

**Gebze Technical University
Computer Engineering**

**CSE 344
System Programming**

MIDTERM PROJECT REPORT

**SEDA KANIK
151044068**

1. General Project Definition

In this project we are responsible to create a program that will simulate the student mess hall of a university. There are 3 actors, supplier, cook and students. Each of them will be a separate process. And there are also 3 places, kitchen, counter and table.

1.1. Supplier

The task of the supplier is to deliver already cooked plates of food to the kitchen. It will read a file which is filePath to deliver all the plates (so LxM times P, C, D). The filePath includes 3xLxM plates and there is no fault among P, C, D.

1.2. Cook

The task of the cook is to place plates which supplier is already delivered to kitchen on the counter, if there is some empty place in.

1.3. Student

The task of the student is to get all the foods and sitting a table and eating food, L times.

2. Implementation

The program checks the given numbers (cook, student, table, counter size, round) and gives the information about if there is something wrong. Forked among all the actors (n+m+1 times) and kept its PID in array children. Processes continue to run and checked if the process is supplier or cook or student. Then semaphore is created with all the necessary informations. After these things, all the work will done infinite loop.

2.1. Supplier

Supplier reads the file which name is filePath. Then it delivers the plates one by one while kitchen has some empty room. Then decrease item of supplier and increase item of kitchen. Print screen about work during all the things are happen. If read plate is P, decrease soup of the supplier and items of the supplier, increase soup of the kitchen and items of the kitchen. If read plate is C, decrease main course of the supplier and items of the supplier, increase main course of the kitchen and items of the kitchen. Then say GOODBYE when all the delivery is done.

Cook gets a plate from the kitchen (P, C, D) and place it on the counter if any there is any empty room. If kitchen is empty, waits for some plates. If counter is full, waits some empty room for place plates. If kitchen has some plates and cook gets a plate from kitchen and place it counter. This work will execute all the plates is placed in counter. Print screen about work during all the things are happen. Then say GOODBYE.

Student gets 3 plates from the counter, goes to look table and if the student finds a table (random generated) sits on it and the table is empty. Student eats its foods and leaves from the table and the table is free now. All the students repeats this L times. If student goes counter before the supplier or cook work, the student will wait on counter.

- Sample Results

There is not an error.

```
> program.c x filePath x
```

```
1 PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
2 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
3 DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
4
```

```
sed@sed: ~/Desktop/midterm
```

```
Student 0 left table 1 to eat again (round 5) - empty tables: 2
Student 9 is done eating L=5 times - going home - GOODBYE!!!
Student 0 got food and is going to get a table (round 1) - # of empty tables: 2
Student 0 sat at table 2 to eat (round 1) - empty tables: 1
Student 0 left table 2 to eat again (round 1) - empty tables: 2
Student 0 is going to the counter (round 2) - # of students at counter: 1 and counter items: P:2, C:2, D:2 = 6
Student 0 got food and is going to get a table (round 2) - # of empty tables: 2
Student 0 sat at table 1 to eat (round 2) - empty tables: 1
Student 0 left table 1 to eat again (round 2) - empty tables: 2
Student 0 is going to the counter (round 3) - # of students at counter: 1 and counter items: P:1, C:1, D:1 = 3
Student 0 got food and is going to get a table (round 3) - # of empty tables: 2
Student 0 sat at table 1 to eat (round 3) - empty tables: 1
Student 0 left table 1 to eat again (round 3) - empty tables: 2
Student 0 is going to the counter (round 4) - # of students at counter: 1 and counter items: P:0, C:0, D:0 = 0
Cook 2 is going to the counter to deliver dessert - counter items P:0, C:0, D:0 = 0
Cook 3 placed dessert on the counter - counter items: P:0, C:0, D:1 = 1
Cook 1 is going to the counter to deliver soup - counter items P:0, C:0, D:1 = 1
Cook 1 placed soup on the counter - counter items: P:1, C:0, D:1 = 2
Cook 1 is going to the counter to deliver main course - counter items P:1, C:0, D:1 = 2
Cook 2 placed main course on the counter - counter items: P:1, C:1, D:1 = 3
Cook 1 is going to the counter to deliver dessert - counter items P:1, C:1, D:1 = 3
Cook 2 placed dessert on the counter - counter items: P:1, C:1, D:2 = 4
Cook 1 is going to the kitchen to wait for/get a plate - kitchen items P:1, C:1, D:0 = 2
Cook 1 is going to the counter to deliver soup - counter items P:1, C:1, D:2 = 4
Cook 1 placed soup on the counter - counter items: P:2, C:1, D:2 = 5
Cook 1 is going to the counter to deliver main course - counter items P:2, C:1, D:2 = 5
Cook 2 placed main course on the counter - counter items: P:2, C:2, D:2 = 6
Cook 2 finished serving - items at kitchen: 0 - going home - GOODBYE!!!
Cook 1 finished serving - items at kitchen: 0 - going home - GOODBYE!!!
Cook 3 finished serving - items at kitchen: 0 - going home - GOODBYE!!!
Student 0 got food and is going to get a table (round 4) - # of empty tables: 2
Student 0 sat at table 2 to eat (round 4) - empty tables: 1
Student 0 left table 2 to eat again (round 4) - empty tables: 2
Student 0 is going to the counter (round 5) - # of students at counter: 1 and counter items: P:1, C:1, D:1 = 3
Student 0 got food and is going to get a table (round 5) - # of empty tables: 2
Student 0 sat at table 1 to eat (round 5) - empty tables: 1
Student 0 left table 1 to eat again (round 5) - empty tables: 2
Student 0 is done eating L=5 times - going home - GOODBYE!!!
```

```
sed@sed:~/Desktop/midterm$
```

