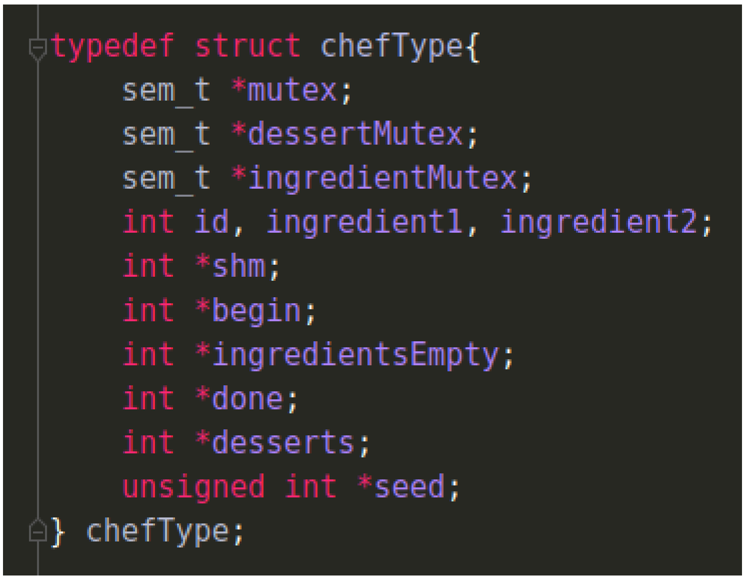
CSE344 SYSTEMS PROGRAMMING
HW04 REPORT

Emire Korkmaz
141044043

In this homework, we manage one wholesaler’s and six chefs’ works in order to make the dessert.

Firstly, I created a structure for chefs to hold all the necessary information that would be used in the dessert creation and ingredient transfer processes.

The type that contains a chef’s needs is shown below.



```
typedef struct chefType{
    sem_t *mutex;
    sem_t *dessertMutex;
    sem_t *ingredientMutex;
    int id, ingredient1, ingredient2;
    int *shm;
    int *begin;
    int *ingredientsEmpty;
    int *done;
    int *desserts;
    unsigned int *seed;
} chefType;
```

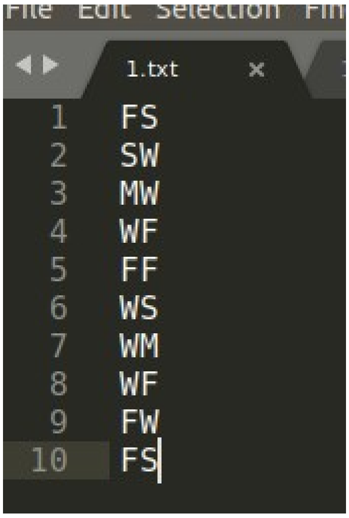
The idea in this homework is using an array with the size of 4 (each index is for one ingredient) and whoever comes in, takes the ingredient that s/he needs by decreasing the related index or adds the ingredient that is read from the file by increasing the related index.

We let the wholesaler start first in order to fill the shared array. After the first ingredients come in, the wholesaler lets all the chefs know that the array is being filled. There is a shared semaphore that is used between chefs to work in synchronization. Every chefs check the array’s indices. If the ingredient they need is there, they decrease the index and checks for the other ingredient if there it is, they decrease it as well, if not, they wait for their turn come again. Since they need for 2 missing ingredients, after getting these two, they start preparing the dessert. There is another semaphore for preparing dessert, too. Because the chefs put the desserts on the same place (by increasing the shared value) and the wholesaler takes the dessert from the same place (by decreasing the shared value).

