Lab 1

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

- Basic arithmetic operations
- Find derivatives and antiderivatives
- Visualize the behavior of solutions to an autonomous ODE
- Solve some ODEs using `DSolve`

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 underscore following x within the brackets is no longer used because f was already defined.

3 Derivatives and antiderivatives

An ODE involves derivatives. Most methods to solve an ODE require the integration of an integral.

(18) Type `D[Cos[x],x]`, then `Shift Enter`.
Type `D[Cos[x],{x,2}]`, then `Shift Enter`.
Type `D[Cos[x],{x,3}]`, then `Shift Enter`.

(19) An alternative way to differentiate a function is the following

```
f[x_] := x^3 + x^2 + x + 1
f'[x]
f''[x]
```

(20) Find the third derivative of the function $f(x) = e^{\cos(x^2)}$.

(21) Check if the function $x(t) = t^2 + t^{-1}$ satisfies the ODE $x'' + tx' + x = t^2$.

(22) To find an antiderivative of a function, one uses the command `Integrate`.

```
Integrate[x^2,x]
f[x_]:=x-1
Integrate[f[x],x]
```

(23) Find an antiderivative of the function $f(x) = \cos(\sqrt{x})$.

4 Visualize the behavior of solutions

Consider an autonomous ODE $x' = f(x)$. Each initial state $x(0) = x_0$ gives us a solution $x = x(t)$. Visually, each initial state gives us a curve on the plane (the graph of function $x(t)$). There are infinitely many such curves on the plane. For each point (a, b) on the plane, the curve $x = x(t)$ that passes through (a, b) will pass through it at a slope of $x'(a) = f(x(a)) = f(b)$. Therefore, this curve is "directed" by the vector $(1, f(b))$ as it passes through (a, b). A collection of these direction vectors is called a *direction field*.

(24) On Mathematica, one can draw a direction field using the command `VectorPlot`. For example, a direction field of the ODE $x' = x^3 - 3x + 1$ is:

```
f[x_]:=x^3-3x+1
VectorPlot[{1,f[x]}, {t,-2,2}, {x,-3,3}]
```

(25) To "connect" these direction vectors to get a solution curve, we use the command `StreamPlot`.

```
StreamPlot[{1,f[x]}, {t,-2,2}, {x,-3,3}]
```

This diagram shows the behavior of solutions with different initial states x_0. How many equilibrium states do you observe? What are they approximately? Which of them are stable/unstable?

(26) Draw the behavior of solutions to the ODE $x' = \cos(x) - x^2$. How many equilibrium states do you observe? What are they approximately? Which of them are stable/unstable?

5 Solve an ODE using DSolve

(27) Many ODEs can be solved by the command `DSolve`. For example, the ODE $y' + y = e^x$ can be solved as follows:

```
DSolve[y'[x] + y[x] == Exp[x], y[x], x]
```

The double equal sign is used to indicate an *equation*. A single equal sign indicates an *assignment*.

(28) If the initial value is given, say $y(0) = 1$, then adjust the command as follows.

```
DSolve[{y'[x] + y[x] == Exp[x], y[0] == 1}, y[x], x]
```

(29) Do Exercise 8.1 (i) on page 72. Double check the result by substituting the solution back into the ODE. Then graph the solution.

(30) Do Exercise 8.1 (ii) on page 72. The function $\tan(x)$ is `Tan[x]` in Mathematica. Double check the result by substituting the solution back into the ODE. Then graph the solution.

6 To turn in

Submit your implementation of Exercises (1) - (30) as a single pdf file.