This example includes 12 P, 12 C, 12 D and all the students makes 3 round, so it can not fit on screen.

There is not an error.

```
program.c x filePath x
1 P P P P P P P P P P
2 C C C C C C C C C C
3 D D D D D D D D D D
4
sed@sed: ~/Desktop/midterm
sed@sed:~/Desktop/midterm$ ./program -N 3 -M 4 -T 2 -S 4 -L 3 -F filePath
The supplier is going to the kitchen to deliver soup: kitchen items: P:0, C:0, D:0 = 0
The supplier delivered soup - after delivery: kitchen items: P:1, C:0, D:0 = 1
The supplier is going to the kitchen to deliver soup: kitchen items: P:1, C:0, D:0 = 1
The supplier delivered soup - after delivery: kitchen items: P:2, C:0, D:0 = 2
The supplier is going to the kitchen to deliver soup: kitchen items: P:2, C:0, D:0 = 2
The supplier delivered soup - after delivery: kitchen items: P:3, C:0, D:0 = 3
The supplier is going to the kitchen to deliver soup: kitchen items: P:3, C:0, D:0 = 3
The supplier delivered soup - after delivery: kitchen items: P:4, C:0, D:0 = 4
The supplier is going to the kitchen to deliver soup: kitchen items: P:4, C:0, D:0 = 4
The supplier delivered soup - after delivery: kitchen items: P:5, C:0, D:0 = 5
The supplier is going to the kitchen to deliver soup: kitchen items: P:5, C:0, D:0 = 5
The supplier delivered soup - after delivery: kitchen items: P:6, C:0, D:0 = 6
The supplier is going to the kitchen to deliver soup: kitchen items: P:6, C:0, D:0 = 6
The supplier delivered soup - after delivery: kitchen items: P:7, C:0, D:0 = 7
The supplier is going to the kitchen to deliver soup: kitchen items: P:7, C:0, D:0 = 7
The supplier delivered soup - after delivery: kitchen items: P:8, C:0, D:0 = 8
The supplier is going to the kitchen to deliver soup: kitchen items: P:8, C:0, D:0 = 8
The supplier delivered soup - after delivery: kitchen items: P:9, C:0, D:0 = 9
```