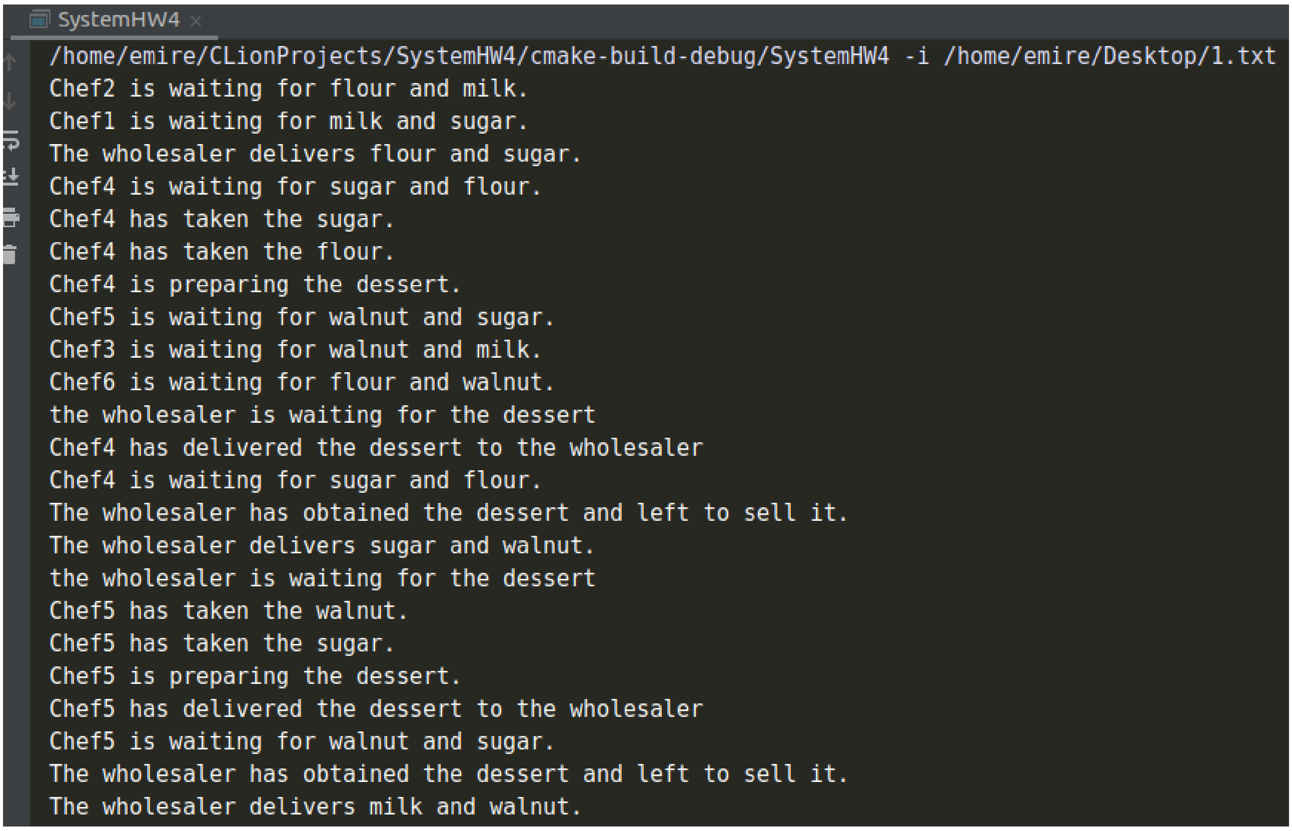
While obtaining the desserts and going to selling, wholesaler locks the semaphores and by using this, the wholesaler makes all chefs wait, until s/he comes back.

This process continues until all the ingredients run out and there are no more desserts.

