

# CSE344 – System Programming - Homework #5 Report

## POSIX threads and synchronization thereof

Student Name: Ozan Şelte

Student Number: 161044061

*In this homework to be POSIX compatible with the latest standart I decided to define “\_POSIX\_C\_SOURCE” as “200809”, “\_FILE\_OFFSET” as “64” and “\_GNU\_SOURCE”.*

### **Problem #1:**

Flowers’ counts and types are arbitrary. They can change, initialize and work dynamically.

### **Solution #1:**

I have initialized global pointer and space at heap for the names. After that, I used them for florists and clients. I kept them as an array of char pointers.

### **Problem #2:**

Florists’ counts, names and the count of flowers hey sell are arbitrary. They can change, initialize and work dynamically.

### **Solution #2:**

I have initialized global pointer and space at heap for the structure. I kept them as a struct array.

### **Problem #3:**

Client count for a florist is arbitrary, they do change anytime. The queue is advised on homework PDF.

### **Solution #3:**

Every florist structure has a client queue. When a thread is running, it enters the critical section, copies the first client from the queue, iterates it and finally exits from the critical section. After that, florists can wait and deliver the flower.

### Problem #4:

Main thread and flower threads should not cause deadlock and data corruption.

### Solution #4:

I have used a mutex and a condition variable. Florists are waiting for the queue.

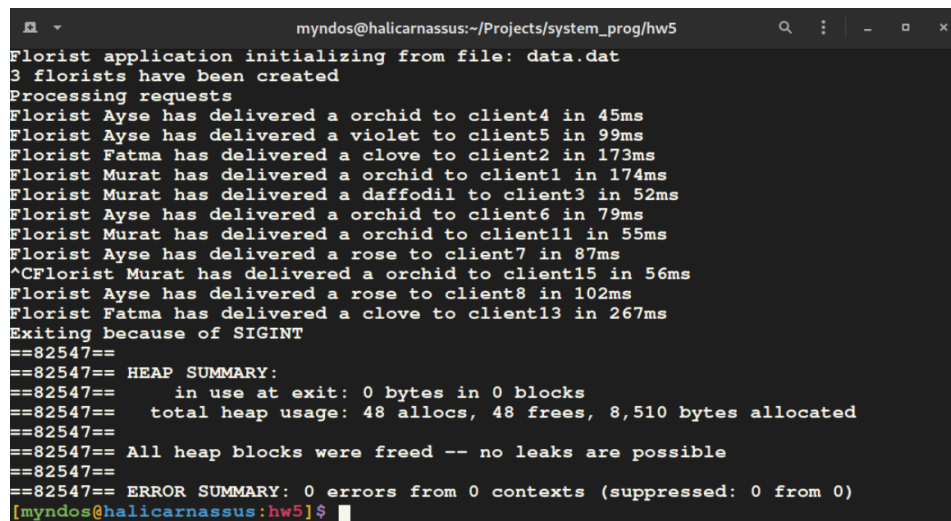
### Problem #5:

In the case of a CTRL+C, SIGINT, all resources must be freed. The free function is not an async signal handler safe function.

### Solution #5:

A signal handler links to the process, all threads. After the end of the thread creation, the main thread blocks the SIGINT signal for itself. When a signal came, it goes to a single thread and we cannot know which. The thread that takes the signal, changes the global static sig\_atomic\_t variable, and every child thread checks this flag regularly.

If someone catch the change, sends all other threads a cancel signal. Because of all threads are PTHREAD\_CANCEL\_ASYNCRONOUS, all of them closes but a single one. Then frees and destroys resources.



```
myndos@halicarnassus:~/Projects/system_prog/hw5
Florist application initializing from file: data.dat
3 florists have been created
Processing requests
Florist Ayse has delivered a orchid to client4 in 45ms
Florist Ayse has delivered a violet to client5 in 99ms
Florist Fatma has delivered a clove to client2 in 173ms
Florist Murat has delivered a orchid to client1 in 174ms
Florist Murat has delivered a daffodil to client3 in 52ms
Florist Ayse has delivered a orchid to client6 in 79ms
Florist Murat has delivered a orchid to client11 in 55ms
Florist Ayse has delivered a rose to client7 in 87ms
^CFlorist Murat has delivered a orchid to client15 in 56ms
Florist Ayse has delivered a rose to client8 in 102ms
Florist Fatma has delivered a clove to client13 in 267ms
Exiting because of SIGINT
==82547==
==82547== HEAP SUMMARY:
==82547==   in use at exit: 0 bytes in 0 blocks
==82547==   total heap usage: 48 allocs, 48 frees, 8,510 bytes allocated
==82547==
==82547== All heap blocks were freed -- no leaks are possible
==82547==
==82547== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
[myndos@halicarnassus:hw5]$
```

### Problem #6:

Florists prints an goodbye message when there is not any client left.

### Solution #6:

When file finishes, the main thread changes a variable for every florist. If the variable shows the file ended and queue is empty, florist exits.