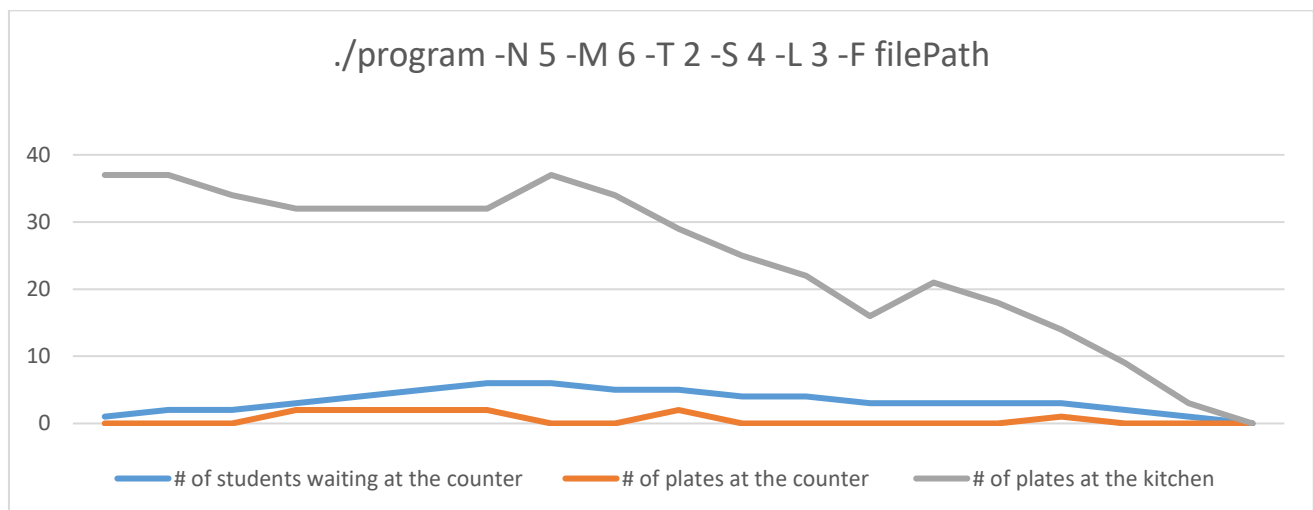
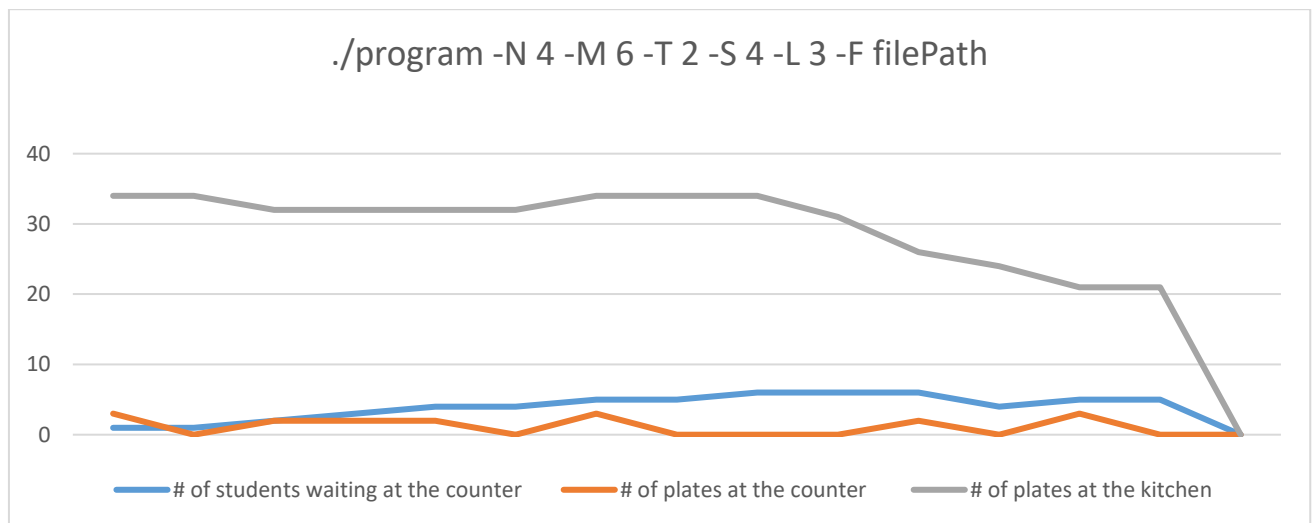
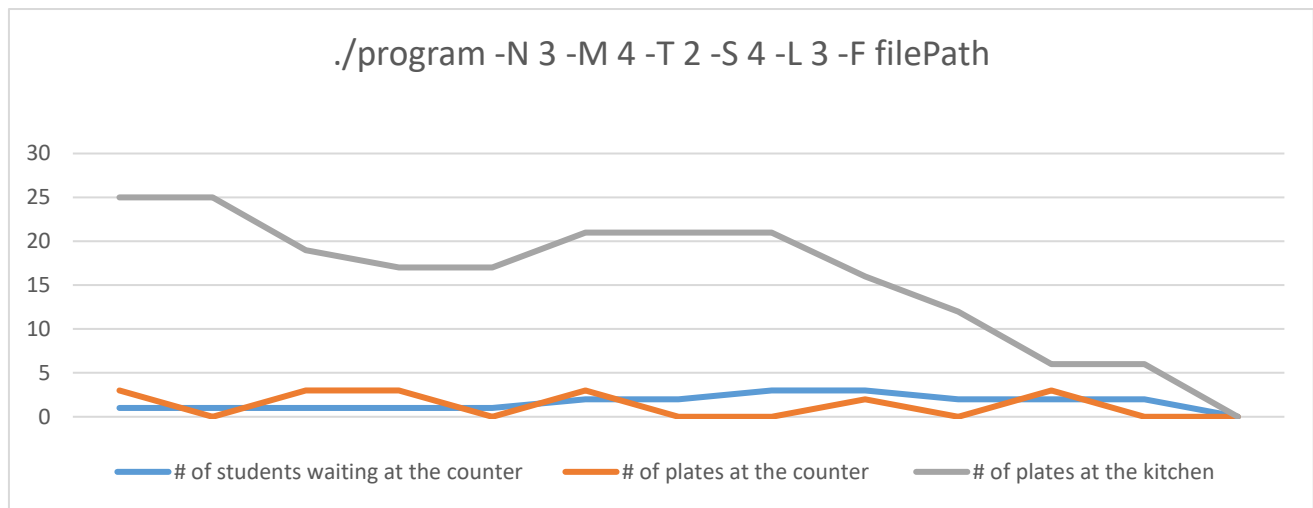
```
program.c x filePath x
1 P P P P P P P P P P
2 C C C C C C C C C C
3 D D D D D D D D D D
4
sed@sed: ~/Desktop/midterm
Cook 2 is going to the counter to deliver soup - counter items P:0, C:0, D:0 = 0
Cook 2 placed soup on the counter - counter items: P:1, C:0, D:0 = 1
Cook 2 is going to the counter to deliver main course - counter items P:1, C:0, D:0 = 1
Cook 3 placed main course on the counter - counter items: P:1, C:1, D:0 = 2
Cook 2 is going to the counter to deliver dessert - counter items P:1, C:1, D:0 = 2
Cook 3 placed dessert on the counter - counter items: P:1, C:1, D:1 = 3
Cook 2 is going to the kitchen to wait for/get a plate - kitchen items P:1, C:1, D:1 = 3
Student 1 got food and is going to get a table (round 3) - # of empty tables: 2
Student 1 sat at table 2 to eat (round 3) - empty tables: 1
Student 1 left table 2 to eat again (round 3) - empty tables: 2
Student 1 is done eating L=3 times - going home - GOODBYE!!!
Cook 0 is going to the counter to deliver soup - counter items P:0, C:0, D:0 = 0
Cook 0 placed soup on the counter - counter items: P:1, C:0, D:0 = 1
Cook 0 is going to the counter to deliver main course - counter items P:1, C:0, D:0 = 1
Cook 1 placed main course on the counter - counter items: P:1, C:1, D:0 = 2
Cook 0 is going to the counter to deliver dessert - counter items P:1, C:1, D:0 = 2
Cook 1 placed dessert on the counter - counter items: P:1, C:1, D:1 = 3
Cook 1 finished serving - items at kitchen: 0 - going home - GOODBYE!!!
Cook 2 finished serving - items at kitchen: 0 - going home - GOODBYE!!!
Student 2 got food and is going to get a table (round 3) - # of empty tables: 2
Student 2 sat at table 2 to eat (round 3) - empty tables: 1
Student 2 left table 2 to eat again (round 3) - empty tables: 2
Student 2 is done eating L=3 times - going home - GOODBYE!!!
Cook 3 finished serving - items at kitchen: 0 - going home - GOODBYE!!!
sed@sed:~/Desktop/midterm$
```