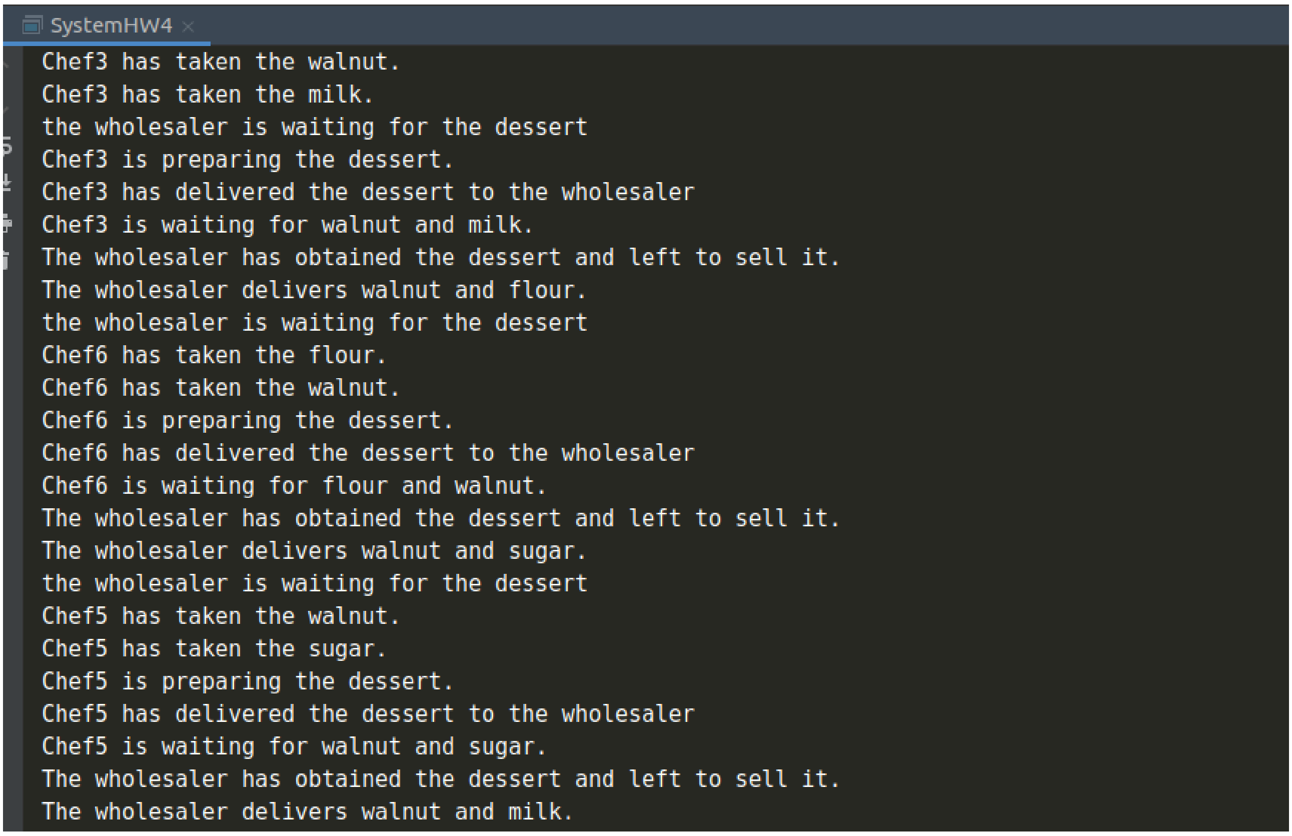
Example input file and output are shown below.



