CSE 489/589
Programming Assignment 2 Report
Reliable Transport Protocols

Notes: (IMPORTANT)

- ➔ One of your group members select <File> - <Make a copy> to make a copy of this report for your group, and share that Google Doc copy with your teammates so that they can also edit it.
- ➔ Report your work in each section. Describe the method you used, the obstacles you met, how you solved them, and the results. You can take screenshots at key points. There are NO hard requirements for your description.
- ➔ For a certain test, if you successfully implemented it, **take a screenshot of the result from the grader as required in section 5 (required).** You can just provide the overall result for each test.
- ➔ For a certain test, if you tried but failed to implement it, properly describe your work. We will partially grade it based on the work you did.
- ➔ Do **NOT** claim anything you didn't implement. If you didn't try on a certain protocol or test, leave that section blank. We will run your code, and if it does not match the work you claimed, you and your group won't get any partial grade score for this WHOLE assignment.
- ➔ There will be **15.0** points for this report. These are NOT bonus points and will be given based on the completion of the analysis part (section 6.1).
- ➔ If you decide not to attempt the analysis part (section 6.1) of the assignment, you will still NEED to submit this report with the requirements stated in section 6.
- ➔ Under **NO** circumstances may you rely on the work of your peers, including but not limited to GitHub repositories or code submissions from previous academic terms.
- ➔ All the analysis results in section 6 should come from one of the provided hosts, NOT on your local machine (see section 3.1 in the handout).
- ➔ The maximum score for PA 2: 85 + 15 = 100

1 - Academic Integrity Policy Statement

I have read and understood the course's academic integrity policy.

[Your submission will NOT be graded without this statement.]

2 - Group and Contributions

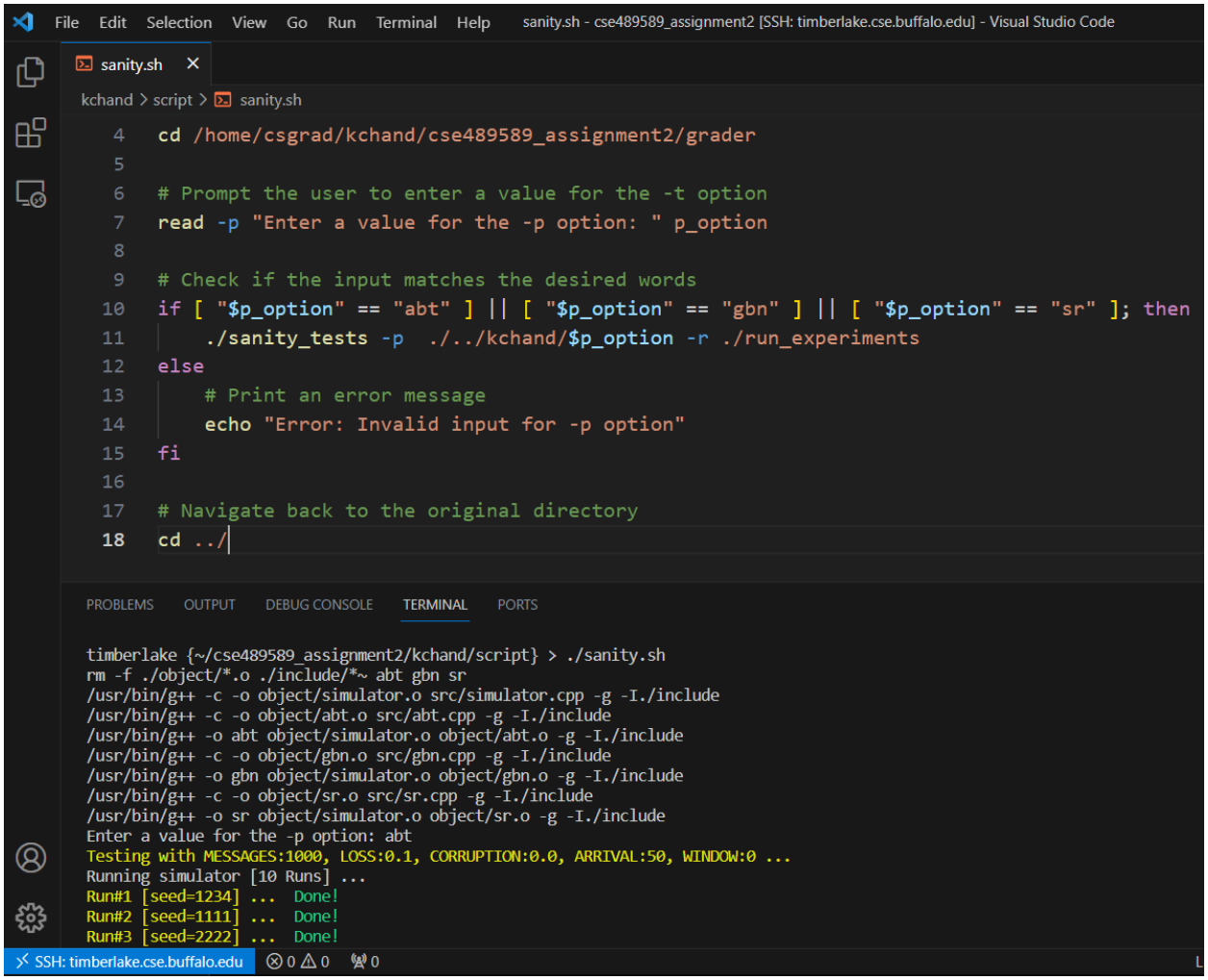
- Name of member 1:
  - UBITName: **kchand**
  - Contributions: gbn.cpp, sr.cpp

- Name of member 2:
    - UBITName: **rahulyad**
    - Contributions: abt.cpp, sr.cpp timer

3 - SANITY Tests

[2.0] ABT

(Put screenshots of the grader here…)



```
Enter a value for the -p option: abt
Testing with MESSAGES:1000, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.2, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.4, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.6, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.8, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.1, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.2, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.4, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

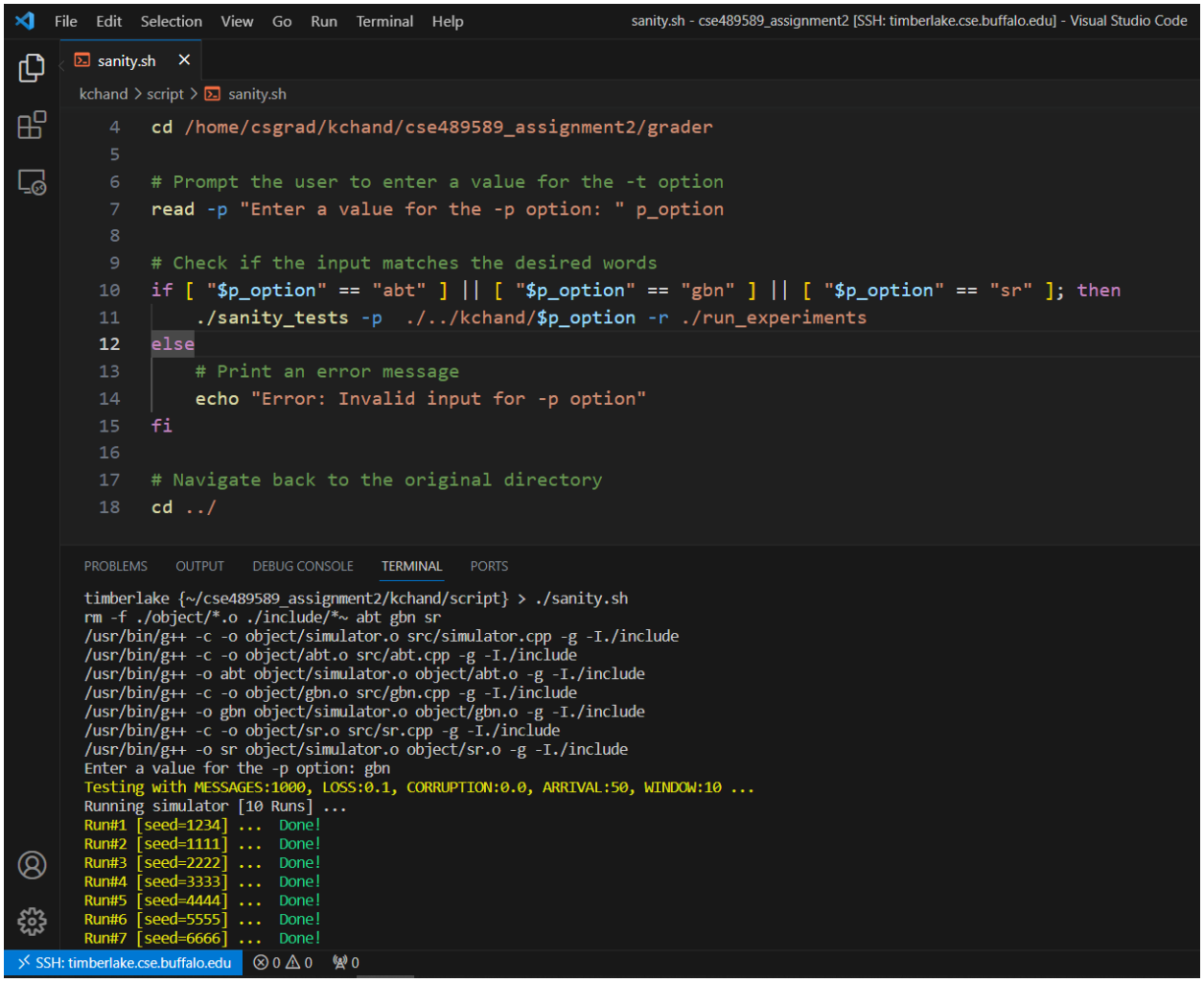
```
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.6, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.8, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:20, LOSS:0.0, CORRUPTION:0.0, ARRIVAL:1000, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:20, LOSS:1.0, CORRUPTION:0.0, ARRIVAL:1000, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:20, LOSS:0.0, CORRUPTION:1.0, ARRIVAL:1000, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
SANITY TESTS:  PASS
timberlake {~/cse489589_assignment2/kchand/script} >
SSH: timberlake.cse.buffalo.edu  ⊗ 0 ⚠ 0  0
```

[5.0] GBN

(Put screenshots of the grader here…)


File Edit Selection View Go Run Terminal Help
sanity.sh - cse489589_assignment2 [SSH: timberlake.cse.buffalo.edu] - Visual Studio Code
sanity.sh
kchand > script > sanity.sh
4 cd /home/csgrad/kchand/cse489589_assignment2/grader
5
6 # Prompt the user to enter a value for the -t option
7 read -p "Enter a value for the -p option: " p_option
8
9 # Check if the input matches the desired words
10 if ["$p_option" == "abt"] || ["$p_option" == "gbn"] || ["$p_option" == "sr"]; then
11 ./sanity_tests -p ./../kchand/$p_option -r ./run_experiments
12 else
13 # Print an error message
14 echo "Error: Invalid input for -p option"
15 fi
16
17 # Navigate back to the original directory
18 cd ../
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
timberlake {~/cse489589_assignment2/kchand/script} > ./sanity.sh
rm -f ./object/*.o ./include/*~ abt gbn sr
/usr/bin/g++ -c -o object/simulator.o src/simulator.cpp -g -I./include
/usr/bin/g++ -c -o object/abt.o src/abt.cpp -g -I./include
/usr/bin/g++ -o abt object/simulator.o object/abt.o -g -I./include
/usr/bin/g++ -c -o object/gbn.o src/gbn.cpp -g -I./include
/usr/bin/g++ -o gbn object/simulator.o object/gbn.o -g -I./include
/usr/bin/g++ -c -o object/sr.o src/sr.cpp -g -I./include
/usr/bin/g++ -o sr object/simulator.o object/sr.o -g -I./include
Enter a value for the -p option: gbn
Testing with MESSAGES:1000, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ... Done!
Run#2 [seed=1111] ... Done!
Run#3 [seed=2222] ... Done!
Run#4 [seed=3333] ... Done!
Run#5 [seed=4444] ... Done!
Run#6 [seed=5555] ... Done!
Run#7 [seed=6666] ... Done!
SSH: timberlake.cse.buffalo.edu
0 0
0

