute a left-hand Riemann sum

To evaluate a left-hand Riemann sum, execute the following commands:

```
f[x_] = x^2
```
(or whatever your function is)

```
P = {0, 1/2, 3/4, 1}
```
(or whatever your partition is)

$\sum_{j=1}^{n}$ `f[ P[[j]] ] (P[[j+1]] - P[[j]])`
(n is the number of subintervals)

To evaluate a right-hand sum, the only thing that changes is the test point c_j, which goes from the left endpoint x_{j-1} (i.e. `P[[j]]`) to the right endpoint x_j (i.e. `P[[j+1]]`). Thus the commands for computing a right-hand Riemann sum are similar:

Syntax to compute a right-hand Riemann sum

To evaluate a right-hand Riemann sum, execute the following commands:

```
f[x_] = x^2
```
(or whatever your function is)

```
P = {0, 1/2, 3/4, 1}
```
(or whatever your partition is)

$\sum_{j=1}^{n}$ `f[ P[[j+1]] ] (P[[j+1]] - P[[j]])`
(n is the number of subintervals)