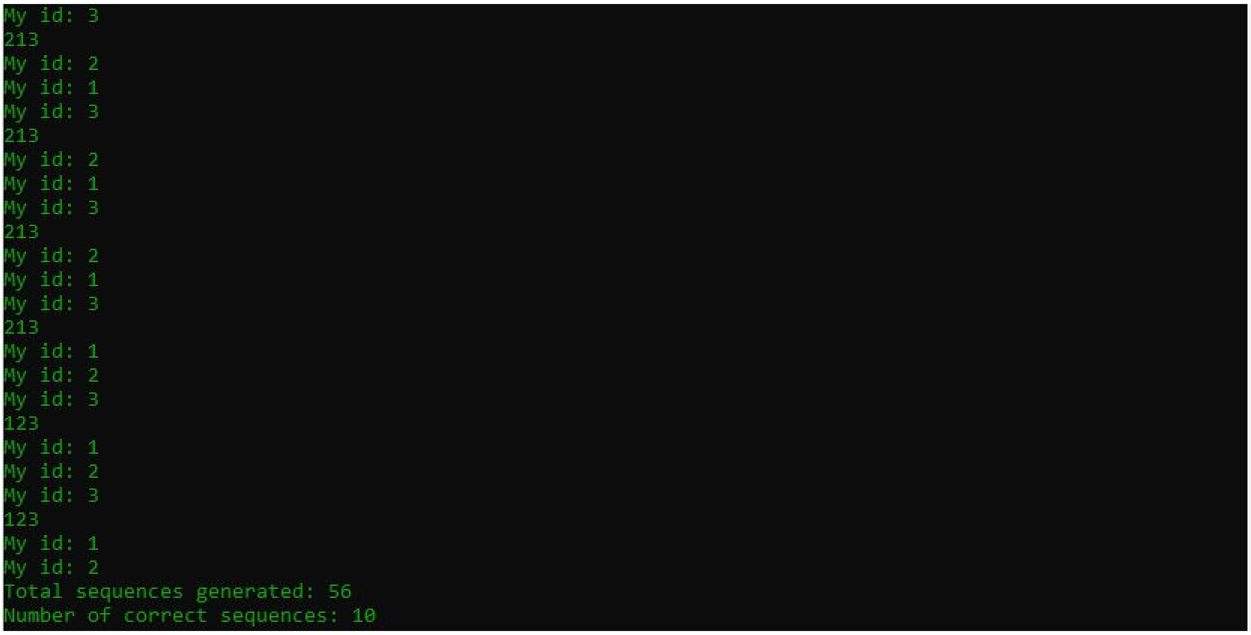
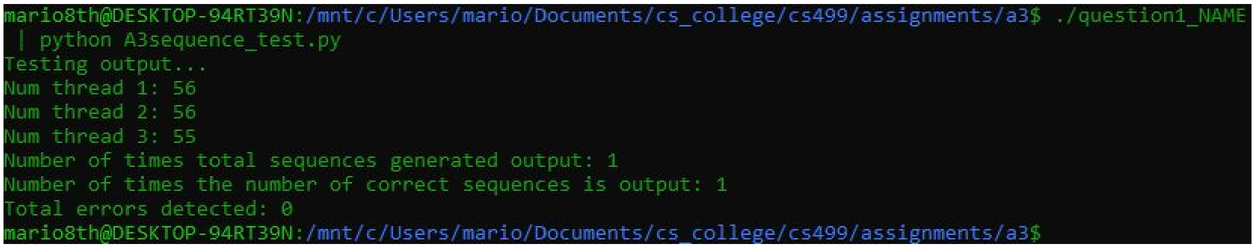
Gavin Valencia
CS499- Intro to Parallel Programming
10/8/19

Question 1:
(code output)



```
My id: 3
213
My id: 2
My id: 1
My id: 3
213
My id: 2
My id: 1
My id: 3
213
My id: 2
My id: 1
My id: 3
213
My id: 1
My id: 2
My id: 3
123
My id: 1
My id: 2
My id: 3
123
My id: 1
My id: 2
Total sequences generated: 56
Number of correct sequences: 10
```