```
Enter a value for the -p option: gbn
Testing with MESSAGES:1000, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.2, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.4, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.6, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.8, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.1, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.2, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.4, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

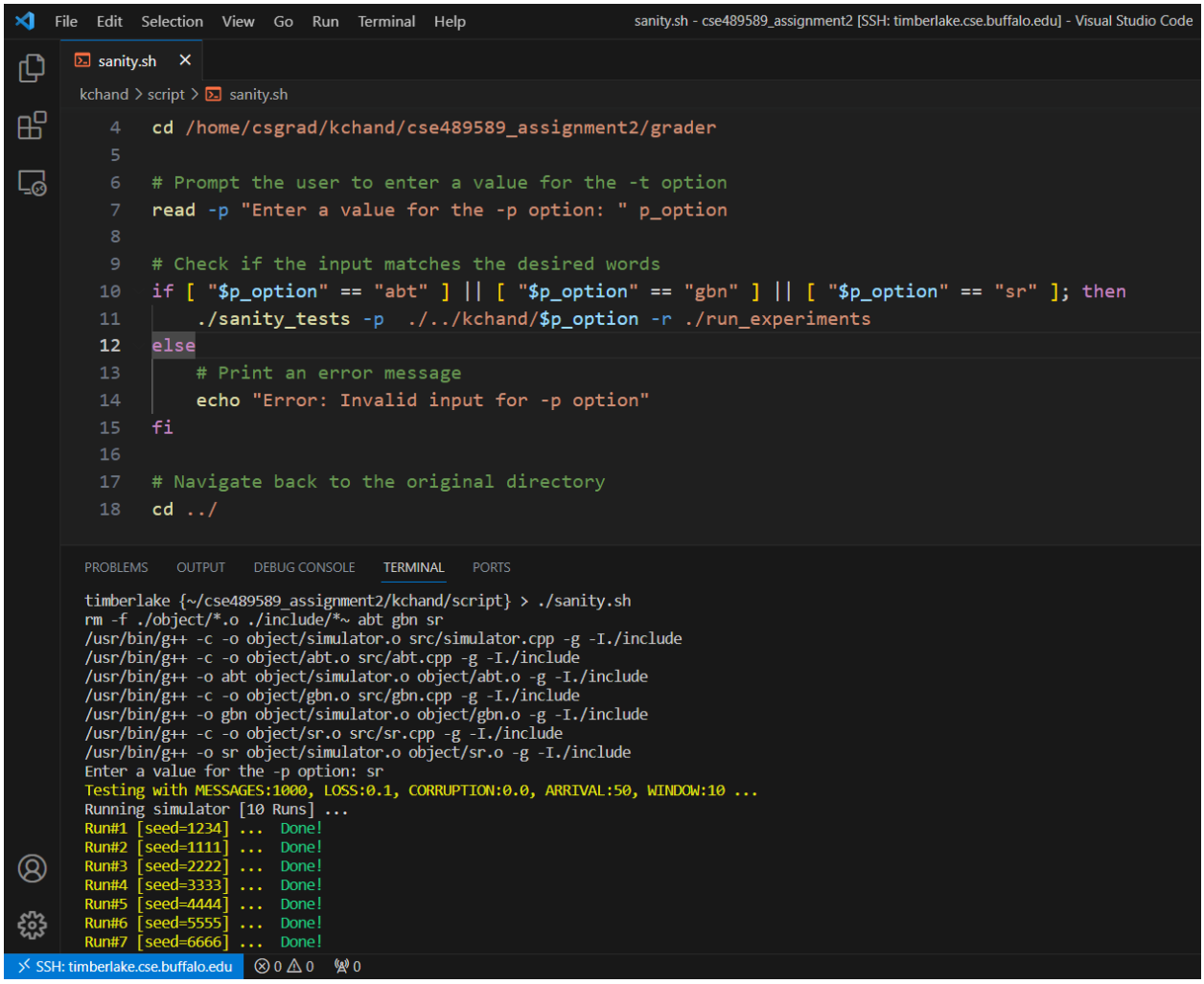
```
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.6, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.8, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:20, LOSS:0.0, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:20, LOSS:1.0, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
PASS!
Testing with MESSAGES:20, LOSS:0.0, CORRUPTION:1.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
SANITY TESTS:  PASS
timberlake {~/cse489589_assignment2/kchand/script} >
SSH: timberlake.cse.buffalo.edu  ⊗ 0 ⚠ 0  0
```

[8.0] SR

(Put screenshots of the grader here…)


