Lab 1

In this lab, we will practice with Mathematica the following topics:

- Basic arithmetic operations
- Define functions and sequences
- Visualize a sequence
- Evaluate partial sums and series

1 Getting access

There are two ways to get free access to Mathematica:

A) Install three free components: *Wolfram Engine*, *JupyterLab*, and *WolframLanguageForJupyter*. You can use the unlimited computing power of Mathematica on your own computer, with JupyterLab acting as a user interface. The instruction is here:

https://web.engr.oregonstate.edu/~phamt3/Resource/Wolfram-Mathematica-with-JupyterLab.pdf

B) Use the cloud-based version of Mathematica: https://www.wolframcloud.com
In this option, you are limited to about 8 minutes of computation per month. Files stored on the cloud will be deleted after 60 days.

2 First experiments

(1) Type `35/6`, then `Shift Enter`.

(2) Type `N[35/6]` (notice the square brackets), then `Shift Enter`.

(3) Type `Sqrt[2]` (notice the capitalized S), then `Shift Enter`.

(4) Type `N[%]`, then `Shift Enter`.

(5) Type `Sin[Pi]`, then `Shift Enter`. Next, try the same command but with lowercase S and/or P. Is Mathematica case sensitive?

(6) Type `34^100;` (with the semicolon), then `Shift Enter`.

(7) Type `34^100` (without semicolon), then `Shift Enter`.

The semicolon is to hold the output. One uses it when output is too long or not of interest. You may have noticed that the function `N` is to evaluate a numerical value of an expression. Each function's name is capitalized and used with square brackets (not with parentheses as we usually write on paper). For example, the function sin(x), cos(x), exp(x), ln(x) are written as `Sin[x]`, `Cos[x]`, `Exp[x]`, `Log[x]`, respectively.

(8) `Exp[1]`, then `Shift Enter`.

(9) `Log[2]`, then `Shift Enter`.

(10) Find a numerical value of $e^{2\cos(\sqrt{2})} + \ln 2$.

(11) `f[x_] := Sin[x]+Cos[x]` (notice the underscore after `x`), then `Shift Enter`.

(12) `f[Pi]+f[Pi/4]`, then `Shift Enter`.

(13) `Clear[f]`, then `Shift Enter`.

(14) `f[Pi]+f[Pi/4]`, then `Shift Enter`.

Command (11) is to define a function. An underscore is required when defining the function f. It is not needed when using the function. The function `Clear` is to remove a defined variable from the memory.

Next, let us plot functions of one variable, for example the sine function. Try the following commands:

(15) `Plot[Sin[x], {x,0,2*Pi}]`, then `Shift Enter`.

(16) For decoration, try

```
Plot[Sin[x], {x,0,2*Pi}, PlotStyle -> {Red, Dashed}]
```

Then `Shift+Enter`. Note that the arrow is typed as $->$.

(17) You can also give the function a name before plotting it. For example,

```
f[x_] := Sin[x];
Plot[f[x], {x,0,2*Pi}, Filling→Axis]
```

Then `Shift+Enter`. Note that the dash following x within the brackets is no longer used because f was already defined.

3 Description of sequences

(18) A sequence can be viewed as a function from $\mathbb{N}$ to $\mathbb{R}$. For example, the sequence $a_n = \frac{(-1)^{n+1}}{2^n}$ can be declared as follows.

```
a[n_]:=(-1)^(n+1)/2^n
```

(19) To view the first 10 terms of this sequence, try the command

```
Table[a[n], {n, 1, 10}]
```

(20) One can also "graph" a sequence

```
DiscretePlot[a[n], {n,1,10}]
```

On the graph, the horizontal axis shows the indices n and the vertical axis shows the values of a_n.

(21) A sequence can also be defined by a recursive formula. For example, the Fibonacci sequence $a_1 = 1$, $a_2 = 1$, $a_n = a_{n-1} + a_{n-2}$ can be declared as

```
a[1] := 1
a[2] := 1
a[n_] := a[n - 1] + a[n - 2]
```

To get a_{15}, use the command `a[15]`

(22) Find the term a_{20} of a sequence defined recursively as $a_0 = 1$, $a_1 = -1$, $a_2 = 2$, $a_n = a_{n-1}a_{n-2} + 2a_{n-3}$.

(23) Write the first 10 terms of this sequence. Hint: use the command `Table` as in Exercise (19).

4 Limit of a sequence

(24) If the general formula for a_n is available, you can simply use the command `Limit`. For example, to compute

$$\lim_{n\to\infty} \frac{2n^2 + n\sin(n)}{n^2+1}$$

try the following:

```
a[n_] := (2 n^2 + n*Sin[n])/(n^2 + 1)
Limit[a[n], n -> Infinity]
```

(25) Graph the above sequence using `DiscretePlot` to confirm the result.

(26) Problem 28 on page 440: find the limit of the sequence and then graph the sequence to confirm the result. Before you proceed, you can enter the command `Clear[a]` to erase the existing sequence $\{a_n\}$ from the memory.

(27) Problem 31 on page 440: find the limit of the sequence and then graph the sequence to confirm the result.

5 Sums and series

(28) To find the sum of finitely many terms (partial sum) or of infinitely many terms (series), we will the command `Sum`. Let us consider the series $\sum \frac{n^2}{2^n}$.

```
a[n_]:=n^2/2^n
Sum[a[n], {n, 1, 5}]
Sum[a[n], {n, 1, Infinity}]
```

(29) The k'th partial sum is

```
s[k_]:=Sum[a[n], {n, 1, k}]
```

(30) To get an idea what if the partial sums converge, you can write down 20 terms.

```
Table[s[k], {k, 1, 20}]
```

To see the decimal point numbers, put `N[...]` around `s[k]`.

```
Table[N[s[k]], {k, 1, 20}]
```

You can also graph the sequence of partial sums.

```
DiscretePlot[s[k], {k, 1, 20}]
```

(31) Check if the series

$$\sum_{n=1}^{\infty} \frac{1}{\ln(n+1)}$$

converges. If so, find its value.

(32) Check if the series

$$\sum_{n=2}^{\infty} \frac{2}{n^2-1}$$

converges. If so, find its value.

6 To turn in

Submit your implementation of Exercises (1) - (32) as a single pdf file.