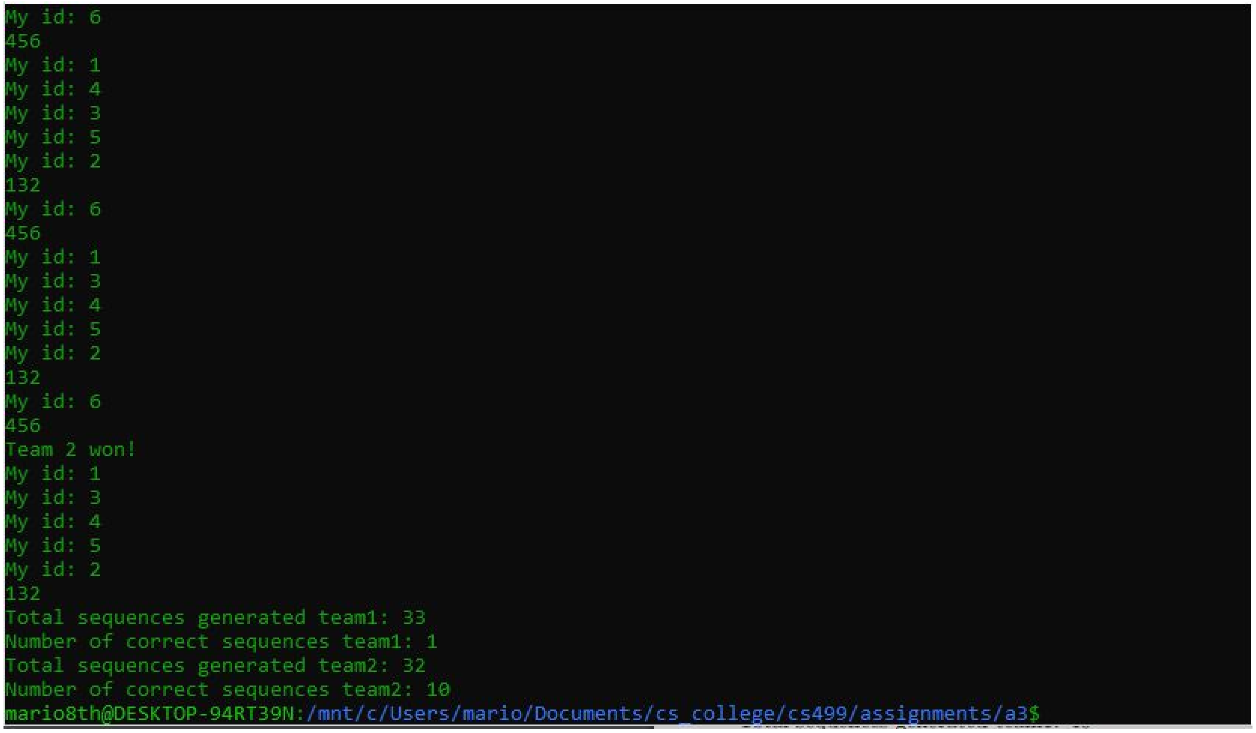
(python output)



```
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$ ./question1_NAME
 | python A3sequence_test.py
Testing output...
Num thread 1: 56
Num thread 2: 56
Num thread 3: 55
Number of times total sequences generated output: 1
Number of times the number of correct sequences is output: 1
Total errors detected: 0
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$
```

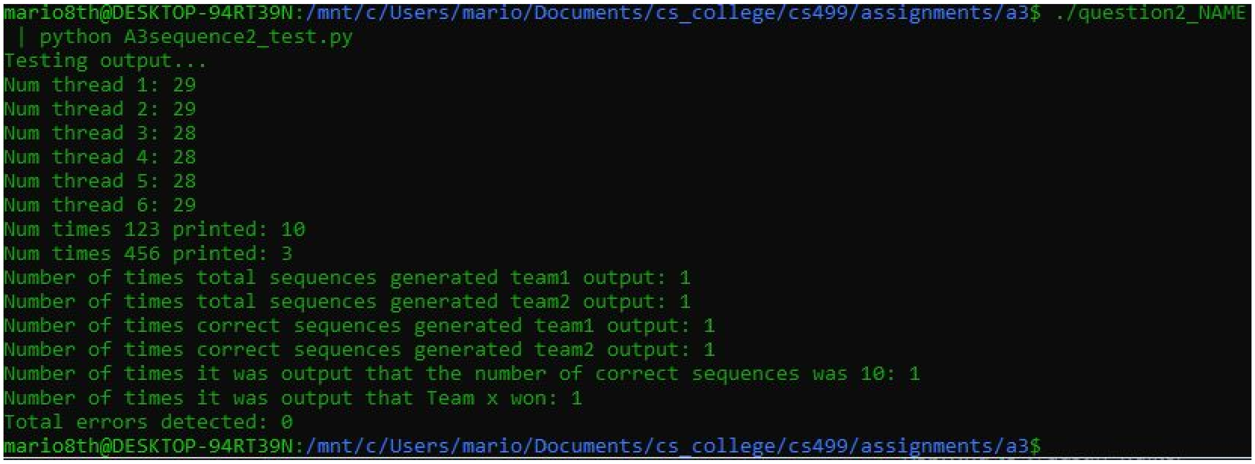
This program generates a series of numbers and displays it, saving it as correct if the sequence is ‘123’. It can take many tries for the pthreads to cycle to the correct output, and usually thread 1 and 2 generate a new number before realizing the win condition has been met. However, sometimes it can take a very long time if thread1 falls into the third place and cannot move back up.