File Edit Selection View Go Run Terminal Help
sanity.sh - cse489589_assignment2 [SSH: timberlake.cse.buffalo.edu] - Visual Studio Code
sanity.sh
kchand > script > sanity.sh
4 cd /home/csgrad/kchand/cse489589_assignment2/grader
5
6 # Prompt the user to enter a value for the -t option
7 read -p "Enter a value for the -p option: " p_option
8
9 # Check if the input matches the desired words
10 if ["$p_option" == "abt"] || ["$p_option" == "gbn"] || ["$p_option" == "sr"]; then
11 ./sanity_tests -p ./../kchand/$p_option -r ./run_experiments
12 else
13 # Print an error message
14 echo "Error: Invalid input for -p option"
15 fi
16
17 # Navigate back to the original directory
18 cd ../
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
timberlake {~/cse489589_assignment2/kchand/script} > ./sanity.sh
rm -f ./object/*.o ./include/*~ abt gbn sr
/usr/bin/g++ -c -o object/simulator.o src/simulator.cpp -g -I./include
/usr/bin/g++ -c -o object/abt.o src/abt.cpp -g -I./include
/usr/bin/g++ -o abt object/simulator.o object/abt.o -g -I./include
/usr/bin/g++ -c -o object/gbn.o src/gbn.cpp -g -I./include
/usr/bin/g++ -o gbn object/simulator.o object/gbn.o -g -I./include
/usr/bin/g++ -c -o object/sr.o src/sr.cpp -g -I./include
/usr/bin/g++ -o sr object/simulator.o object/sr.o -g -I./include
Enter a value for the -p option: sr
Testing with MESSAGES:1000, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ... Done!
Run#2 [seed=1111] ... Done!
Run#3 [seed=2222] ... Done!
Run#4 [seed=3333] ... Done!
Run#5 [seed=4444] ... Done!
Run#6 [seed=5555] ... Done!
Run#7 [seed=6666] ... Done!
SSH: timberlake.cse.buffalo.edu
0 0
0

```
Enter a value for the -p option: sr
Testing with MESSAGES:1000, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.2, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.4, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.6, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.8, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.1, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.2, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.4, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.6, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.8, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:20, LOSS:0.0, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:20, LOSS:1.0, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

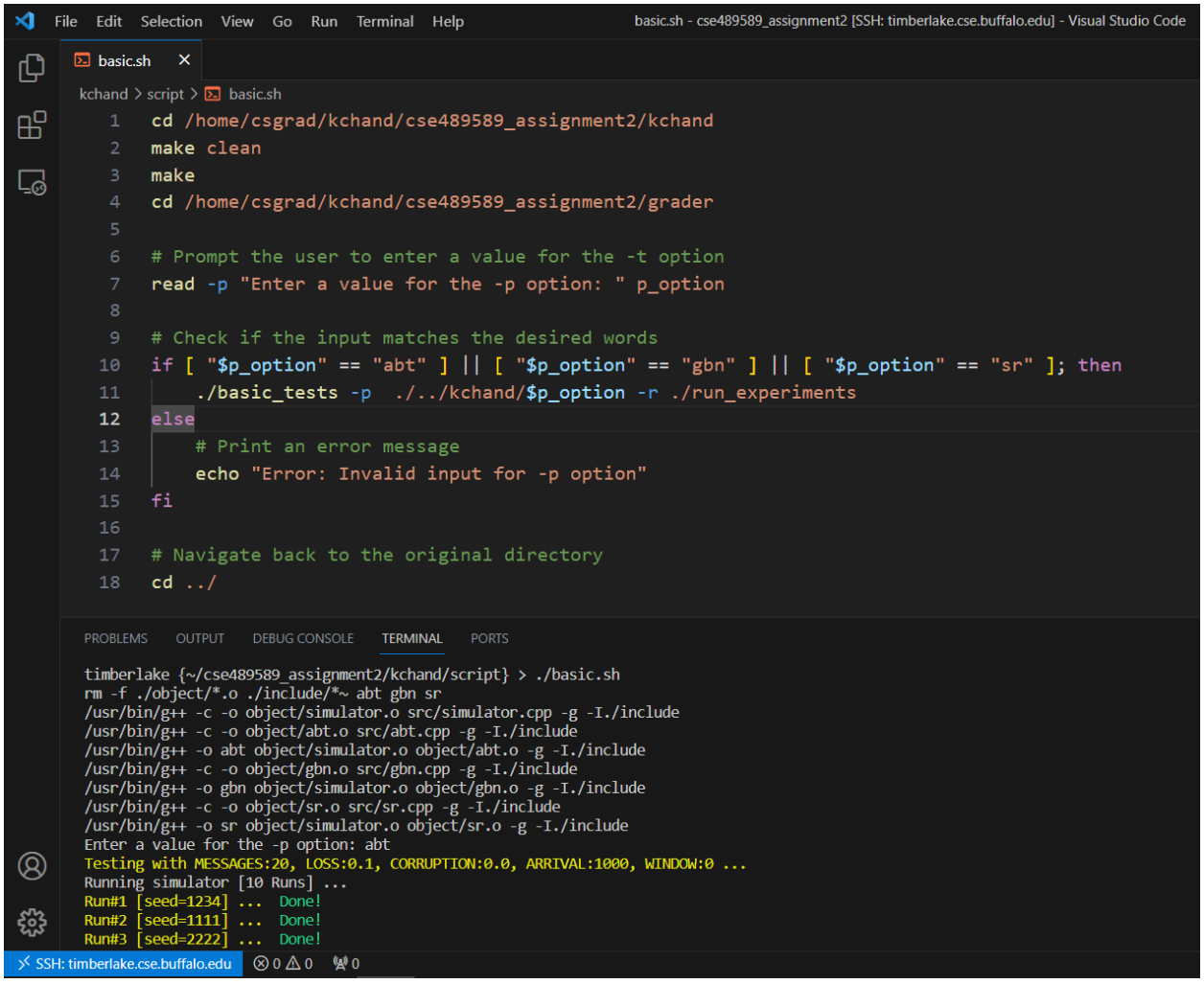
```
Testing with MESSAGES:20, LOSS:0.0, CORRUPTION:1.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
SANITY TESTS:  PASS
timberlake {~/cse489589_assignment2/kchand/script} >
SSH: timberlake.cse.buffalo.edu  ⊗ 0 ⚠ 0  0
```

[No further grading for the protocol that fails a SANITY test.]

4 - BASIC Tests

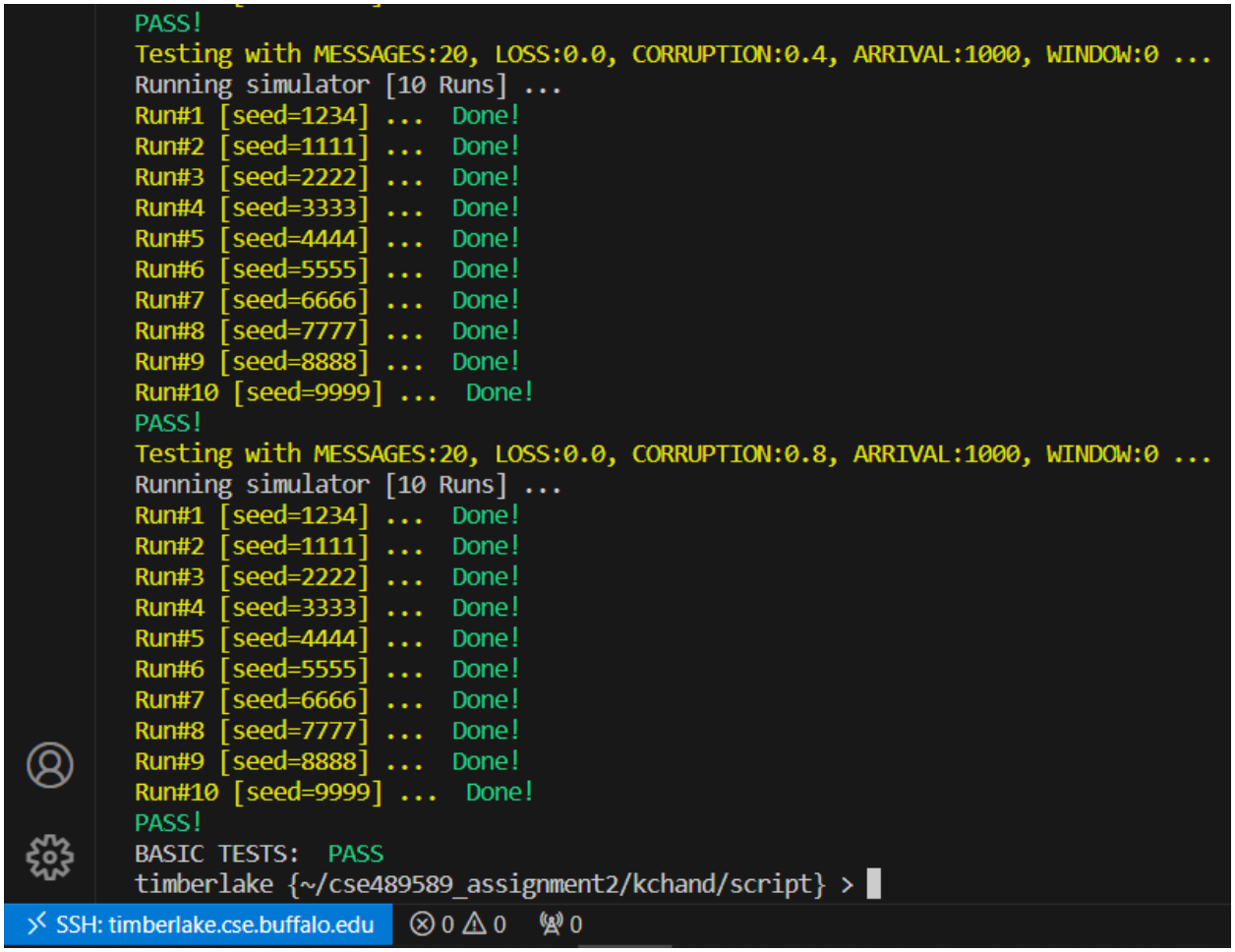
[5.0] ABT

(Put screenshots of the grader here…)



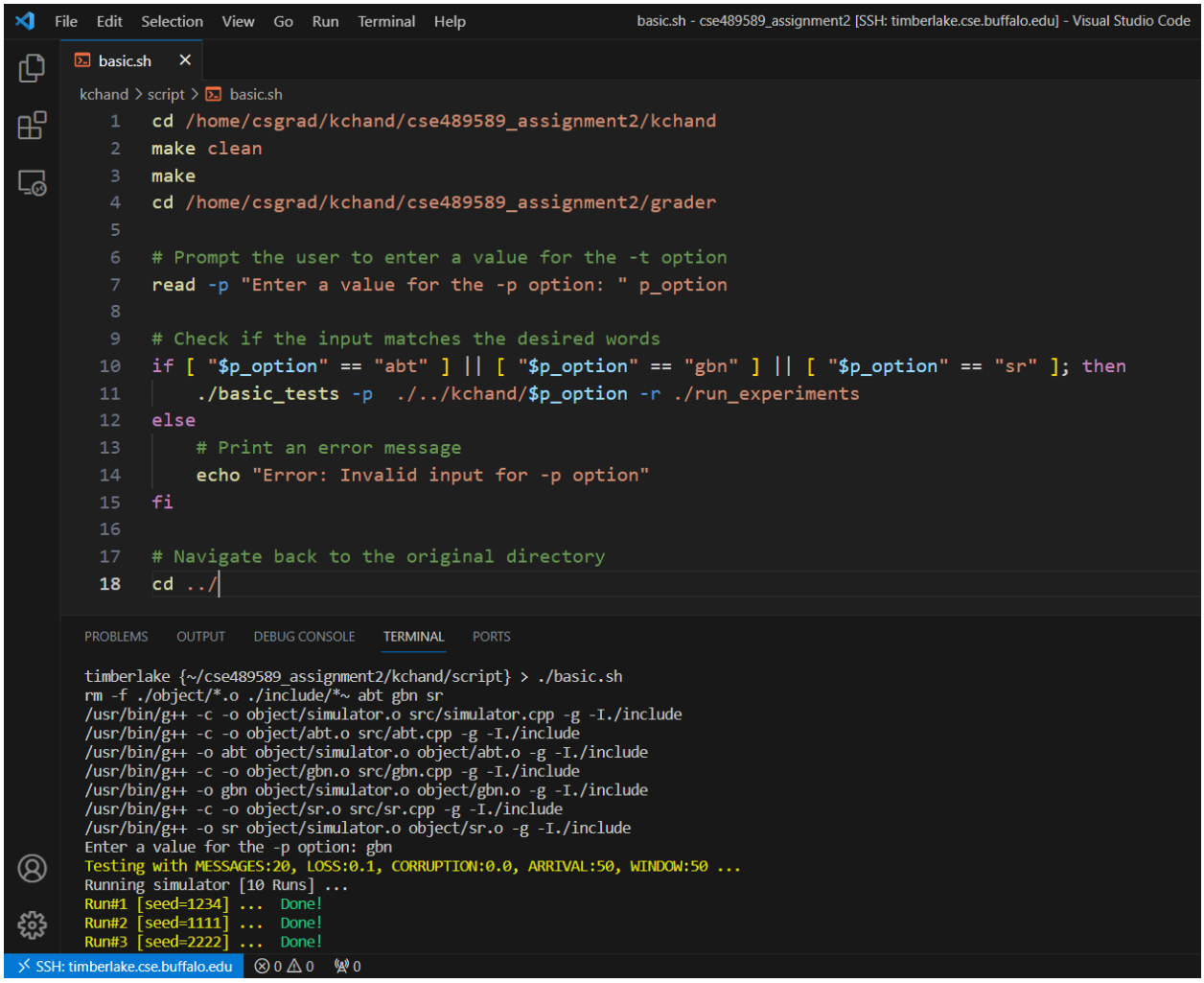
```
Enter a value for the -p option: abt
Testing with MESSAGES:20, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:1000, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:20, LOSS:0.4, CORRUPTION:0.0, ARRIVAL:1000, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:20, LOSS:0.8, CORRUPTION:0.0, ARRIVAL:1000, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:20, LOSS:0.0, CORRUPTION:0.1, ARRIVAL:1000, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```



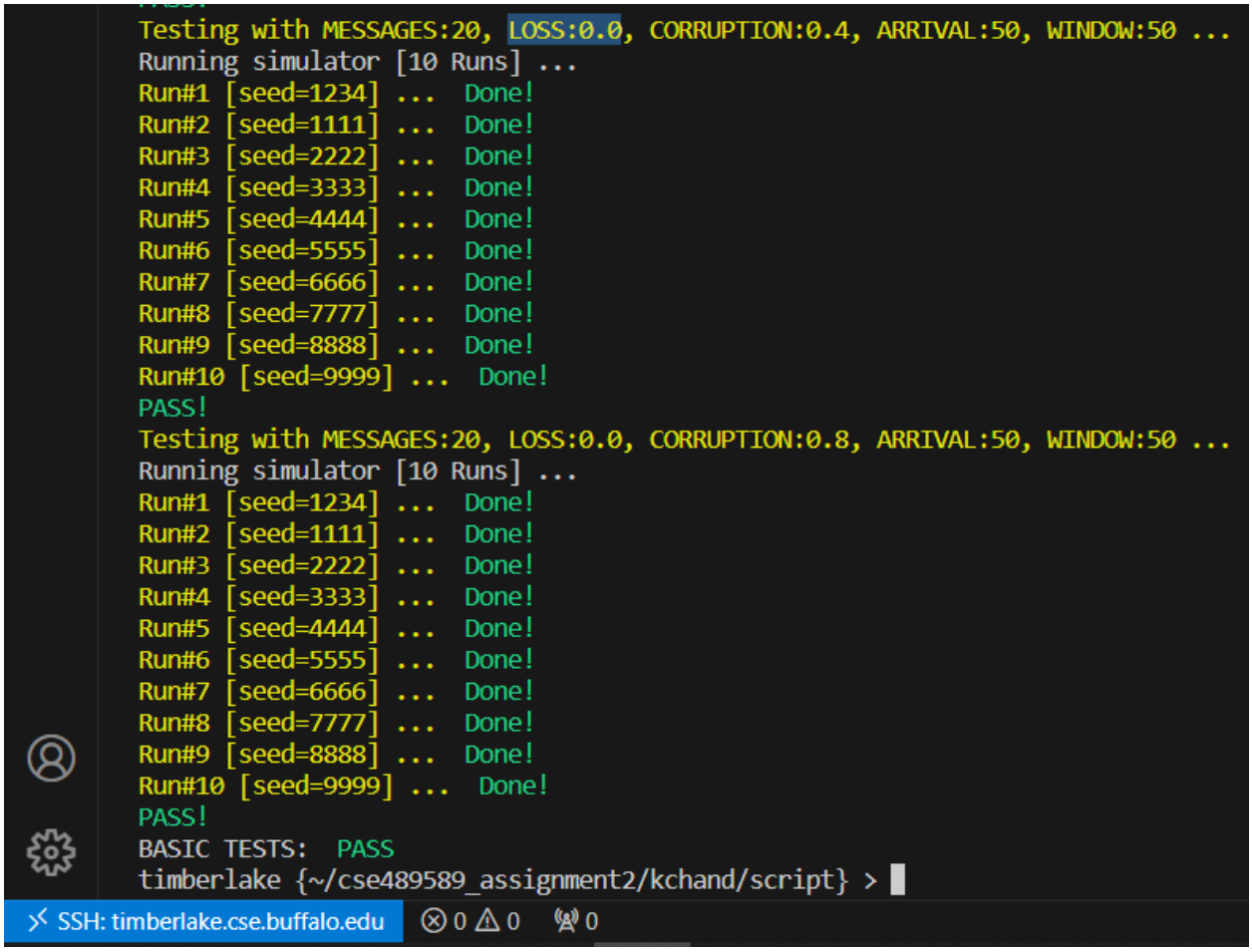
[12.0] GBN

(Put screenshots of the grader here…)


