

HW4

Sezer Demir

May 16, 2021

1 Requirements

For this homework I achieved all the requirements those are mentioned in homework.

2 Design Decisions

- a.) All arguments those are passed to program are checked. Arguments must be formed in following rules. Otherwise an error message will be displayed and program ends immediately.
 - 1.) First argument is first file's name. If file is not exist program will be terminated immediately.
 - 2.) Second argument is second file's name. If file is not exist program will be terminated immediately.
 - 3.) Third argument is total amount of money that student will have. It cannot be equal or less than 0 and it cannot include letters in it. Otherwise an error message will be displayed and program will be terminated immediately.
 - 4.) You must enter exactly 3 arguments. More or less argument will not work too. Otherwise an error message will be displayed and program will be terminated immediately.
- b.) I didn't check second input file's content is proper or not. It must be formed like in homework's pdf. For example, each line must be written like "odtulu 5 3 900". So delimiter must be " " and there will not be a missing information. Like odtulu missing and line consist of "5 3 900".
- c.) If second input file have not any student info, means empty file, message of "There is no student in the input file, so no homework is solved" will be printed.

- d.) Program just reads 'S', 'Q', and 'C' characters from first input file.
So priorities must be written in uppercase like in homework pdf.
Otherwise it will skip them.

3 How It Works

I used a structure named "Students" to store each of hired-student's information. Their Q S C values and unnamed semaphore object. Also I created 3 different arrays. Each array stores hired-student indices according to their best aspect. Lets say we have 3 hired-students and their Q values are 2, 5, and 1. So the array that have indices of hired-students according to their Q values will be like that: [1,0,2] since second one have best Q value, first one second and third one last. So when main thread want to check best hired-student that has highest Q value, then it will check that the student is available at the moment (not solving any homework) by looking at array's first element. That element, which is '1', indicates index of the student so it's second thread that have best Q value. If it is not available, then it will check first hired-student thread which have index of '0' since array's next element is '0'. So I have 2 more arrays those have indices according to best values of 'C'(ascending order) and 'S'(descending order). So when main thread gets a homework to deliver, it can check best student according to priority of the homework.

Main thread checks if a hired-student is available or not by looking at available array. Each element of the array is '0' or '1'. If it is '1', that means hired-student, which have same index with element, is available at the moment. Lets say array is like that: [0,0,1], then that means third thread is available at the moment. It cannot send the homework first or second thread. Whenever main thread sends a homework to a hired-student, it just make the element '0' so it knows that thread is not available at the moment for next turns.

When main thread finds best available student for the homework it will deliver, it just post the semaphore of the associated hired-student thread. So that thread can print a proper message and sleep according to it's speed. When that student awakes from sleeping, it just turns element of available array, associated with it's own index, into '1'. So main thread can deliver new homework to it again now.

Thread-g just reads the input-file-1 character by character and puts them in queue. Each time it receives a homework from file and puts into queue prints it's current total amount of money left. If it's money is not enough to have homeworks to be solved or there is no more homeworks in input-file-1, then it prints a proper message and terminates itself.

4 OUTPUTS

```

sezer@sezer-Lenovo-V50-70:~/Desktop/hw$ valgrind --leak-check=full --show-leak-kinds=all --track-origins=yes --verbose --log-file=valgrind-out.txt --trace-childreny
es ./hw$ input1 input2 1
Amount of total money must consist of digits and be a positive number!
sezer@sezer-Lenovo-V50-70:~/Desktop/hw$ valgrind --leak-check=full --show-leak-kinds=all --track-origins=yes --verbose --log-file=valgrind-out.txt --trace-childreny
es ./hw$ input1 input2 1
Amount of total money must consist of digits and be a positive number!
sezer@sezer-Lenovo-V50-70:~/Desktop/hw$ valgrind --leak-check=full --show-leak-kinds=all --track-origins=yes --verbose --log-file=valgrind-out.txt --trace-childreny
es ./hw$ input1 inputx 1000
File named as "inputx" does not exist!
sezer@sezer-Lenovo-V50-70:~/Desktop/hw$ valgrind --leak-check=full --show-leak-kinds=all --track-origins=yes --verbose --log-file=valgrind-out.txt --trace-childreny
es ./hw$ input1 input2 1000 555
You must enter exactly 3 argument(input file 1, input file 2, amount of money)