Question 2:
(code output)



```
My id: 6
456
My id: 1
My id: 4
My id: 3
My id: 5
My id: 2
132
My id: 6
456
My id: 1
My id: 3
My id: 4
My id: 5
My id: 2
132
My id: 6
456
Team 2 won!
My id: 1
My id: 3
My id: 4
My id: 5
My id: 2
132
Total sequences generated team1: 33
Number of correct sequences team1: 1
Total sequences generated team2: 32
Number of correct sequences team2: 10
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$
```

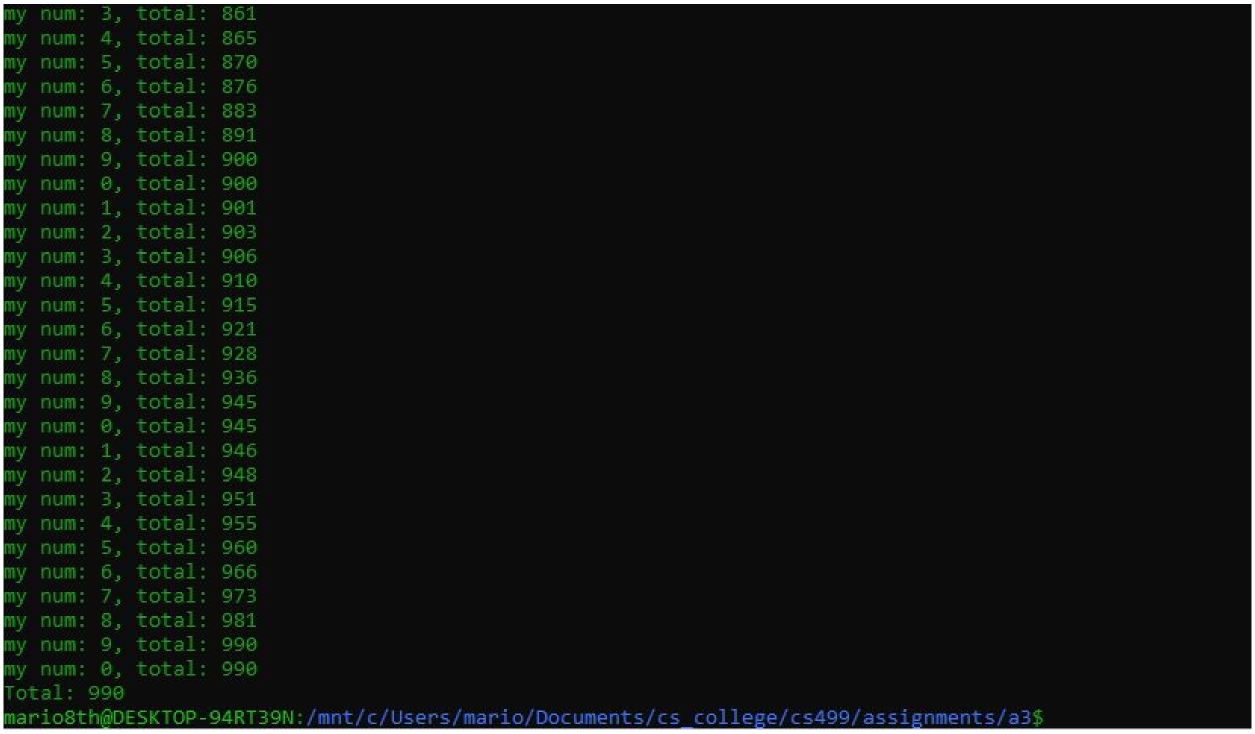
(python output)



