Quiz 4

Course: Operating Systems
Section: BSE-5A
Name: Tayyab Kamran Sami

Course Code: CS 2006
Total Marks:10
RollNo: 22i-2505

Question 1: **[10 Marks]**

A parking garage can accommodate up to 50 cars. Each car entering the garage must wait if the garage is full, and cars can only enter or exit one at a time due to limited space at the entrance. Create a synchronization solution with **binary and counting semaphores** to:

- Allow cars to enter and leave safely, ensuring only one car is moving in or out at any given moment.
- Track the availability of parking spaces so that cars entering the garage wait if it's full.

You can assume that there are two functions carEnter(int car_id) and carExit(int car_id) that are used, you need to define these functions, on successful entry or exit you will print ("car 123 successfully Enter or Exit the garage")

```
full = 0; empty = N; //where N is the number of empty space
Semaphore check = 1;        int N = 50

void carEnter(int car-id)
{ do {
    wait(empty);
    wait(check);
cout<<"Car "<< car-id <<
 "successfully enter the
 garage"<<endl;
   signal(check);
   signal(full);
  }
  while(1)
}
```

```
void carExit(int car_id) {
do{wait(full);
   wait(check);
cout << "Car " << car-id <<
" successfully exit the garage";
   Signal(check);
   signal(empty);

}
while(1)

}
```