MOL518: Intro to Data Analysis

Homework 1

Due: _____

Name: _____ **NetID:** _____

Section/Preceptor (if applicable): _____

Instructions

- Submit a single PDF generated from this LaTeX file.
- Show your work clearly. When you write code, include the exact code you ran (copy/paste into a `lstlisting` block) and the relevant output.
- Unless stated otherwise, you may use standard Python libraries covered in Lecture 1–2: built-ins, NumPy, and basic plotting tools if introduced in your section.
- If you use AI tools, follow the course AI-use policy and cite what you used and how (briefly).

Problem 1: Thinking Like a Computer (Algorithms)

(10 points)

a) Write a step-by-step algorithm (plain English) for computing the mean (average) of a list of numbers.

b) Write a step-by-step algorithm (plain English) for finding the maximum value in a 1D array, without using any built-in `max` or NumPy convenience functions.

Problem 2: Variables, Types, and Common Errors

(15 points)

a) Explain the difference between the number `5` and the string `"5"`. Give one example of an operation that behaves differently for each.

b) The code below throws an error. Rewrite the code so it runs, and briefly explain what was wrong.

```
number = 5
text = "5"
print(text + number)
```

c) The code below throws an error. Rewrite the code so it runs, and briefly explain what was wrong.

```
radius = 2
pi = 3.14159
surface_area = 4 * Pi * radius**2
print("The sphere surface area is", surface_area)
```

Problem 3: NumPy Arrays (Indexing, Slicing, Shape)

(25 points)

a) Create a 1D NumPy array containing the odd integers from 1 to 21 (inclusive).

b) Print every 3rd element of that array (starting from the first element of the array).

c) How many elements remain if you remove every 3rd element? Show your reasoning and/or code.

d) Create the following 2D array as a NumPy array:

0	0	1	0	0
0	2	0	2	0
3	0	0	0	3
0	2	0	2	0
0	0	1	0	0

e) Print the 3rd row (be careful about Python indexing).

f) Compute the total number of elements in the array using its `.shape`.

g) Replace the last row with `[99, 99, 99, 99, 99]` and print the modified array.

Problem 4: Loading and Inspecting Data (CSV)

(25 points)

In Lecture 2 you loaded a growth curve CSV file into a NumPy array.

a) Load `Lecture_2/data/growth_curve1.csv` using `np.loadtxt(..., delimiter=",")`. Include the code you used.

b) Print the shape of the loaded array.

c) Print the first 5 rows and the last row.

d) Extract the time column and OD column into 1D arrays called `time` and `od`.

e) Compute the total duration of the experiment in hours.

Problem 5: Saving Derived Data

(15 points)

a) Create a normalized OD array `od_norm` by dividing all OD values by the first OD measurement.

b) Create a 2D array with time in hours and `od_norm` as columns.

c) Save the result as a new CSV file (do *not* overwrite the original). Name it `growth_curve1_hr_norm.csv` and place it alongside the original in the same `data/` directory.

Optional (Extra Credit): Two Files, Lists, and Change in OD

(+5 points)

Load both `growth_curve1.csv` and `growth_curve2.csv` using a Python list of filenames. For each dataset, compute the change in OD from the first to last time point. Print the two values.

Checklist before submitting:

- PDF compiles without errors.
- All questions answered and labeled.
- Code included where requested.
- Output/units reported where relevant.