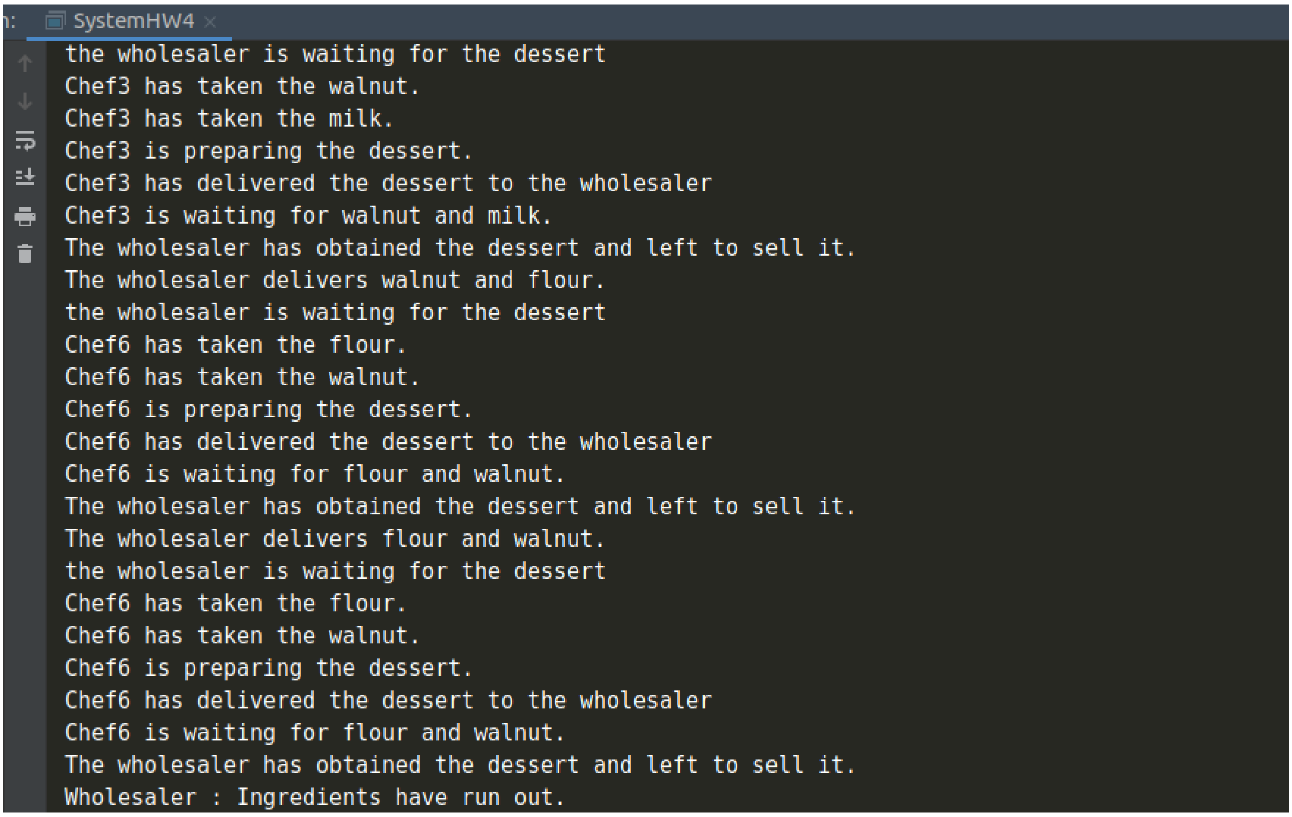
```
FS
SW
MW
WF
FF
WS
WM
WF
FW
FS
```



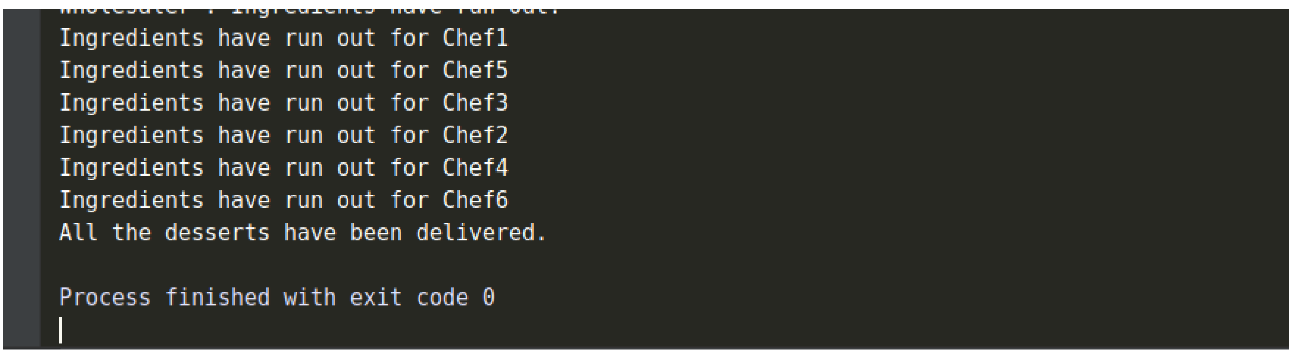
```
/home/emire/CLionProjects/SystemHW4/cmake-build-debug/SystemHW4 -i /home/emire/Desktop/1.txt
Chef2 is waiting for flour and milk.
Chef1 is waiting for milk and sugar.
The wholesaler delivers flour and sugar.
Chef4 is waiting for sugar and flour.
Chef4 has taken the sugar.
Chef4 has taken the flour.
Chef4 is preparing the dessert.
Chef5 is waiting for walnut and sugar.
Chef3 is waiting for walnut and milk.
Chef6 is waiting for flour and walnut.
the wholesaler is waiting for the dessert
Chef4 has delivered the dessert to the wholesaler
Chef4 is waiting for sugar and flour.
The wholesaler has obtained the dessert and left to sell it.
The wholesaler delivers sugar and walnut.
the wholesaler is waiting for the dessert
Chef5 has taken the walnut.
Chef5 has taken the sugar.
Chef5 is preparing the dessert.
Chef5 has delivered the dessert to the wholesaler
Chef5 is waiting for walnut and sugar.
The wholesaler has obtained the dessert and left to sell it.
The wholesaler delivers milk and walnut.
```



```
Chef3 has taken the walnut.
Chef3 has taken the milk.
the wholesaler is waiting for the dessert
Chef3 is preparing the dessert.
Chef3 has delivered the dessert to the wholesaler
Chef3 is waiting for walnut and milk.
The wholesaler has obtained the dessert and left to sell it.
The wholesaler delivers walnut and flour.
the wholesaler is waiting for the dessert
Chef6 has taken the flour.
Chef6 has taken the walnut.
Chef6 is preparing the dessert.
Chef6 has delivered the dessert to the wholesaler
Chef6 is waiting for flour and walnut.
The wholesaler has obtained the dessert and left to sell it.
The wholesaler delivers walnut and sugar.
the wholesaler is waiting for the dessert
Chef5 has taken the walnut.
Chef5 has taken the sugar.
Chef5 is preparing the dessert.
Chef5 has delivered the dessert to the wholesaler
Chef5 is waiting for walnut and sugar.
The wholesaler has obtained the dessert and left to sell it.
