assignment_one

March 4, 2023

0.1 Timothy Miller

0.2 GTECH 73100, Dr. Sun

1 Assignment One

Basics

1.1 Import modules

Math and Numpy are used only to verify the custom implementations

[ ]:
```
import math
import platform
import random
import numpy as np
```

1.2 Task 1

Get the Python version on your computer

[ ]:
```
print(f"python version for platform {platform.python_version()}")
```

```
python version for platform 3.11.1
```

1.3 Task 2, part one

Generate a list of random numbers user `random` module in Python

[ ]:
```
rand_ints_count = 20
rand_min = 1
rand_max = 50
rand_ints = [None] * rand_ints_count

for i in range (rand_ints_count):
    rand_ints[i] = random.randint(rand_min, rand_max)

print(f"random integers {rand_ints}")
```

```
random integers [25, 39, 26, 38, 47, 38, 10, 44, 48, 13, 33, 46, 46, 3, 32, 16,
49, 38, 27, 47]
```

1.4 Task 2, part two

Calculate the mean value for the list by looping through its elements

[ ]:
```
rand_ints_total = 0
for rand_int in rand_ints:
    rand_ints_total += rand_int

rand_ints_avg = rand_ints_total / rand_ints_count
print(f"mean value for the randomt integers {rand_ints_avg}")


"""Check the custom mean implementation against the numpy implementation
"""
assert rand_ints_avg == np.average(rand_ints)
```

```
mean value for the randomt integers 33.25
```

1.5 Task 2, part three

Calculate the variance of the list

[ ]:
```
rand_ints_sqrd_difs = 0
for rand_int in rand_ints:
  rand_ints_sqrd_difs += pow(rand_int - rand_ints_avg, 2)

rand_ints_var = rand_ints_sqrd_difs / rand_ints_count
print(f"variance of random integers {rand_ints_var}")

"""Check the custom implementation against the numpy implementation

Only take the integer to account for rounding errors
"""
assert math.trunc(rand_ints_var) == math.trunc(np.var(rand_ints))
```

```
variance of random integers 185.4875
```