In some error cases.

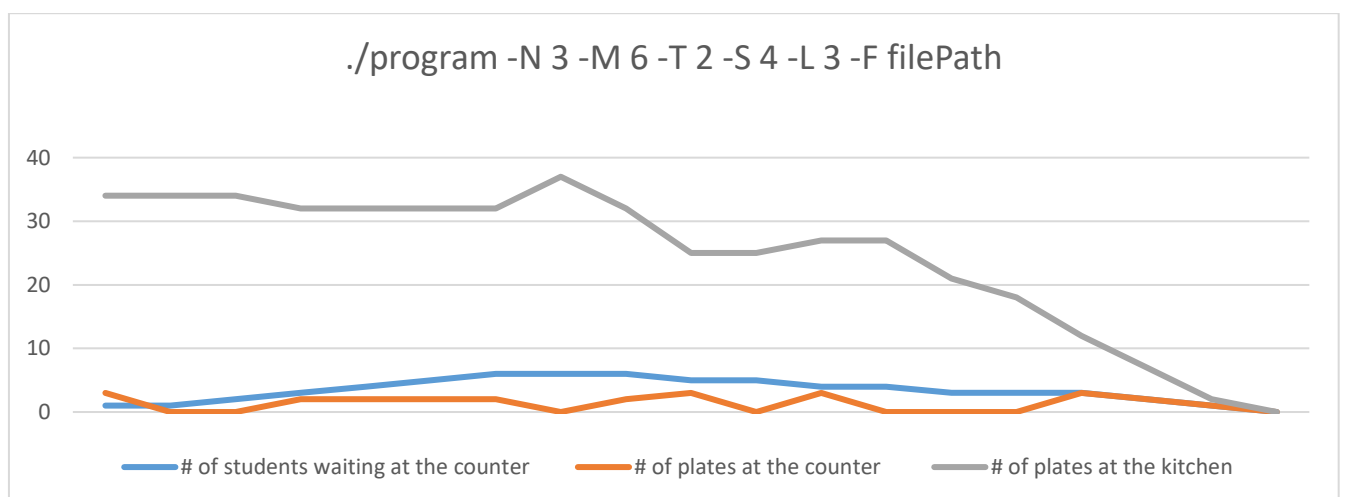
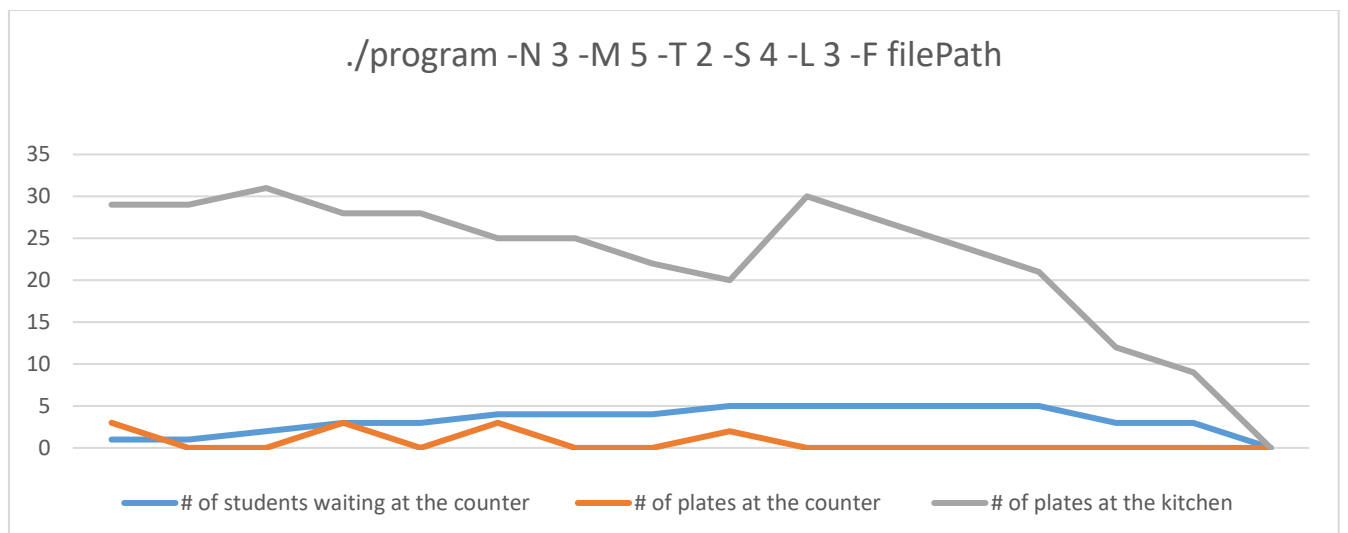
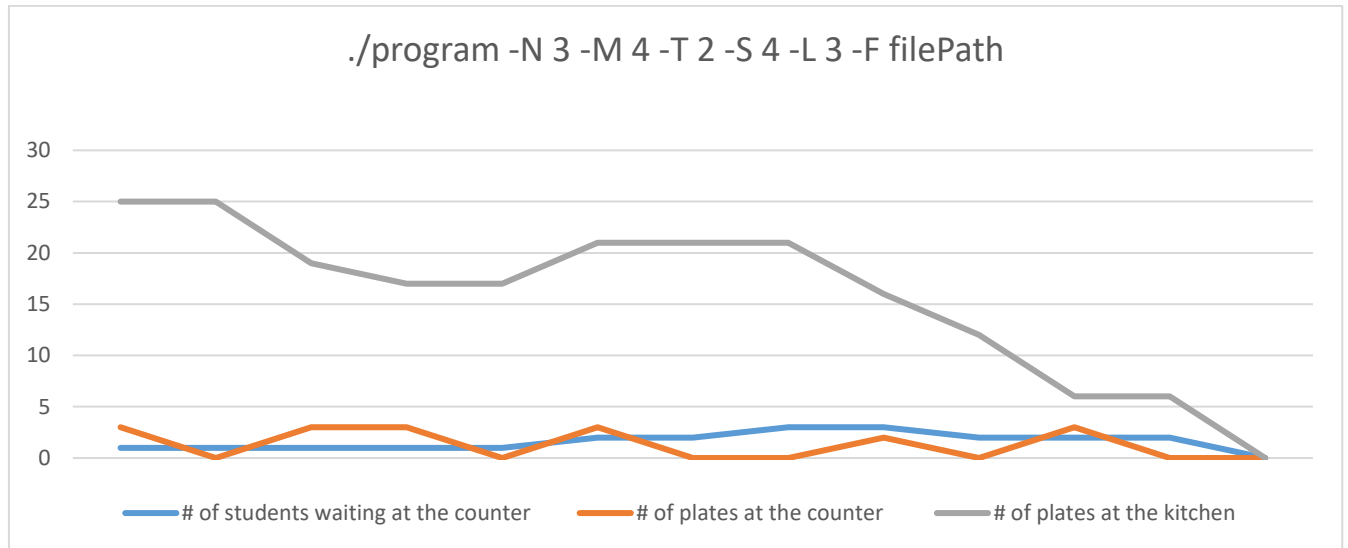
```
sed@sed:~/Desktop/midterm$ make all
gcc -c program.c
gcc program.o -lm -o program -pthread -lrt
sed@sed:~/Desktop/midterm$ ./program -N 3 -M 3 -T 2 -S 4 -L 1 -F filePath
Should be M>N. Check your value!
: Success
sed@sed:~/Desktop/midterm$ ./program -N 2 -M 3 -T 2 -S 4 -L 1 -F filePath
Should be N>2. Check your value!
: Success
sed@sed:~/Desktop/midterm$ ./program -N 3 -M 4 -T 2 -S 4 -L 1 -F filePath
Should be L>=3. Check your value!
: Success
```

