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.

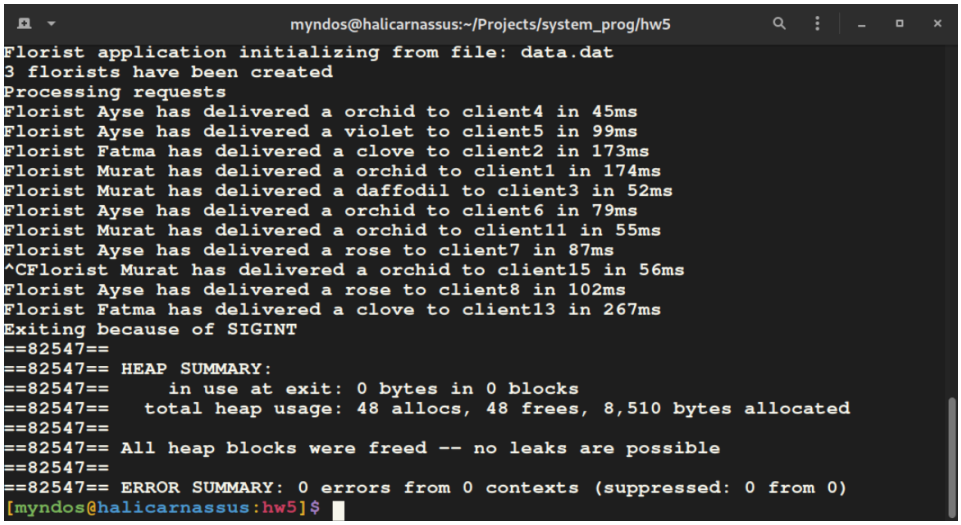
Problem #5:

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.



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.