The wholesaler delivers walnut and milk.
```



```
the wholesaler is waiting for the dessert
Chef3 has taken the walnut.
Chef3 has taken the milk.
Chef3 is preparing the dessert.
Chef3 has delivered the dessert to the wholesaler
Chef3 is waiting for walnut and milk.
The wholesaler has obtained the dessert and left to sell it.
The wholesaler delivers walnut and flour.
the wholesaler is waiting for the dessert
Chef6 has taken the flour.
Chef6 has taken the walnut.
Chef6 is preparing the dessert.
Chef6 has delivered the dessert to the wholesaler
Chef6 is waiting for flour and walnut.
The wholesaler has obtained the dessert and left to sell it.
The wholesaler delivers flour and walnut.
the wholesaler is waiting for the dessert
Chef6 has taken the flour.
Chef6 has taken the walnut.
Chef6 is preparing the dessert.
Chef6 has delivered the dessert to the wholesaler
Chef6 is waiting for flour and walnut.
The wholesaler has obtained the dessert and left to sell it.
Wholesaler : Ingredients have run out.
```



```
Wholesaler : Ingredients have run out.
Ingredients have run out for Chef1
Ingredients have run out for Chef5
Ingredients have run out for Chef3
Ingredients have run out for Chef2
Ingredients have run out for Chef4
Ingredients have run out for Chef6
All the desserts have been delivered.

Process finished with exit code 0
```

Tests:

I created files with different sizes. They all worked correctly. Minimum size of the given file was not clear. So, I assume the minimum size as 10 lines each contains 2 distinct ingredient. (As written in the homework).

Also I created a file with the size of 163.000 lines. It took a lot of time, I had to terminate it. Since it worked correctly for a long time, if there is enough time, it could work.

If any other letter (except M, F, W, S) is written in the given file, this ingredient pair will be ignored. If the given pair of ingredients are not distinct, then this will be ignored, as well.

Uppercase or lowercase characters do not matter. Both are considered as valid input.