REPORT

Operating System

Assignment simulation Based

Student Name: N Saranya

Registration no:11714211

Sec:EE034

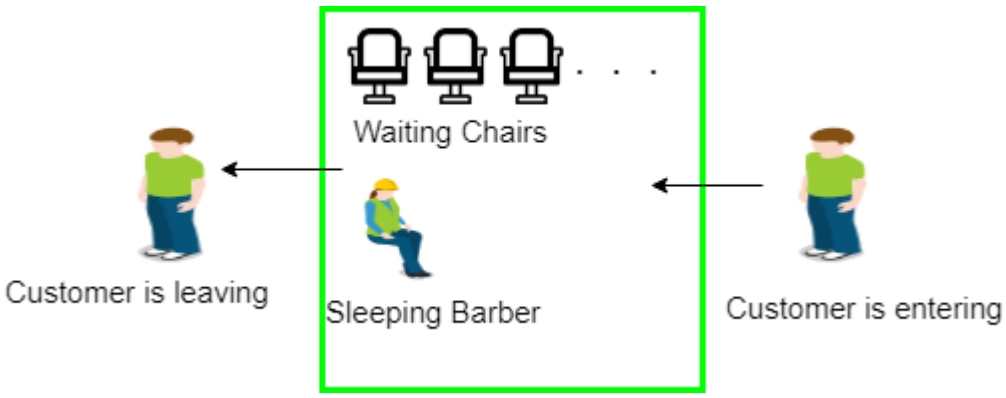
Roll no:63

Email address:saranyanittala@gmail.com

19) A barber consists of a waiting room with n chairs and a barber room with one barber chair. If there are no customers to be served ,the barber goes to sleep . If a customer enters the shop and all the chairs are occupied,then the customer leaves the shop.If the barber is busy but chairs are available ,then the customer sits in one of the free chairs. If the barber is asleep,then customer wakes the barber. Write a program to coordinate the barber and the customers.Using synchronization tools like locks,semaphores and monitors provide a solution to his problem.

PROBLEM: "The analogy is based upon a hypothetical barber shop with one barber. There is a barber shop which has one barberone barber chair, and n chairs for waiting for customers if there are any to sit on the chair.

- If there is no customer, then the barber sleeps in his own chair.
- When a customer arrives, he has to wake up the barber.
- If there are many customers and the barber is cutting a customer's hair, then the remaining customers either wait if there are empty chairs in the waiting room or they leave if no chairs are empty.



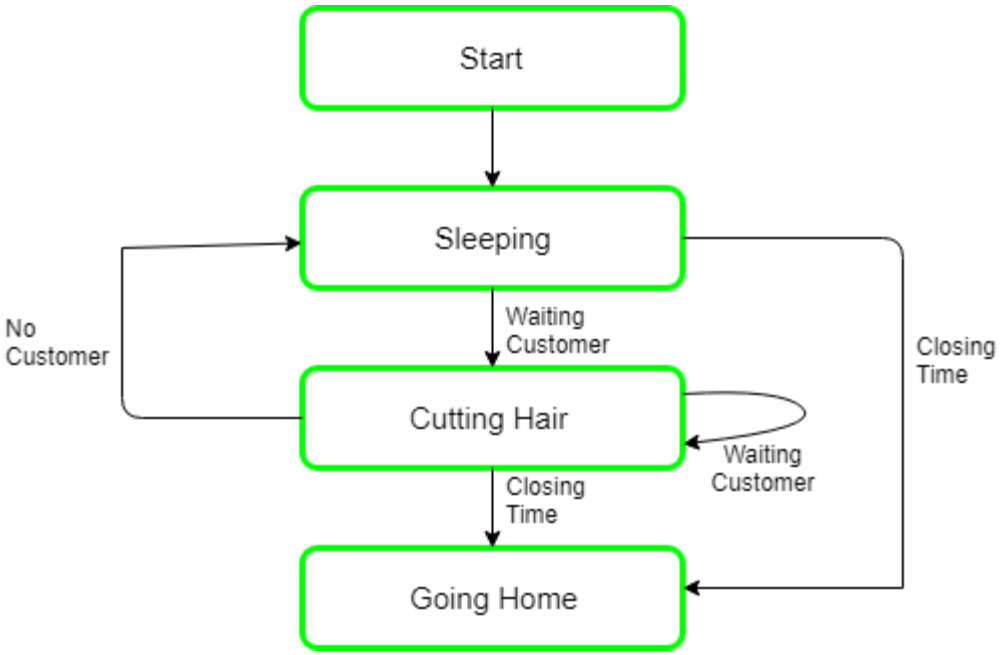
Solution : The solution to this problem includes three semaphores.First is for the customer which counts the number of customers present in the waiting room (customer in the barber chair is not included because he is not waiting). Second, the barber 0 or 1 is used to tell whether the barber is idle or is working, And the third mutex is used to provide the mutual exclusion which is required for the process to execute. In the solution, the customer has the record of the number of customers waiting in the waiting room if the number of customers is equal to the number of chairs in the waiting room then the upcoming customer leaves the barbershop.