```

Figure 1: Wrong argument example

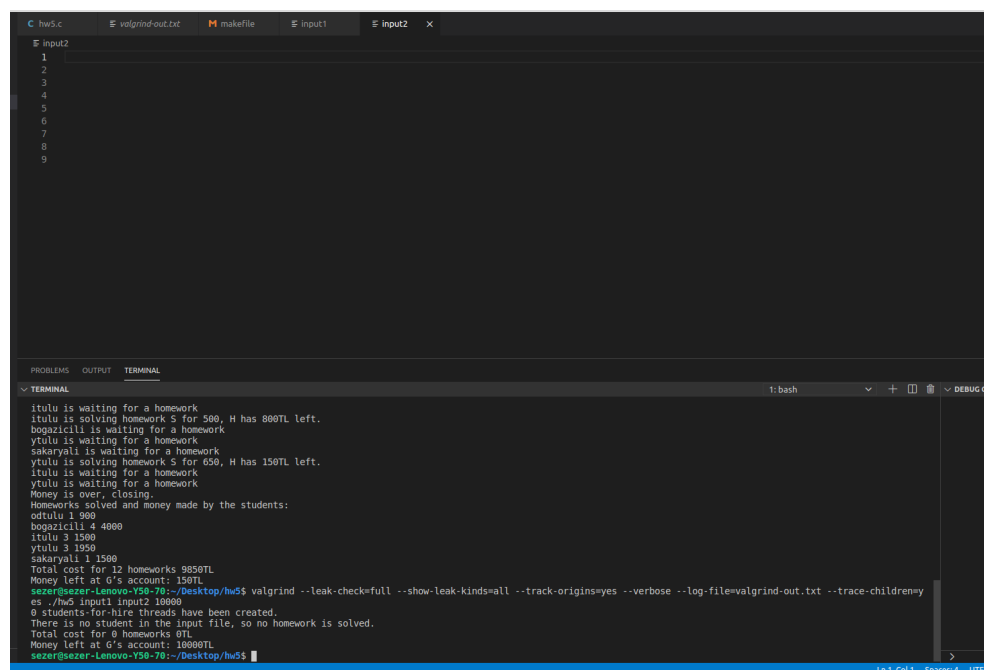


Figure 2: Example of no student info in file

```
1 odutulu 5 3 900
2 bogazicili 4 5 1000
3 itulu 4 4 500
4 ytulu 3 4 650
5 sakaryali 1 2 1500
6
7
8
9
10
11
12
13
```

```

bogazicili is solving homework 5 for 1000, H has 2950TL left.
ytulu is waiting for a homework
ytulu is solving homework 5 for 650, H has 2300TL left.
odutulu is waiting for a homework
bogazicili is waiting for a homework
bogazicili is solving homework 5 for 1000, H has 1300TL left.
itulu is waiting for a homework
itulu is solving homework 5 for 500, H has 800TL left.
bogazicili is waiting for a homework
sakaryali is waiting for a homework
ytulu is waiting for a homework
ytulu is solving homework 5 for 650, H has 150TL left.
itulu is waiting for a homework
ytulu is waiting for a homework
Money is over, closing.
Homeworks solved and money made by the students:
odutulu 1 900
bogazicili 4 4000
itulu 3 1500
ytulu 3 1950
sakaryali 1 1500
Total cost for 12 homeworks 9850TL
Money left at G's account: 150TL
sezer@sezer-Lenovo-Y50-70:~/Desktop/hw5$
```

Figure 3: Proper run