File Edit Selection View Go Run Terminal Help
basic.sh - cse489589_assignment2 [SSH: timberlake.cse.buffalo.edu] - Visual Studio Code
basic.sh
kchand > script > basic.sh
1 cd /home/csgrad/kchand/cse489589_assignment2/kchand
2 make clean
3 make
4 cd /home/csgrad/kchand/cse489589_assignment2/grader
5
6 # Prompt the user to enter a value for the -t option
7 read -p "Enter a value for the -p option: " p_option
8
9 # Check if the input matches the desired words
10 if ["$p_option" == "abt"] || ["$p_option" == "gbn"] || ["$p_option" == "sr"]; then
11 ./basic_tests -p ./../kchand/$p_option -r ./run_experiments
12 else
13 # Print an error message
14 echo "Error: Invalid input for -p option"
15 fi
16
17 # Navigate back to the original directory
18 cd ../
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
timberlake {~/cse489589_assignment2/kchand/script} > ./basic.sh
rm -f ./object/*.o ./include/*~ abt gbn sr
/usr/bin/g++ -c -o object/simulator.o src/simulator.cpp -g -I./include
/usr/bin/g++ -c -o object/abt.o src/abt.cpp -g -I./include
/usr/bin/g++ -o abt object/simulator.o object/abt.o -g -I./include
/usr/bin/g++ -c -o object/gbn.o src/gbn.cpp -g -I./include
/usr/bin/g++ -o gbn object/simulator.o object/gbn.o -g -I./include
/usr/bin/g++ -c -o object/sr.o src/sr.cpp -g -I./include
/usr/bin/g++ -o sr object/simulator.o object/sr.o -g -I./include
Enter a value for the -p option: gbn
Testing with MESSAGES:20, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ... Done!
Run#2 [seed=1111] ... Done!
Run#3 [seed=2222] ... Done!
SSH: timberlake.cse.buffalo.edu

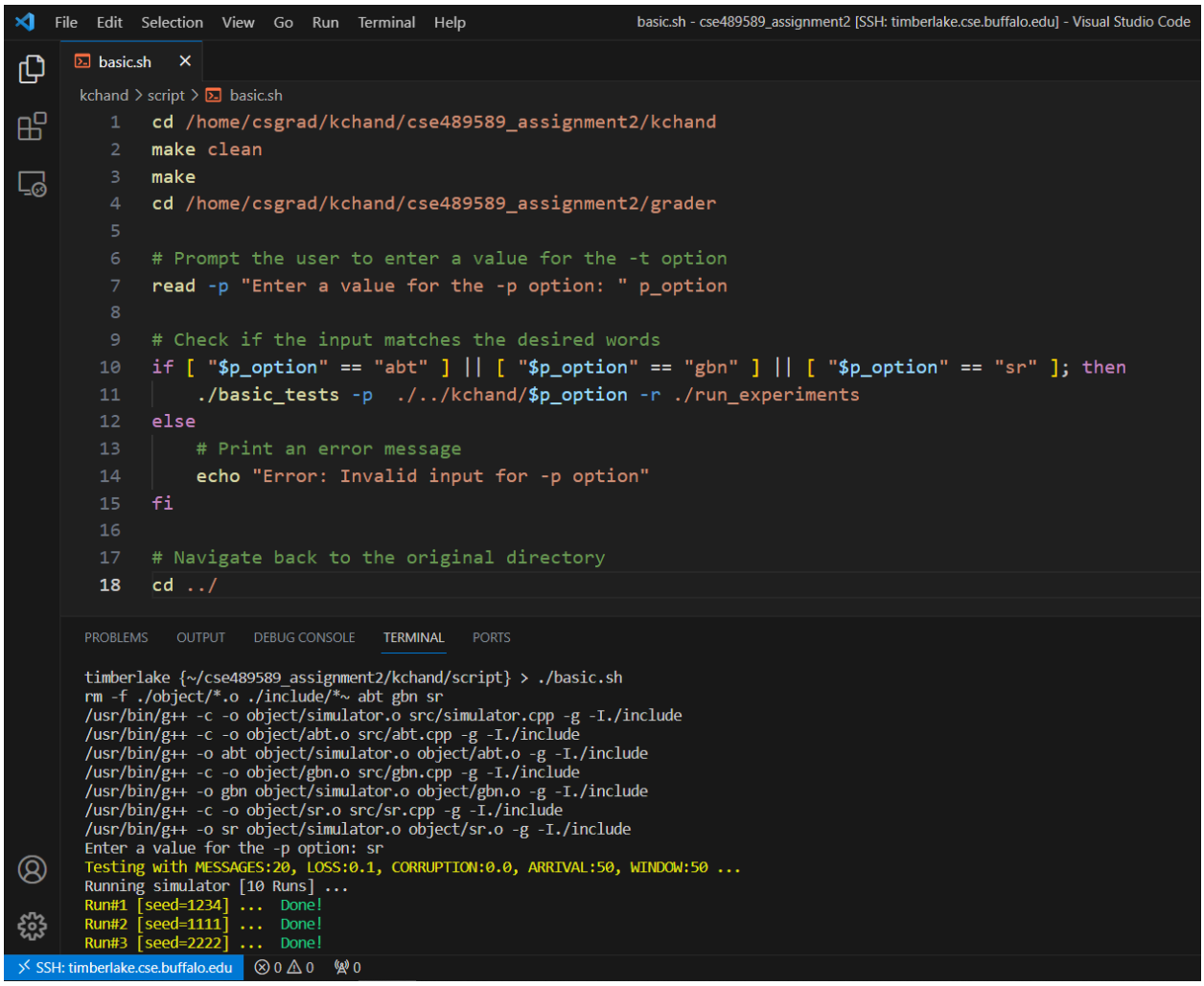
```
Enter a value for the -p option: gbn
Testing with MESSAGES:20, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:20, LOSS:0.4, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:20, LOSS:0.8, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:20, LOSS:0.0, CORRUPTION:0.1, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```



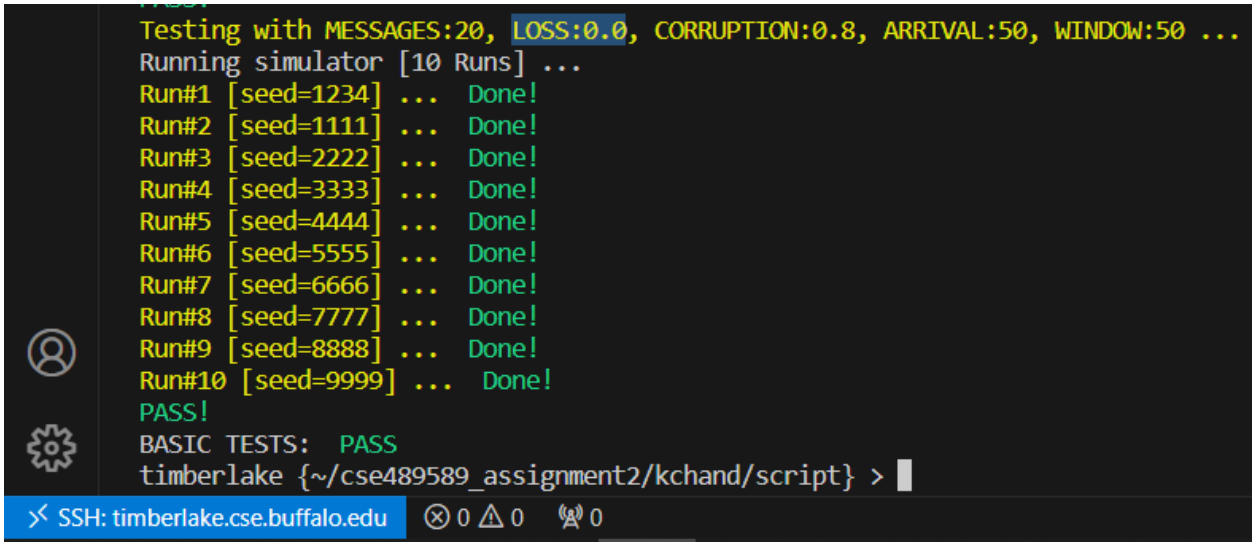
[18.0] SR

(Put screenshots of the grader here…)


File Edit Selection View Go Run Terminal Help
basic.sh - cse489589_assignment2 [SSH: timberlake.cse.buffalo.edu] - Visual Studio Code
basic.sh
kchand > script > basic.sh
1 cd /home/csgrad/kchand/cse489589_assignment2/kchand
2 make clean
3 make
4 cd /home/csgrad/kchand/cse489589_assignment2/grader
5
6 # Prompt the user to enter a value for the -t option
7 read -p "Enter a value for the -p option: " p_option
8
9 # Check if the input matches the desired words
10 if ["$p_option" == "abt"] || ["$p_option" == "gbn"] || ["$p_option" == "sr"]; then
11 ./basic_tests -p ./../kchand/$p_option -r ./run_experiments
12 else
13 # Print an error message
14 echo "Error: Invalid input for -p option"
15 fi
16
17 # Navigate back to the original directory
18 cd ../
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
timberlake {~/cse489589_assignment2/kchand/script} > ./basic.sh
rm -f ./object/*.o ./include/*~ abt gbn sr
/usr/bin/g++ -c -o object/simulator.o src/simulator.cpp -g -I./include
/usr/bin/g++ -c -o object/abt.o src/abt.cpp -g -I./include
/usr/bin/g++ -o abt object/simulator.o object/abt.o -g -I./include
/usr/bin/g++ -c -o object/gbn.o src/gbn.cpp -g -I./include
/usr/bin/g++ -o gbn object/simulator.o object/gbn.o -g -I./include
/usr/bin/g++ -c -o object/sr.o src/sr.cpp -g -I./include
/usr/bin/g++ -o sr object/simulator.o object/sr.o -g -I./include
Enter a value for the -p option: sr
Testing with MESSAGES:20, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ... Done!
Run#2 [seed=1111] ... Done!
Run#3 [seed=2222] ... Done!
SSH: timberlake.cse.buffalo.edu

```
Enter a value for the -p option: sr
Testing with MESSAGES:20, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:20, LOSS:0.4, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:20, LOSS:0.8, CORRUPTION:0.0, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:20, LOSS:0.0, CORRUPTION:0.1, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:20, LOSS:0.0, CORRUPTION:0.4, ARRIVAL:50, WINDOW:50 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