3. Pilot Graphics

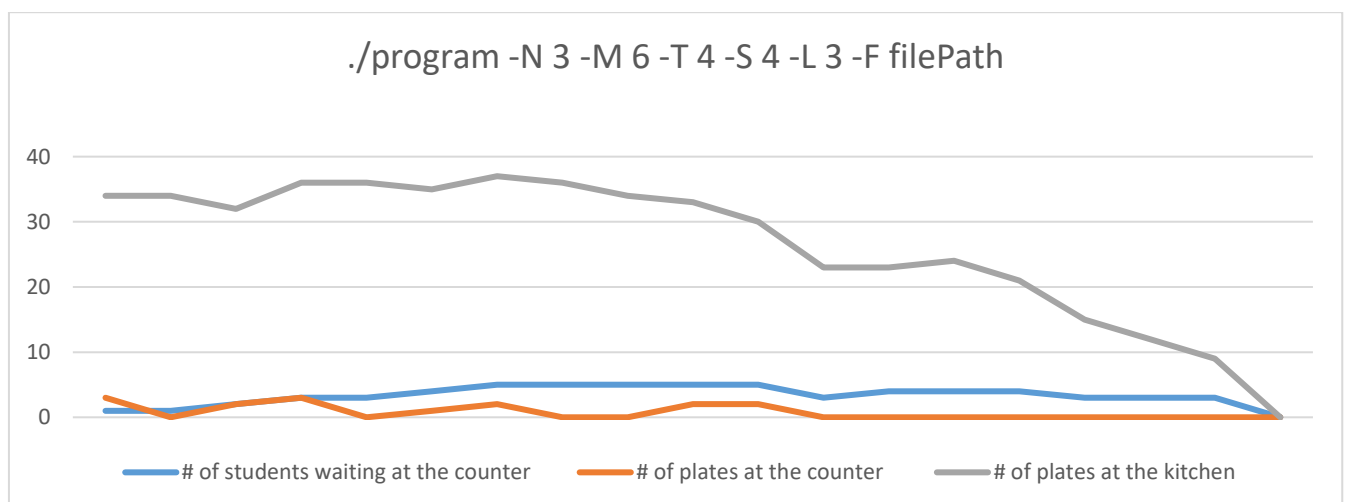
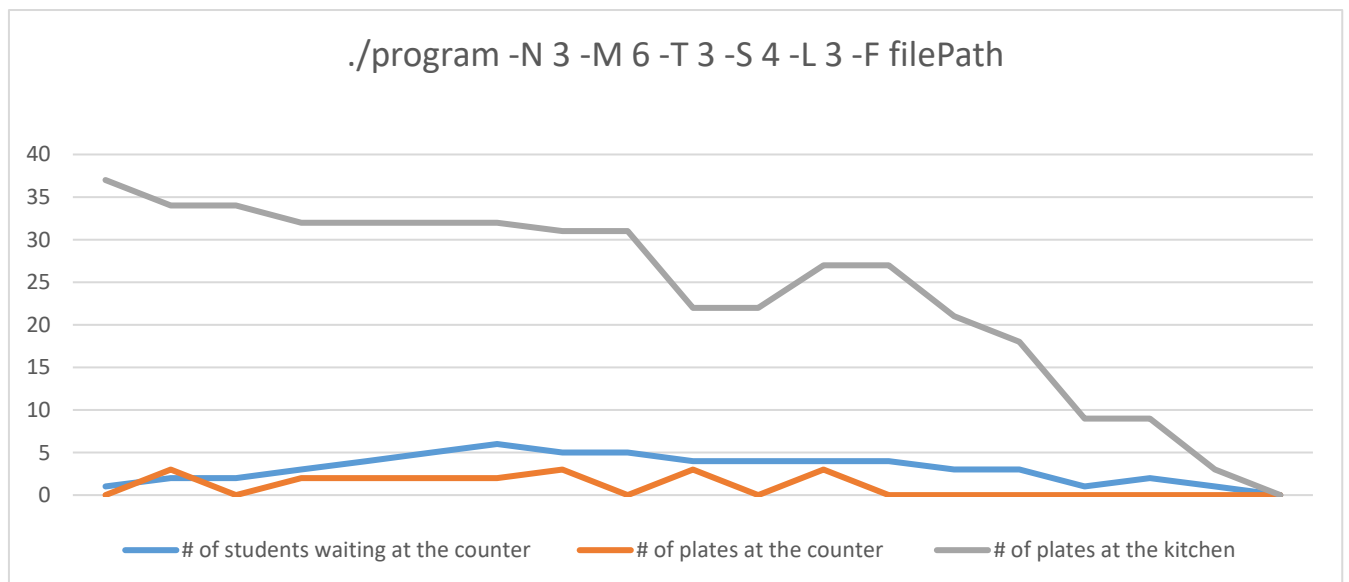
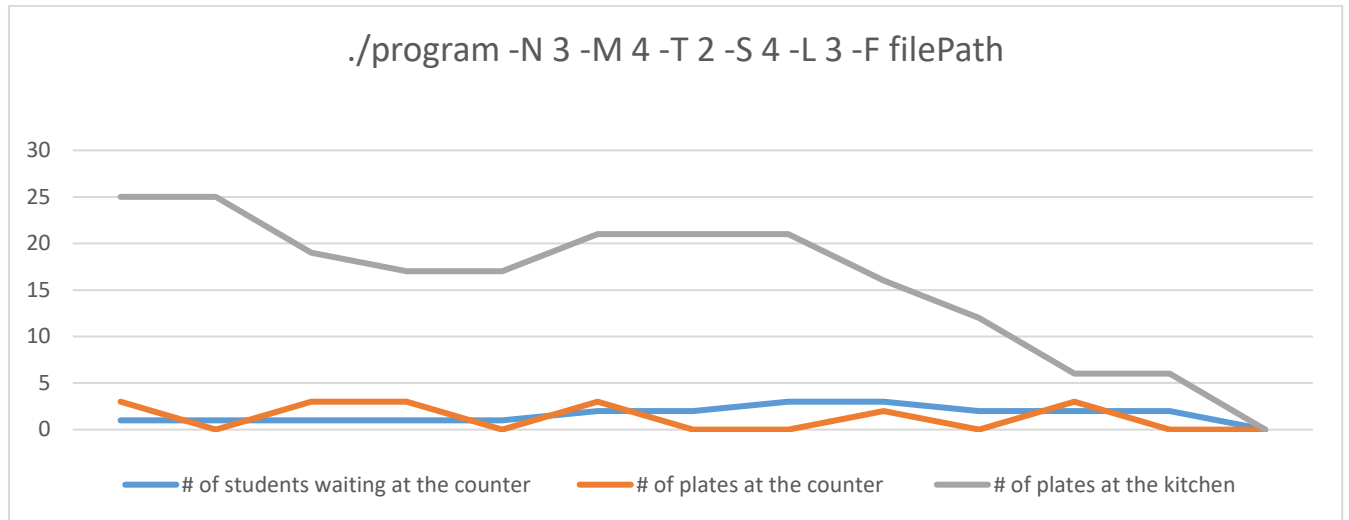
- N is not constant



