

APPENDIX B: USE OF AI TOOL(S) IN PROJECT WORK

Each team member should indicate either A or B.

- A. I affirm that my contribution(s) to the lab work is my own, produced without help from any AI tool(s)
- B. I affirm that my contribution(s) to the lab work has been produced with the help from AI tool(s)

Full Name	Date	A or B
Vikas Vittanala	2025.11.7	A
Zhang Zexuan	2025.11.7	A
Tu Jiani	2025.11.7	A
Vaid Ramith	2025.11.7	B
Xu Hanwen	2025.11.7	B

By including this information in your jupyter notebook, you declare that the above affirmation made is true and that you have read and understood NTU's policy on the use of AI tools.

If any team member answered B, the team member(s) must indicate and replicate the table below for every instance AI tool(s) is used.

Name of AI tool	DeepSeek
Input prompt	draw a bar graph with x-axis: schools (obtained from the 'schools' column in records.csv), y-axis: number of students in each tutorial group (obtained from 'tut grp' column in 'records.csv') represent the gender statistics in a bar graph and it should be an individual function to display the results, I only need the number of male and female students in each tutorial group of 50
Date generated	Oct. 31
Output generated	<pre>import matplotlib.pyplot as plt import csv from collections import defaultdict # Read data from records.csv file school_tutorial_counts = defaultdict(lambda: defaultdict(int)) with open('records.csv', 'r', encoding='utf-8') as file: reader = csv.DictReader(file) for row in reader: school = row['School'] tutorial_group = row['Tutorial Group'] school_tutorial_counts[school][tutorial_group] += 1 # Calculate number of tutorial groups per school</pre>

```
schools = []
num_tutorial_groups = []

import matplotlib.pyplot as plt import numpy as np
def plot_gender_statistics (gender_counts):
    un"Create a bar graph showing male and female counts for each
    tutorial group.
    Args:
    gender_counts: Dictionary returned by
    count_gender_by_tutorial_group ()
    if not gender_counts:
    print ("No data available for plotting.")
    return
    # Sort tutorial groups for consistent ordering
    sorted_groups = sorted (gender_counts. keys ())
```

Output
screenshot

Save Student Group Data to CSV File

represent the gender statistics in a bar graph and it should be an individual function to display the results, I only need the number of male and female students in each tutorial group of 50

Here's a function to create a bar graph showing gender statistics for each tutorial group:

python

```
import matplotlib.pyplot as plt
import numpy as np

def plot_gender_statistics(gender_counts):
    """Create a bar graph showing male and female counts for each tutorial group.

    Args:
        gender_counts: Dictionary returned by count_gender_by_tutorial_group()
    """
    if not gender_counts:
        print("No data available for plotting.")
        return

    # Sort tutorial groups for consistent ordering
    sorted_groups = sorted(gender_counts.keys())

    # Extract data for plotting
```

Message DeepSeek

DeepThink Search

Analyzing GPA Distribution in Student Groups

draw a bar graph with x-axis: schools (obtained from the 'schools' column in records.csv), y-axis: number of students in each tutorial group (obtained from 'tut grp' column in 'records.csv')

python

```
import matplotlib.pyplot as plt
import csv
from collections import defaultdict

# Read data from records.csv file
school_tutorial_counts = defaultdict(lambda: defaultdict(int))

with open('records.csv', 'r', encoding='utf-8') as file:
    reader = csv.DictReader(file)
    for row in reader:
        school = row['School']
        tutorial_group = row['Tutorial Group']
        school_tutorial_counts[school][tutorial_group] += 1

# Calculate number of tutorial groups per school
schools = []
num_tutorial_groups = []
```

Message DeepSeek

DeepThink Search

AI-generated, for reference only

Impact on submission	Allow us to use python to create a graph to demonstrate diversity.
----------------------	--

Name of AI tool	ChatGPT
Input prompt	how to sort rows of csv by a certain characteristic (column value) in python without using pandas
Date generated	Oct. 25
Output generated	<pre>import csv # Step 1: Read the CSV with open ("data.csv", newline="") as f: reader = csv.DictReader(f) rows = list(reader) # convert to list of dicts # Step 2: Sort by a specific column (say "age") # If "age" is numeric, cast it to int for correct ordering rows.sort (key=lambda row: int (row["age"])) # Step 3: Write back to a new CSV file with open("sorted_data.csv", "w", newline="") as f: writer = csv.DictWriter(f, fieldnames=reader. fieldnames) writer .writeheader () writer.writerows (rows)</pre>

Output screenshot	 <p>how to sort rows of csv by a certain characteristic (column) in python without using pandas</p> <p>Perfect — here's how to do it without using pandas, just with the built-in csv module.</p> <p>✅ Example: Sort a CSV file by a column (e.g. "age")</p> <pre>python import csv # Step 1: Read the CSV with open("data.csv", newline='') as f: reader = csv.DictReader(f) rows = list(reader) # convert to list of dicts # Step 2: Sort by a specific column (say "age") # If "age" is numeric, cast it to int for correct ordering rows.sort(key=lambda row: int(row["age"])) # Step 3: Write back to a new CSV file with open("sorted_data.csv", "w", newline='') as f: writer = csv.DictWriter(f, fieldnames=reader.fieldnames) writer.writeheader() writer.writerows(rows)</pre>
Impact on submission	Allow us do sorting using the lambda function.