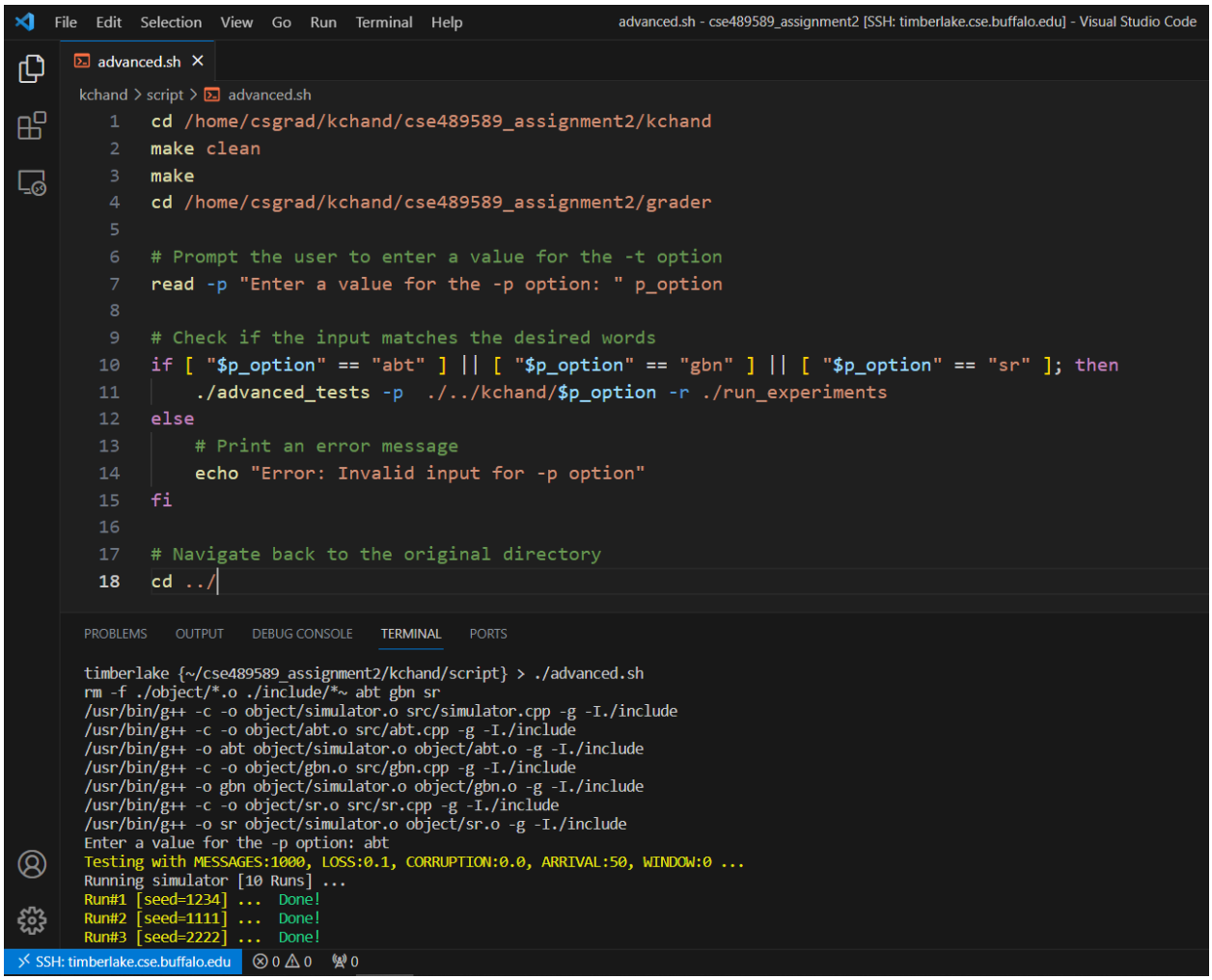


[No further grading for the protocol that fails a BASIC test.]

5 - ADVANCED Tests

[5.0] ABT

(Put screenshots of the grader here…)


File Edit Selection View Go Run Terminal Help
advanced.sh - cse489589_assignment2 [SSH: timberlake.cse.buffalo.edu] - Visual Studio Code
advanced.sh
kchand > script > advanced.sh
1 cd /home/csgrad/kchand/cse489589_assignment2/kchand
2 make clean
3 make
4 cd /home/csgrad/kchand/cse489589_assignment2/grader
5
6 # Prompt the user to enter a value for the -t option
7 read -p "Enter a value for the -p option: " p_option
8
9 # Check if the input matches the desired words
10 if ["$p_option" == "abt"] || ["$p_option" == "gbn"] || ["$p_option" == "sr"]; then
11 ./advanced_tests -p ./../kchand/$p_option -r ./run_experiments
12 else
13 # Print an error message
14 echo "Error: Invalid input for -p option"
15 fi
16
17 # Navigate back to the original directory
18 cd ../
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
timberlake {~/cse489589_assignment2/kchand/script} > ./advanced.sh
rm -f ./object/*.o ./include/*~ abt gbn sr
/usr/bin/g++ -c -o object/simulator.o src/simulator.cpp -g -I./include
/usr/bin/g++ -c -o object/abt.o src/abt.cpp -g -I./include
/usr/bin/g++ -o abt object/simulator.o object/abt.o -g -I./include
/usr/bin/g++ -c -o object/gbn.o src/gbn.cpp -g -I./include
/usr/bin/g++ -o gbn object/simulator.o object/gbn.o -g -I./include
/usr/bin/g++ -c -o object/sr.o src/sr.cpp -g -I./include
/usr/bin/g++ -o sr object/simulator.o object/sr.o -g -I./include
Enter a value for the -p option: abt
Testing with MESSAGES:1000, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ... Done!
Run#2 [seed=1111] ... Done!
Run#3 [seed=2222] ... Done!
SSH: timberlake.cse.buffalo.edu
0 0 0

```
Enter a value for the -p option: abt
Testing with MESSAGES:1000, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.2, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.4, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

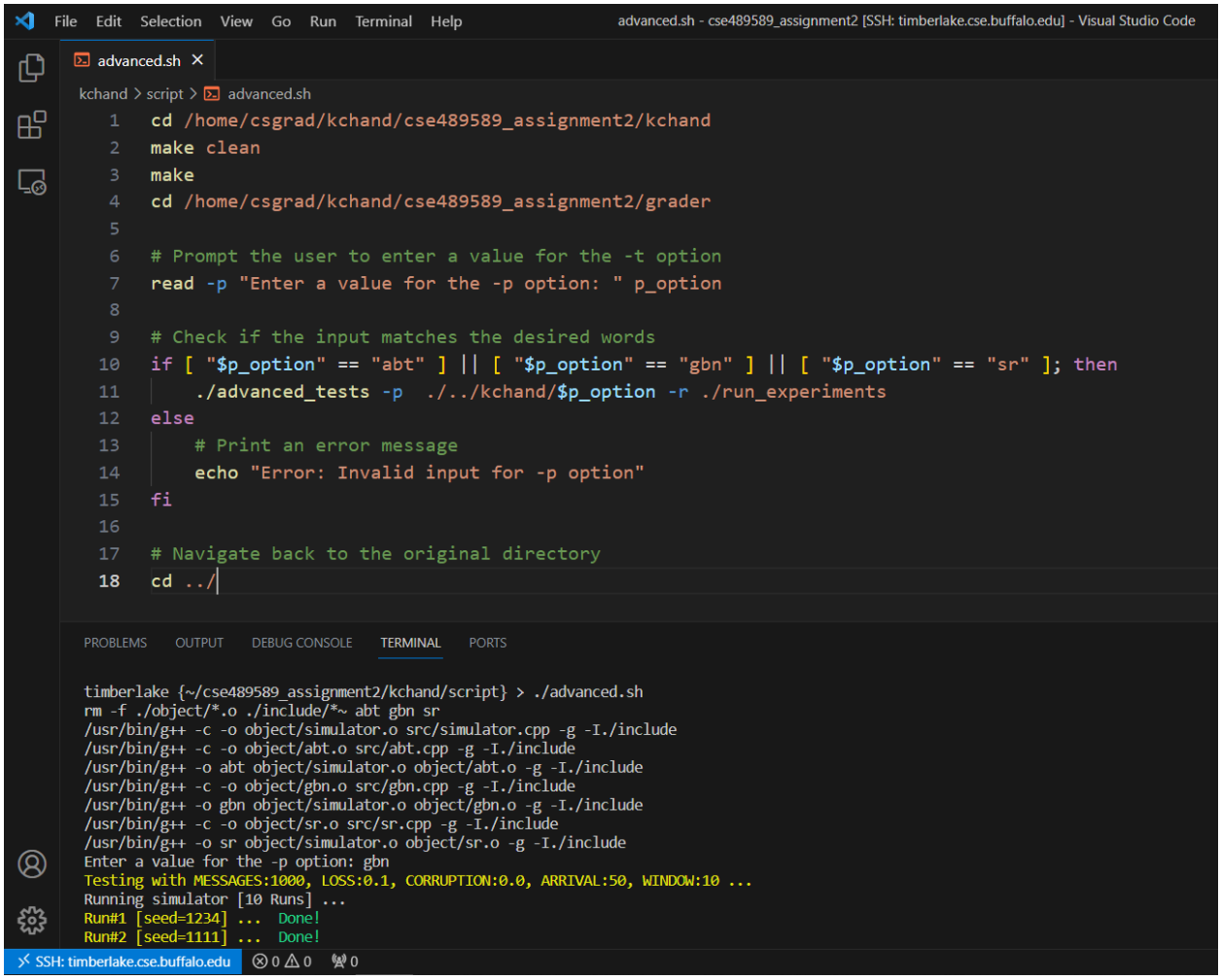
```
Testing with MESSAGES:1000, LOSS:0.6, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.8, CORRUPTION:0.0, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.1, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.2, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.4, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.6, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.8, ARRIVAL:50, WINDOW:0 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
ADVANCED TESTS:  PASS
timberlake {~/cse489589_assignment2/kchand/script} >
SSH: timberlake.cse.buffalo.edu  ⊗ 0 ⚠ 0  0
```

[10.0] GBN

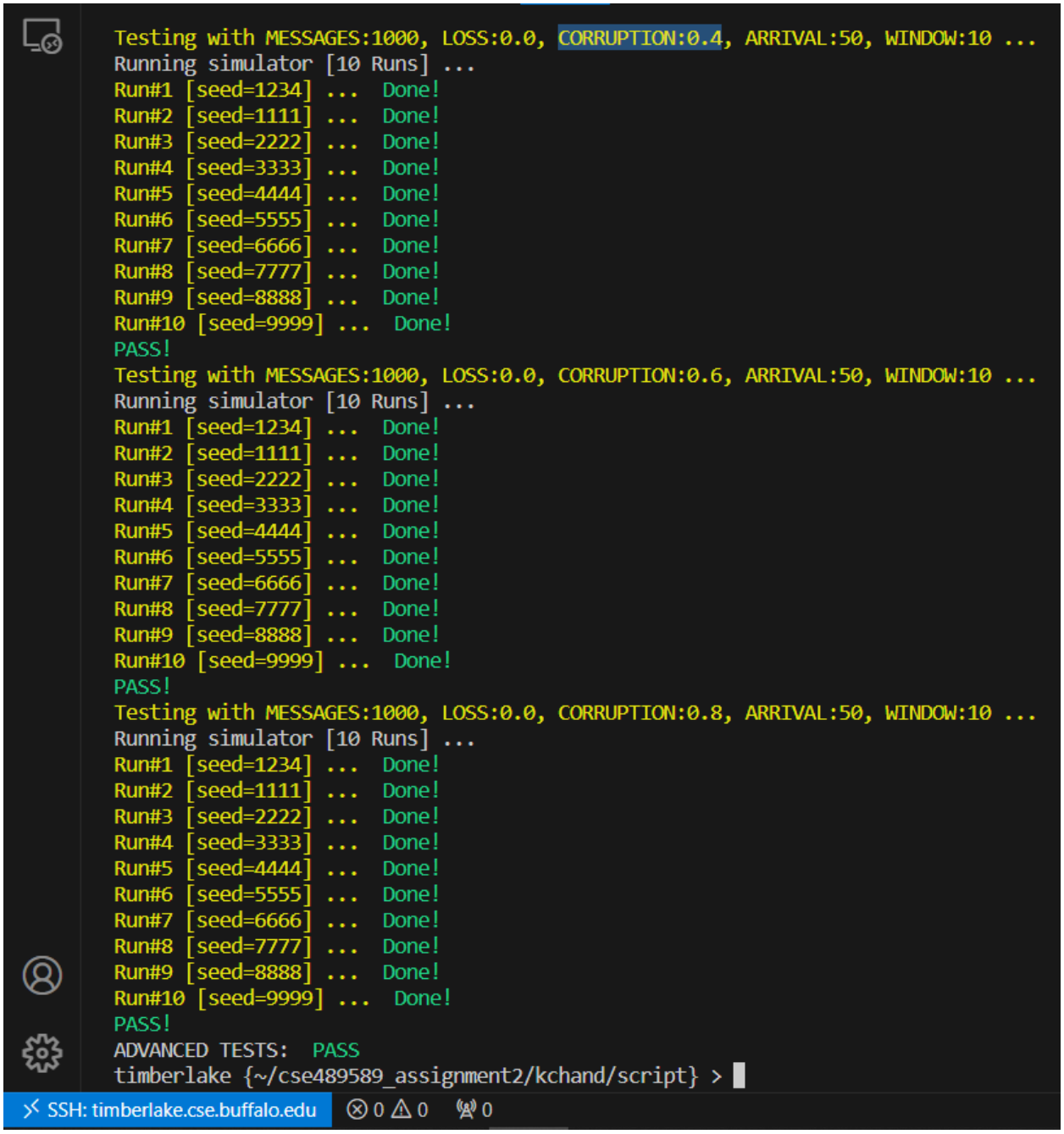
(Put screenshots of the grader here…)