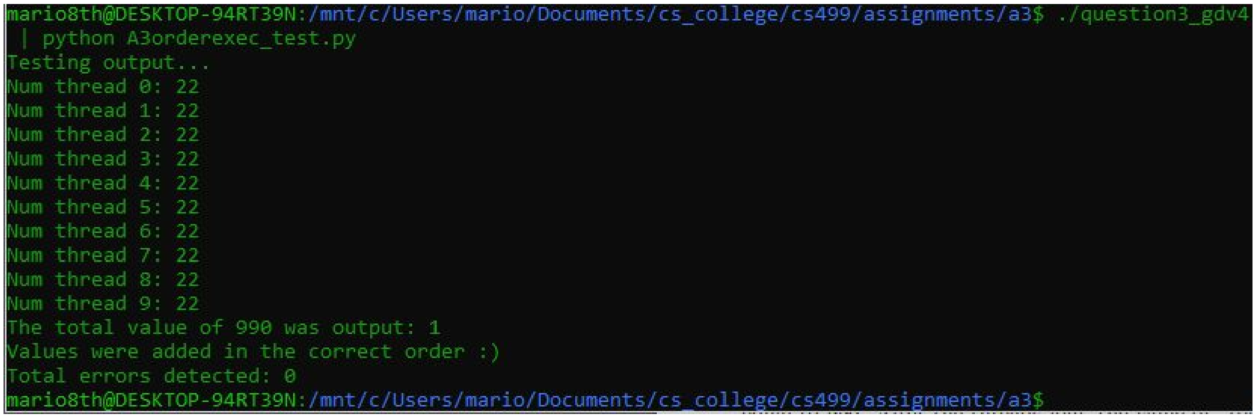
```
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$ ./question2_NAME
 | python A3sequence2_test.py
Testing output...
Num thread 1: 29
Num thread 2: 29
Num thread 3: 28
Num thread 4: 28
Num thread 5: 28
Num thread 6: 29
Num times 123 printed: 10
Num times 456 printed: 3
Number of times total sequences generated team1 output: 1
Number of times total sequences generated team2 output: 1
Number of times correct sequences generated team1 output: 1
Number of times correct sequences generated team2 output: 1
Number of times it was output that the number of correct sequences was 10: 1
Number of times it was output that Team x won: 1
Total errors detected: 0
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$
```

This program shares mutex locks for both do_works, and the first to generate their respective number sequence (‘123’ for team 1, and ‘456’ for team 2) it sets a flag for all threads to quit and outputs the total sequences. After the flag is set, a number of the threads still start generating a new sequence before quitting.

Question 3:



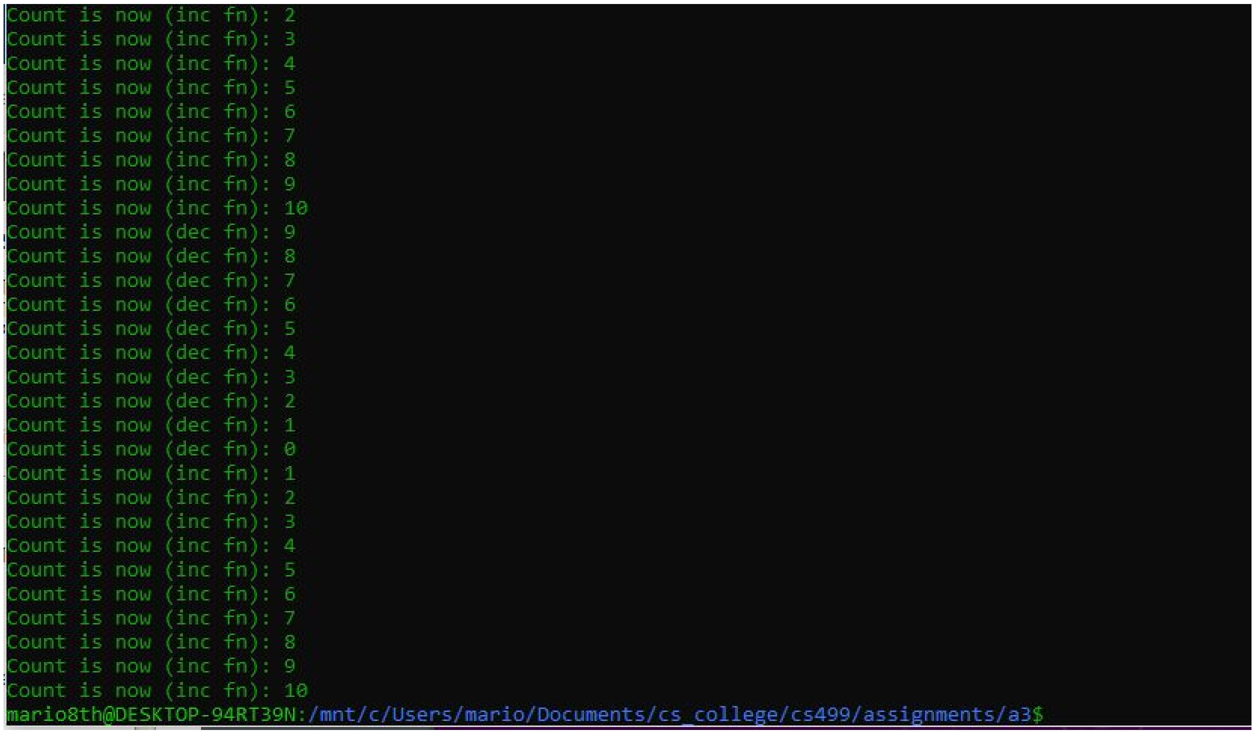
```
my num: 3, total: 861
my num: 4, total: 865
my num: 5, total: 870
my num: 6, total: 876
my num: 7, total: 883
my num: 8, total: 891
my num: 9, total: 900
my num: 0, total: 900
my num: 1, total: 901
my num: 2, total: 903
my num: 3, total: 906
my num: 4, total: 910
my num: 5, total: 915
my num: 6, total: 921
my num: 7, total: 928
my num: 8, total: 936
my num: 9, total: 945
my num: 0, total: 945
my num: 1, total: 946
my num: 2, total: 948
my num: 3, total: 951
my num: 4, total: 955
my num: 5, total: 960
my num: 6, total: 966
my num: 7, total: 973
my num: 8, total: 981
my num: 9, total: 990
my num: 0, total: 990
Total: 990
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$
```



```
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$ ./question3_gdv4
 | python A3orderexec_test.py
Testing output...
Num thread 0: 22
Num thread 1: 22
Num thread 2: 22
Num thread 3: 22
Num thread 4: 22
Num thread 5: 22
Num thread 6: 22
Num thread 7: 22
Num thread 8: 22
Num thread 9: 22
The total value of 990 was output: 1
Values were added in the correct order :)
Total errors detected: 0
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$
```