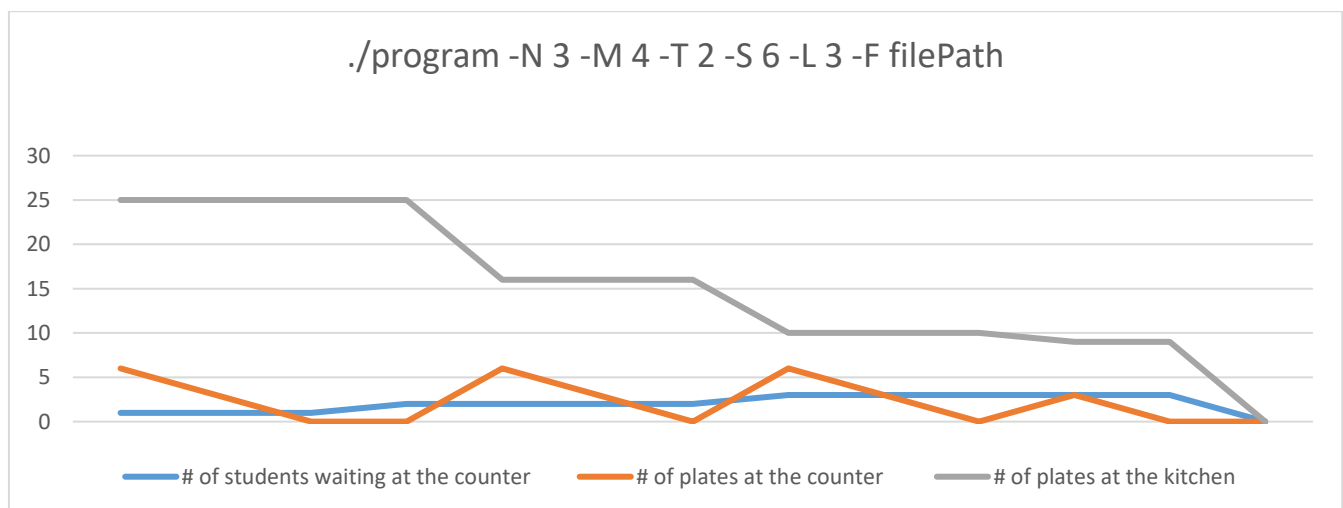
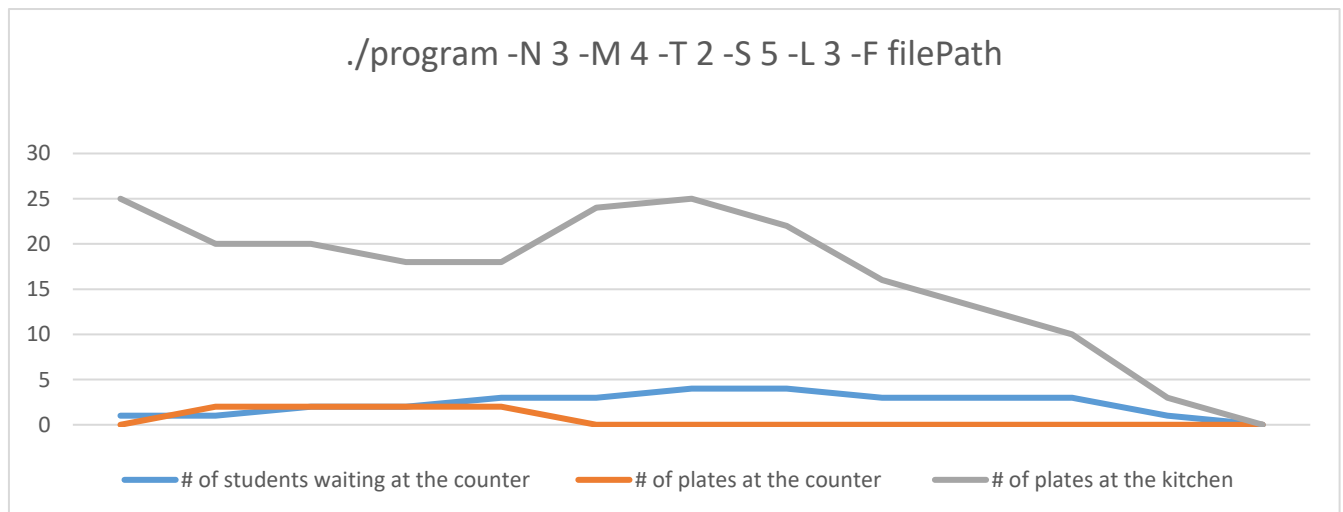
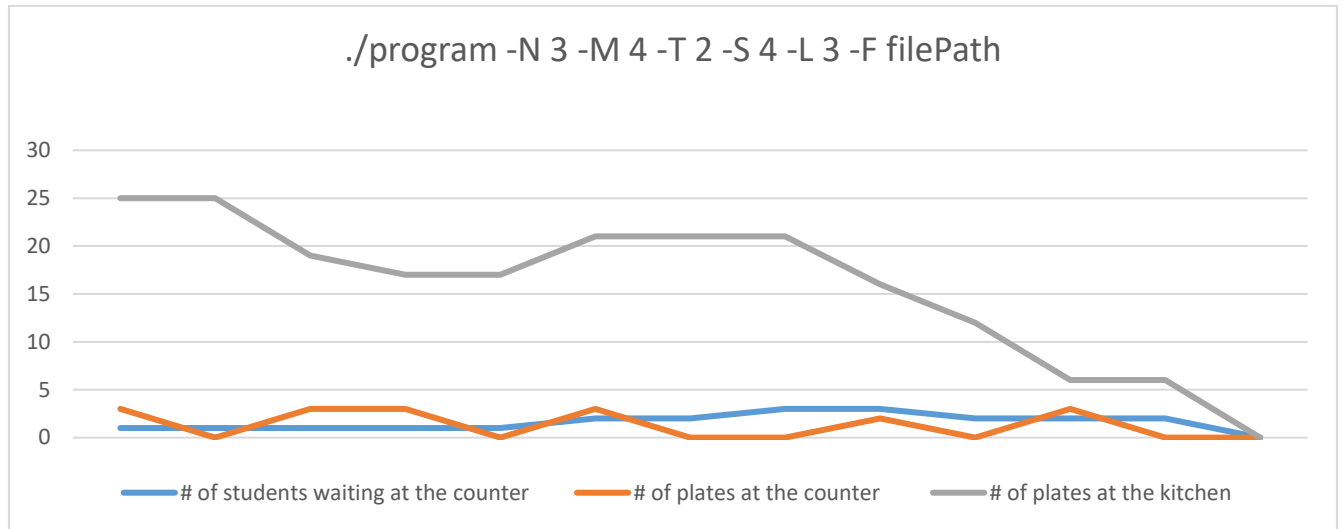
- M is not constant



- T is not constant



- S is constant



- L is constant