```
Enter a value for the -p option: gbn
Testing with MESSAGES:1000, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.2, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

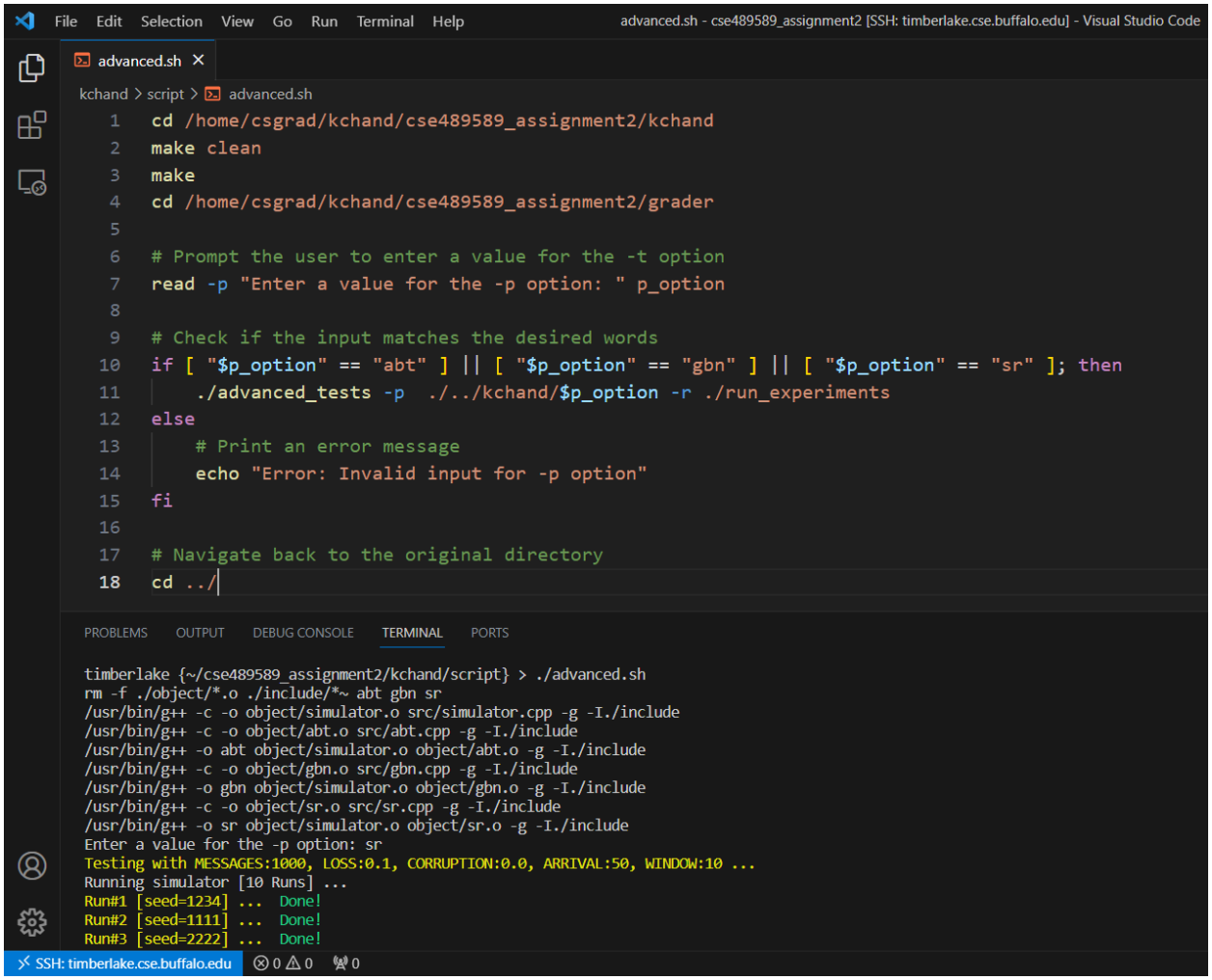
```
Testing with MESSAGES:1000, LOSS:0.4, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.6, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.8, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.1, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.2, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```



[20.0] SR

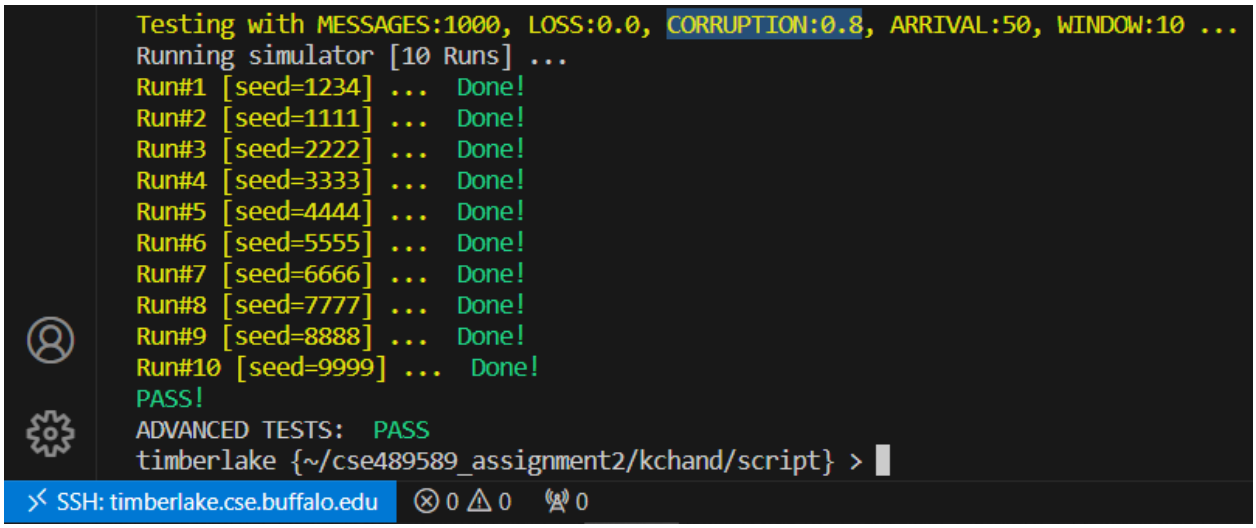
(Put screenshots of the grader here…)


File Edit Selection View Go Run Terminal Help
advanced.sh - cse489589_assignment2 [SSH: timberlake.cse.buffalo.edu] - Visual Studio Code
advanced.sh
kchand > script > advanced.sh
1 cd /home/csgrad/kchand/cse489589_assignment2/kchand
2 make clean
3 make
4 cd /home/csgrad/kchand/cse489589_assignment2/grader
5
6 # Prompt the user to enter a value for the -t option
7 read -p "Enter a value for the -p option: " p_option
8
9 # Check if the input matches the desired words
10 if ["$p_option" == "abt"] || ["$p_option" == "gbn"] || ["$p_option" == "sr"]; then
11 ./advanced_tests -p ./../kchand/$p_option -r ./run_experiments
12 else
13 # Print an error message
14 echo "Error: Invalid input for -p option"
15 fi
16
17 # Navigate back to the original directory
18 cd ../
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
timberlake {~/cse489589_assignment2/kchand/script} > ./advanced.sh
rm -f ./object/*.o ./include/*~ abt gbn sr
/usr/bin/g++ -c -o object/simulator.o src/simulator.cpp -g -I./include
/usr/bin/g++ -c -o object/abt.o src/abt.cpp -g -I./include
/usr/bin/g++ -o abt object/simulator.o object/abt.o -g -I./include
/usr/bin/g++ -c -o object/gbn.o src/gbn.cpp -g -I./include
/usr/bin/g++ -o gbn object/simulator.o object/gbn.o -g -I./include
/usr/bin/g++ -c -o object/sr.o src/sr.cpp -g -I./include
/usr/bin/g++ -o sr object/simulator.o object/sr.o -g -I./include
Enter a value for the -p option: sr
Testing with MESSAGES:1000, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ... Done!
Run#2 [seed=1111] ... Done!
Run#3 [seed=2222] ... Done!
SSH: timberlake.cse.buffalo.edu
0 0 0

```
Enter a value for the -p option: sr
Testing with MESSAGES:1000, LOSS:0.1, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.2, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.4, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.6, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.8, CORRUPTION:0.0, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.1, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```

```
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.2, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.4, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.6, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ...  Done!
Run#2 [seed=1111] ...  Done!
Run#3 [seed=2222] ...  Done!
Run#4 [seed=3333] ...  Done!
Run#5 [seed=4444] ...  Done!
Run#6 [seed=5555] ...  Done!
Run#7 [seed=6666] ...  Done!
Run#8 [seed=7777] ...  Done!
Run#9 [seed=8888] ...  Done!
Run#10 [seed=9999] ...  Done!
PASS!
```


Testing with MESSAGES:1000, LOSS:0.0, CORRUPTION:0.8, ARRIVAL:50, WINDOW:10 ...
Running simulator [10 Runs] ...
Run#1 [seed=1234] ... Done!
Run#2 [seed=1111] ... Done!
Run#3 [seed=2222] ... Done!
Run#4 [seed=3333] ... Done!
Run#5 [seed=4444] ... Done!
Run#6 [seed=5555] ... Done!
Run#7 [seed=6666] ... Done!
Run#8 [seed=7777] ... Done!
Run#9 [seed=8888] ... Done!
Run#10 [seed=9999] ... Done!
PASS!
ADVANCED TESTS: PASS
timberlake {~/cse489589_assignment2/kchand/script} >
SSH: timberlake.cse.buffalo.edu
0 0
0

6 - ANALYSIS & REPORT [15.0]

In this report we are presenting experiments related to Reliable Transfer Protocols such as Alternating Bit Protocol (ABT), Go-Back-N (GBN) and Selective Repeat (SR).

ABT is a simple Reliable Transfer Protocol that uses a sequence number to alternate between sending packets of data and acknowledgments. Each packet is sent with a unique sequence number, and the receiver sends an acknowledgment with the same sequence number to indicate successful receipt. If the sender doesn't receive an acknowledgment within a certain time period, it retransmits the packet.