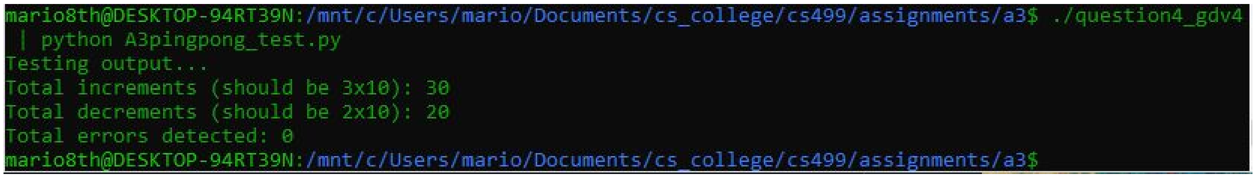
This program goes through and checks that the previous thread that went was the one it is supposed to go after, adds it’s number to a total sum, and then lets the next thread go. When the total gets to 990, all the threads exit the loop and return.

Question 4:
(code output)



```
Count is now (inc fn): 2
Count is now (inc fn): 3
Count is now (inc fn): 4
Count is now (inc fn): 5
Count is now (inc fn): 6
Count is now (inc fn): 7
Count is now (inc fn): 8
Count is now (inc fn): 9
Count is now (inc fn): 10
Count is now (dec fn): 9
Count is now (dec fn): 8
Count is now (dec fn): 7
Count is now (dec fn): 6
Count is now (dec fn): 5
Count is now (dec fn): 4
Count is now (dec fn): 3
Count is now (dec fn): 2
Count is now (dec fn): 1
Count is now (dec fn): 0
Count is now (inc fn): 1
Count is now (inc fn): 2
Count is now (inc fn): 3
Count is now (inc fn): 4
Count is now (inc fn): 5
Count is now (inc fn): 6
Count is now (inc fn): 7
Count is now (inc fn): 8
Count is now (inc fn): 9
Count is now (inc fn): 10
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$
```

(python output)



```
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$ ./question4_gdv4
 | python A3pingpong_test.py
Testing output...
Total increments (should be 3x10): 30
Total decrements (should be 2x10): 20
Total errors detected: 0
mario8th@DESKTOP-94RT39N:/mnt/c/Users/mario/Documents/cs_college/cs499/assignments/a3$
```

This code has two threads, one that increments a counter, and one that decrements. If the decrement counter gets to the lock first, it checks if the counter is at zero and then lets increment go. Increment counts until it gets to 10, and then signals decrement to go. When the total decrements hits 20 the decrement thread signals one last time and exits, and when the total increments reaches 30 it exits.