COMP 310 – Assignment 1
Sami Riaz (260630430)

Q1. **Uni-programming** is a term used to describe computers that are only able to do a single task at a time. This is because they are only able to hold one task in the main memory at a time. In uni-programming, an entire process must be completed before proceeding, therefore there is no need for the concept of processes in uni-programming.

Multi-programming allows multiple programs to be held in main memory at the same time. Still only one program is active at a time, but multi-programming has the ability to switch between programs. Therefore, while one program is loading/waiting we can run another program.

In a **Time-Sharing System**, each users' console is connected to a main computer which is in charge of running the programs on each console sequentially. These systems were used around the 1950s because building strong computers was expensive, therefore it was more economical to connect all the consoles to a stronger 'main' computer.

Q2. **Uni-programming System:**

A: 2msec (run time)
10msec (wait time)
+ 4msec (run time)
16msec (total time)

B: 2msec (run time)
10msec (wait time)
+ 4msec (run time)
16msec (total time)

C: 2msec (run time)
10msec (wait time)
+ 4msec (run time)
16msec (total time)

Total time: 16 + 16 + 16 = 48msec
Therefore, all 3 programs end at t = 48msec in a uni-programming system.

Multi-programming System:

Each program takes 16msec in this case too, but this time the next program starts running when the previous one is waiting. Therefore,

t = 0msec -> A starts running for 2msec
t = 2msec -> A waits on resource for 10msec, B starts running for 2msec

t = 4msec -> B waits on resource for 10msec, C starts running for 2msec
t = 6msec -> C waits on resource for 10msec
t = 12msec -> A starts running for 4msec
t = 14msec -> B starts running for 4msec
t = 16msec -> C starts sunning for 4msec, A is completed
t = 18msec -> B is completed
t = 20msec -> C is completed

Therefore, all 3 programs end at t = 20msec in a multi-programming system.

Q3.

```
#include <stdio.h>
#include <unistd.h>
#include <fcntl.h>
int main()
{
        int fd;

        close(1);
        fd = open("redirect.txt", O_CREAT | O_RDWR |  O_TRUNC, 0666);

        printf("A simple program output.");
        return 0;
}
```