GBN is a sliding window protocol that allows the sender to transmit multiple packets without waiting for an acknowledgment for each one. The receiver sends cumulative acknowledgments, acknowledging all packets up to a certain sequence number. If the sender doesn't receive an acknowledgment for a particular packet, it retransmits all packets starting from that sequence number.

SR is also a sliding window protocol, but it allows the receiver to individually acknowledge each packet. The sender maintains a buffer of packets that have been sent but not yet acknowledged. If a packet is not acknowledged within a certain time period, the sender retransmits that packet only. This reduces the amount of unnecessary retransmissions, improving efficiency.

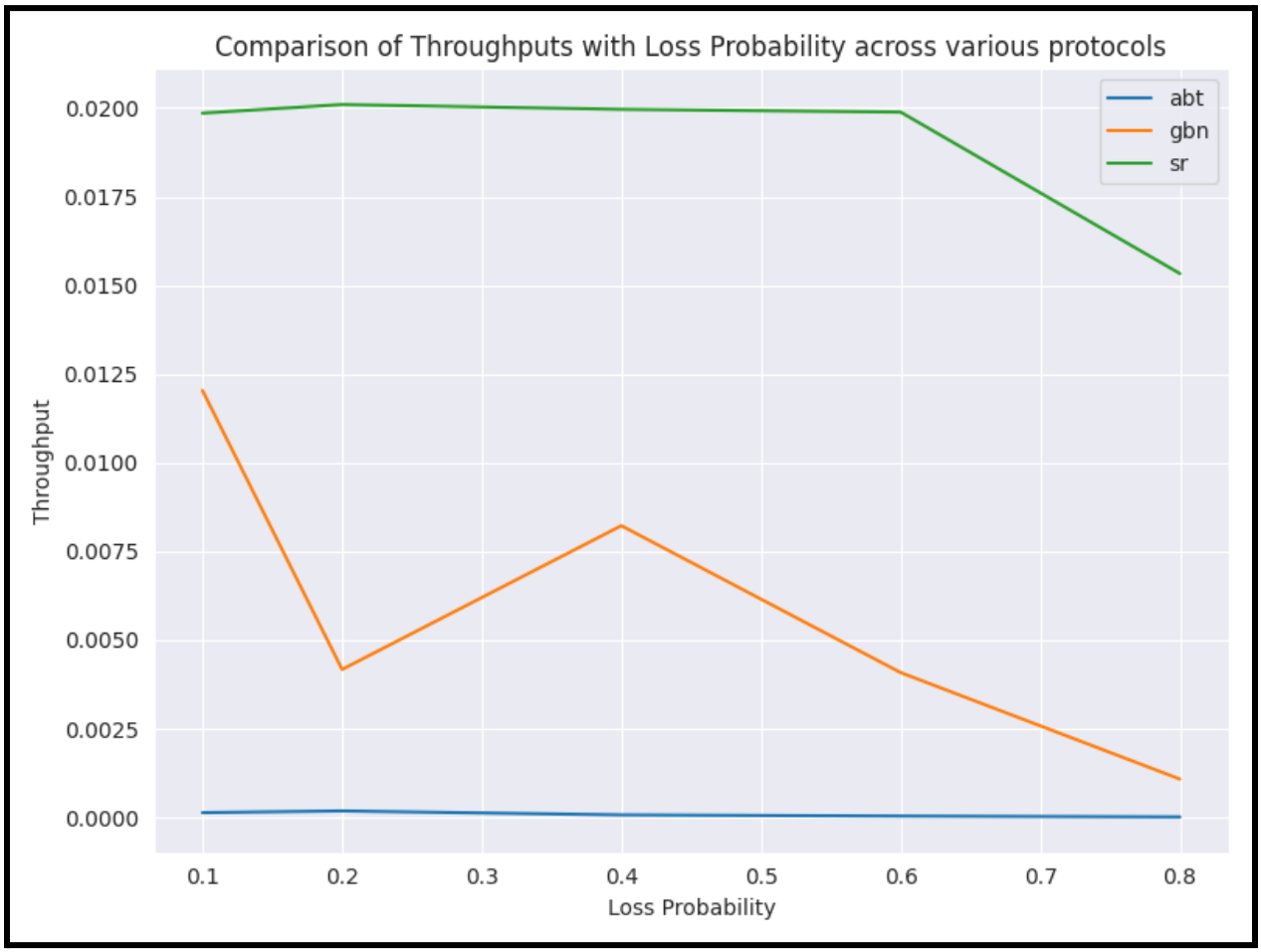
Timer Description in SR protocol
SR uses ACK packets to confirm the receipt of packets, and if a packet is missing, the receiver waits for a retransmission of the missing packet. To implement multiple software timers in SR using a single hardware timer, we used a timer wheel logic. We used an array of timers, where each timer is associated with a packet. When a packet is sent, its corresponding timer is started, and the timer value is set to the retransmission timeout. When the timer expires, the corresponding packet is retransmitted, and its timer is restarted.
We used a separate class definition for store the timer information for each packet:

```
class pkt_Class {
public:
   pkt packet;
   float time;
};
```

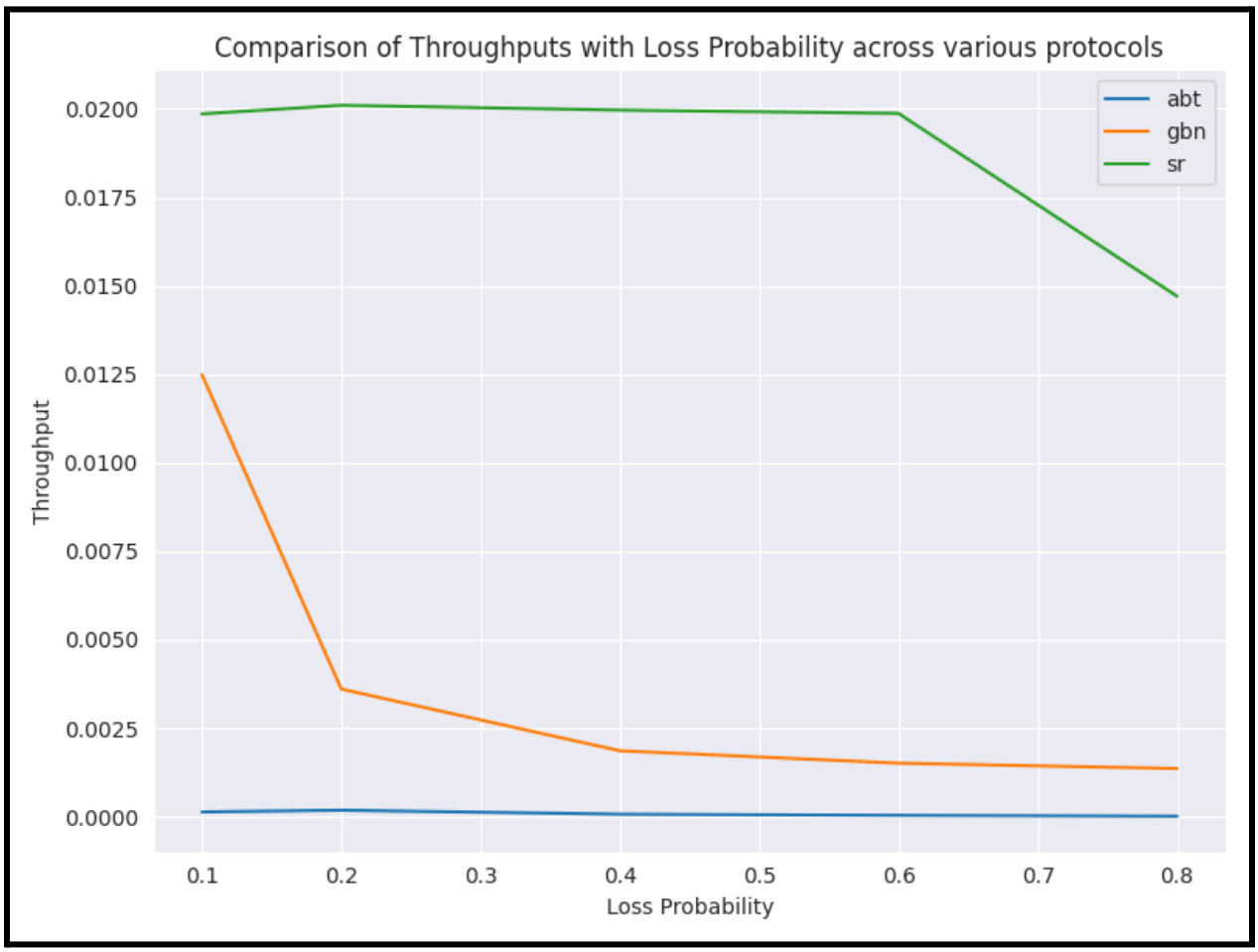
This allowed us to implement multiple software timers using a single hardware timer.

Experiment 1:

Window size: 10; X-axis: Loss probability; Y-axis: Throughput (ABT, GBN and SR) in one graph/plot.



Window size: 50; X-axis: Loss probability; Y-axis: Throughput (ABT, GBN and SR) in one graph/plot.



Observations and Analysis for Experiment 1:

When we increase the loss probability and keep the window size constant in the Selective Repeat Protocol, the throughput should decrease. This is because as the loss probability increases, there is a higher likelihood that packets will be lost or damaged during transmission. When this happens, the receiver will request retransmission of the lost packets, which will increase the total number of transmissions and reduce the overall throughput.

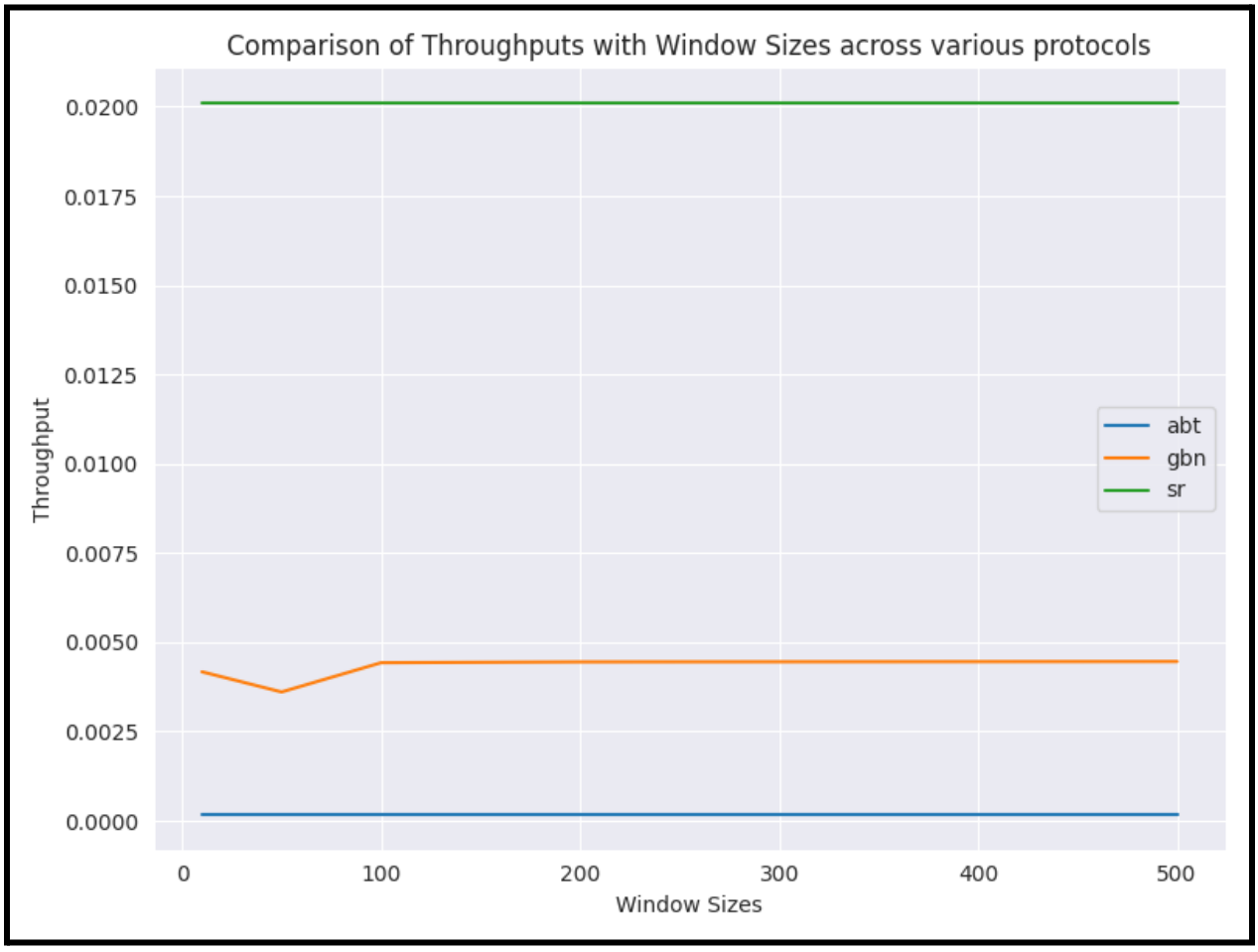
When we increase the loss probability and keep the window size constant in the Go-Back-N protocol, the throughput should decrease, just like in the Selective Repeat Protocol. The reason for this is similar: as the loss probability increases, the likelihood of packet loss or damage during transmission also increases. This means that the sender will have to retransmit packets more frequently, which will reduce the overall throughput.

When we increase the loss probability and keep the window size constant in the ABT protocol, the throughput should decrease, similar to the Selective Repeat and Go-Back-N protocols. The reason for this is also the same: as the loss probability increases, there is a higher chance that packets will be lost or damaged during transmission, and the sender will have to retransmit them, which reduces the overall throughput.

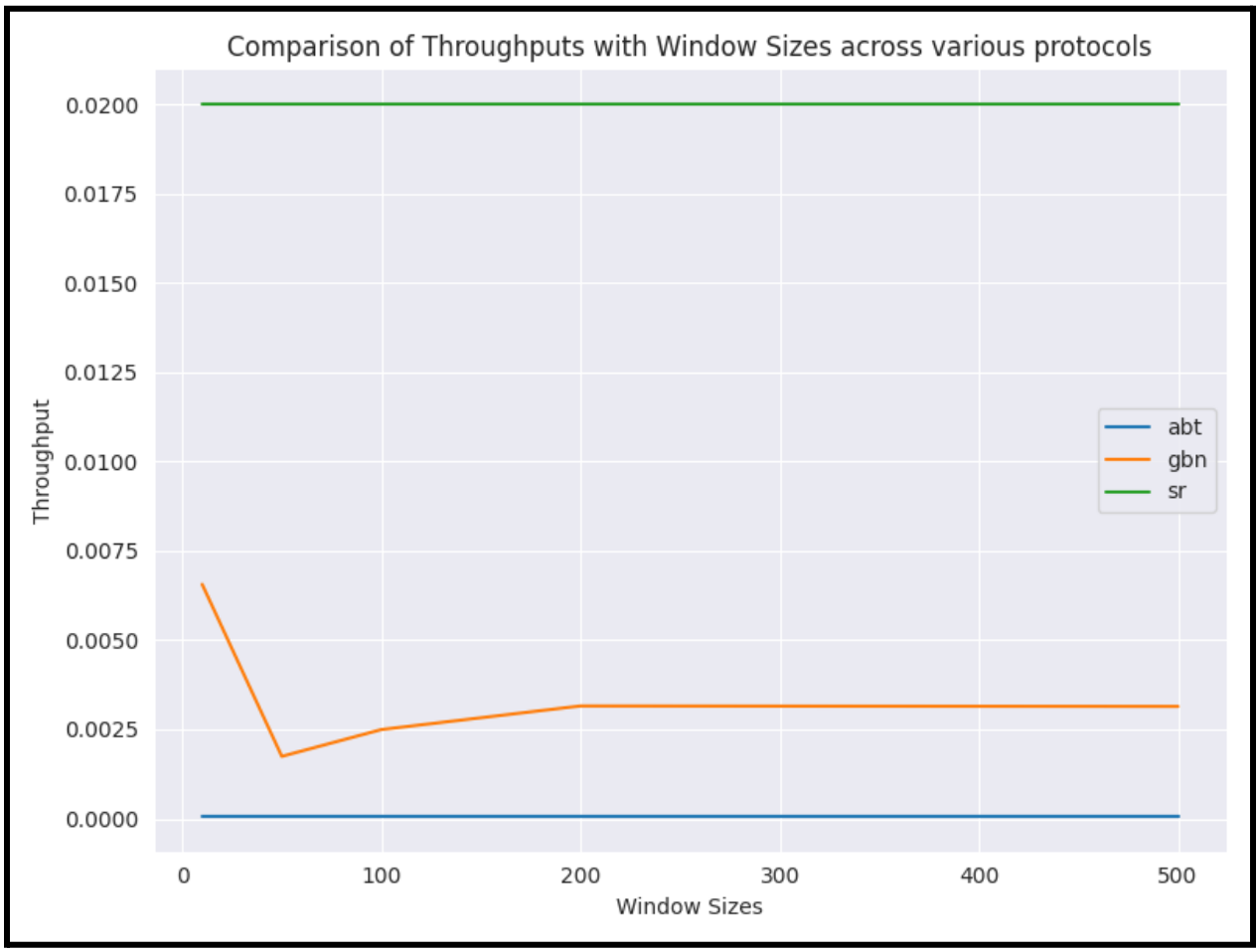
SR outperforms rest of the two protocols from start to end in our observation.

Experiment 2:

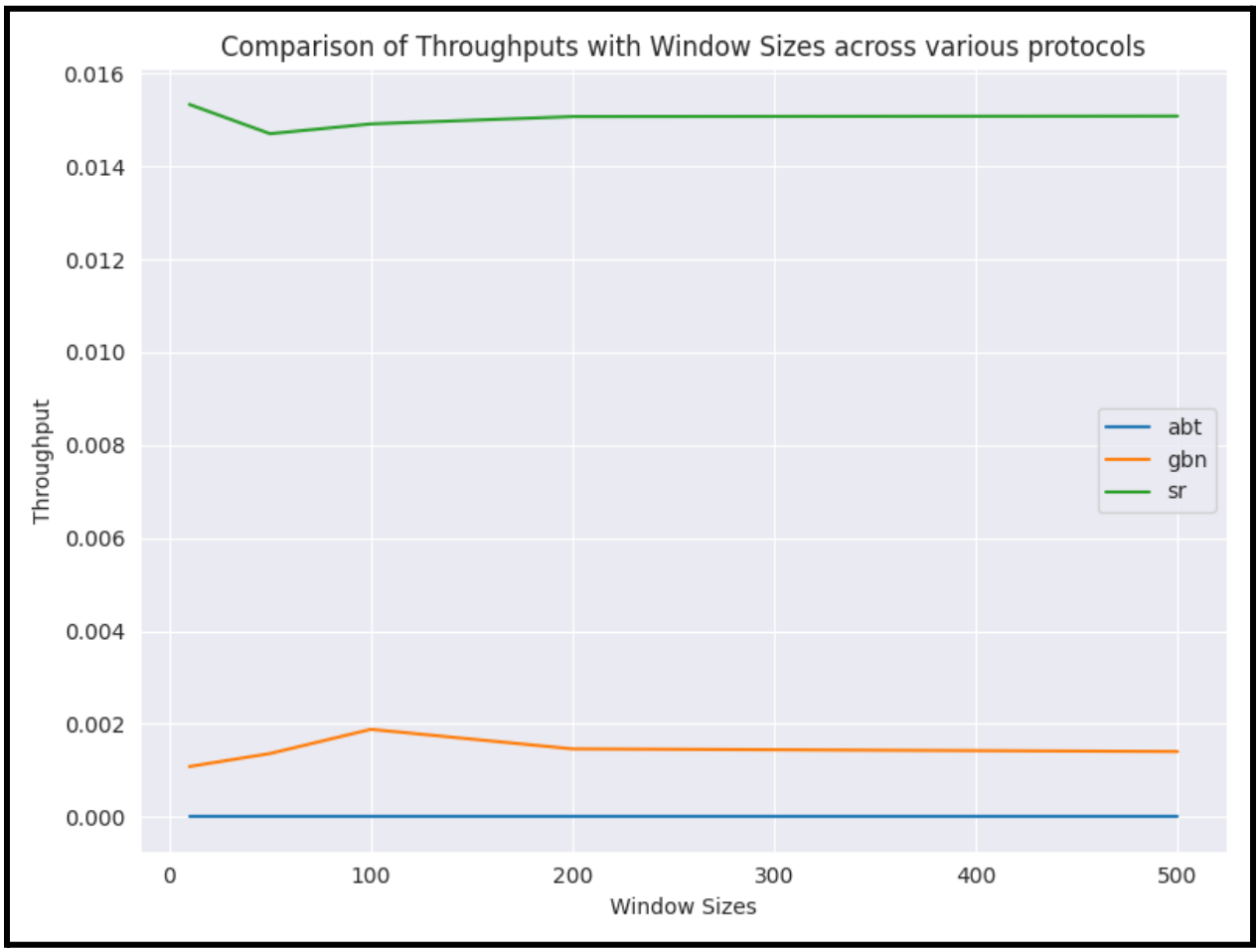
Loss probability: 0.2; X-axis: Window size; Y-axis: Throughput (ABT, GBN and SR) in one graph/plot.



Loss probability: 0.5; X-axis: Window size; Y-axis: Throughput (ABT, GBN and SR) in one graph/plot.



Loss probability: 0.8; X-axis: Window size; Y-axis: Throughput (ABT, GBN and SR) in one graph/plot.



Observations and Analysis for Experiment 2:

Ideally, as window size increases, the throughput of all three protocols generally increases until a certain point, after which further increase in window size does not result in significant improvement in throughput. This is due to the limited available bandwidth and increasing delay caused by larger window sizes.

As per our observation we didn't see much spikes and drops in the throughput as throughput is remaining constant for all the three protocols, maybe we can observe more difference for much larger window sizes.

ABT is the most affected by changes in window size, as it is a stop-and-wait protocol and can only send one packet at a time. Hence, its throughput is limited by its window size.
GBN performs better than ABT as it can send multiple packets before waiting for an acknowledgment. However, GBN's throughput is still limited by its window size, and increasing

window size beyond a certain point does not result in significant improvement.
In our observation we saw a sudden drop in throughput of GBN when window size increased.

SR has the best performance among the three protocols as it can handle out-of-order packets and can selectively retransmit only lost packets. Hence, even with a small window size, SR can achieve high throughput.
This observation of SR performing best was as expected

Conclusion

We can clearly see throughput of SR > Go Back N > ABT for both the experiments.
This happens due to the window size involved in the protocol of the receiver and the way each of the protocols deal with the acknowledgement and missed packets retransfer.

References:

Computer Networking A Top-Down Approach, Global Edition, 8th Edition (Kurose, James, Ross, Keith).pdf | Powered by Box