When the barber shows up in the morning, he executes the procedure barber, causing him to block on the semaphore customers because it is initially 0. Then the barber goes to sleep until the first customer comes up.

When a customer arrives, he executes customer procedure the customer acquires the mutex for entering the critical region, if another customer enters thereafter, the second one will not be able to anything until the first one has released the mutex. The customer then checks the chairs in the waiting room if waiting customers are less then the number of chairs then he sits otherwise he leaves and releases the mutex.

If the chair is available then customer sits in the waiting room and increments the variable waiting value and also increases the customer's semaphore this wakes up the barber if he is sleeping.

At this point, customer and barber are both awake and the barber is ready to give that person a haircut. When the haircut is over, the customer exits the procedure and if there are no customers in waiting room barber sleeps.



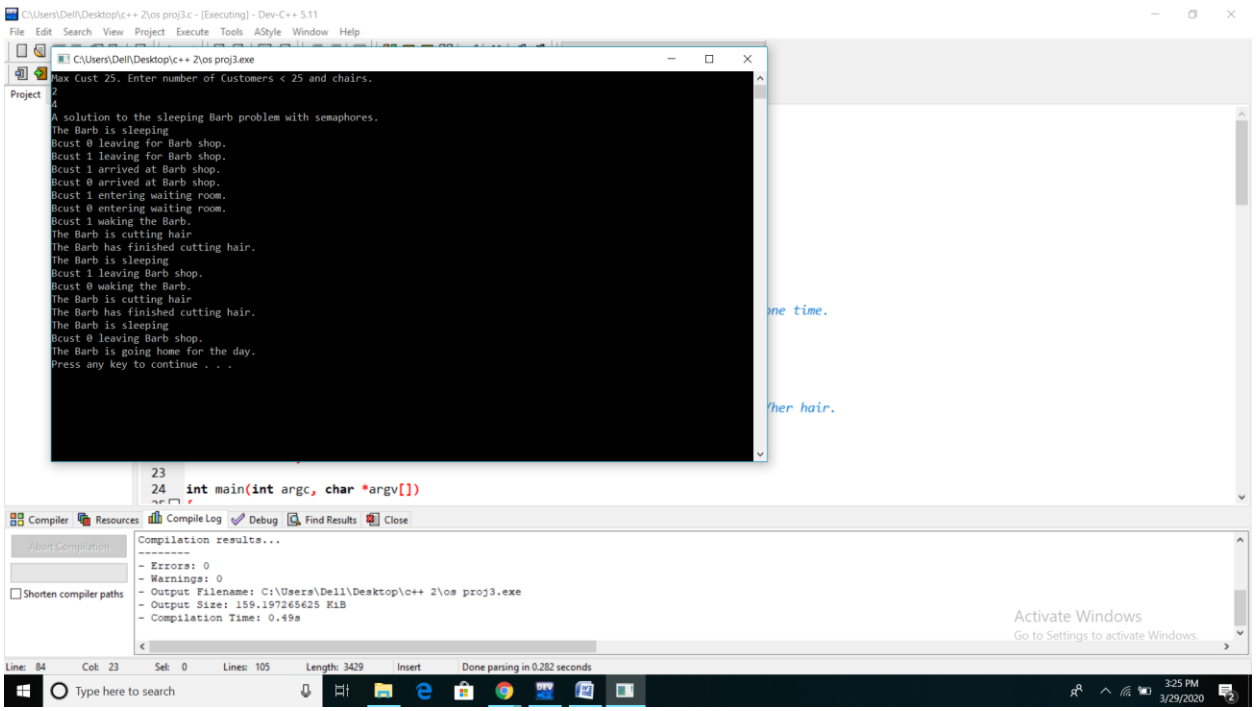
Algorithm for Sleeping Barber problem:

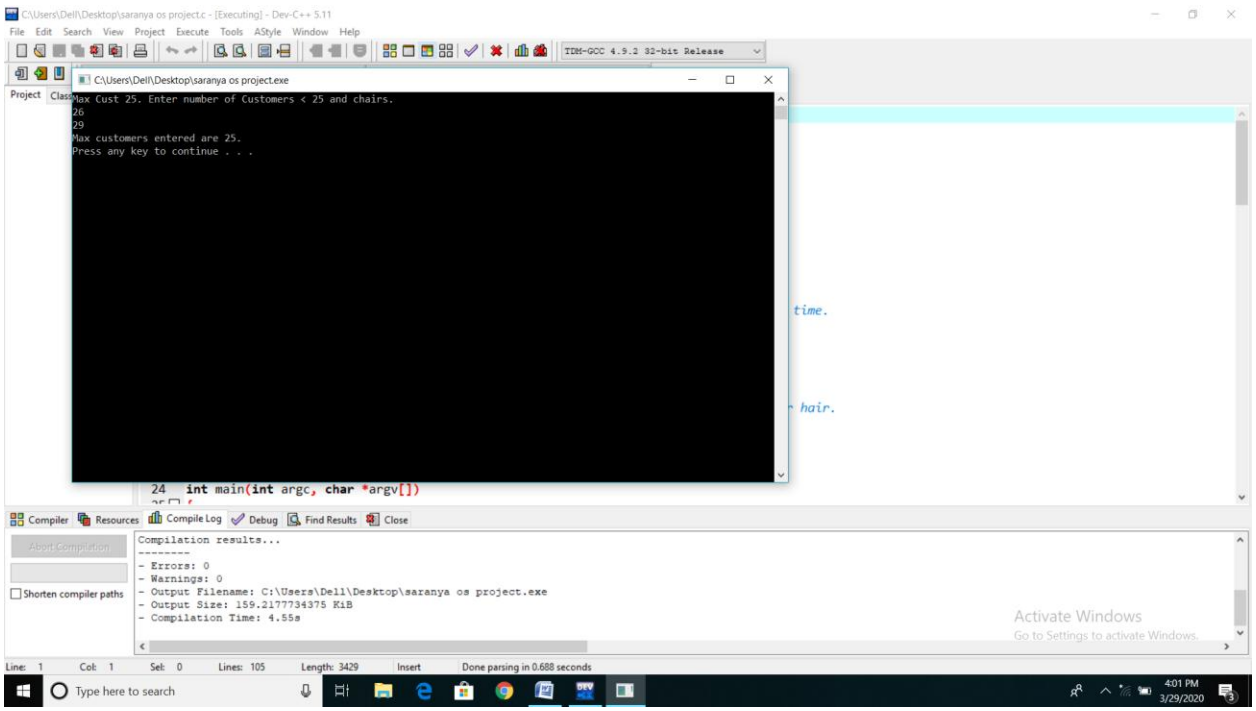
filter_none

brightness_4

```
Semaphore Customers = 0;
Semaphore Barber = 0;
Mutex Seats = 1;
int FreeSeats = N;

Barber {
      while(true) {
            /* waits for a customer (sleeps). */
            down(Customers);

            /* mutex to protect the number of available
seats.*/
            down(Seats);

            /* a chair gets free.*/
            FreeSeats++;

            /* bring customer for haircut.*/
            up(Barber);

            /* release the mutex on the chair.*/
            up(Seats);
```

```
            /* barber is cutting hair.*/
      }
}

Customer {
      while(true) {
            /* protects seats so only 1 customer tries to sit
               in a chair if that's the case.*/
            down(Seats); //This line should not be here.
            if(FreeSeats > 0) {
 /* sitting down.*/
                  FreeSeats--;

                  /* notify the barber. */
                  up(Customers);

                  /* release the lock */
                  up(Seats);

                  /* wait in the waiting room if barber is
busy. */
                  down(Barber);
                  // customer is having hair cut
            } else {
                  /* release the lock */
                  up(Seats);
                  // customer leaves
            }
      }
}
```


C:\Users\Dell\Desktop\c++ 2\os proj3.exe
Max Cust 25. Enter number of Customers < 25 and chairs.
2
4
A solution to the sleeping Barb problem with semaphores.
The Barb is sleeping
Bcust 0 leaving for Barb shop.
Bcust 1 leaving for Barb shop.
Bcust 1 arrived at Barb shop.
Bcust 0 arrived at Barb shop.
Bcust 1 entering waiting room.
Bcust 0 entering waiting room.
Bcust 1 waking the Barb.
The Barb is cutting hair
The Barb has finished cutting hair.
The Barb is sleeping
Bcust 1 leaving Barb shop.
Bcust 0 waking the Barb.
The Barb is cutting hair
The Barb has finished cutting hair.
The Barb is sleeping
Bcust 0 leaving Barb shop.
The Barb is going home for the day.
Press any key to continue . . .
int main(int argc, char *argv[])
Compiler
Resources
Compile Log
Debug
Find Results
Close
Abort Compilation
Shorten compiler paths
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Dell\Desktop\c++ 2\os proj3.exe
- Output Size: 159.197265625 KiB
- Compilation Time: 0.49s
Activate Windows
Go to Settings to activate Windows.
Done parsing in 0.282 seconds
Type here to search
3:25 PM
3/29/2020


C:\Users\Dell\Desktop\saranya os project.c - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 32-bit Release
C:\Users\Dell\Desktop\saranya os project.exe
Max Cust 25. Enter number of Customers < 25 and chairs.
26
29
Max customers entered are 25.
Press any key to continue . . .
time.
hair.
24 int main(int argc, char *argv[])
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Dell\Desktop\saranya os project.exe
- Output Size: 159.2177734375 KiB
- Compilation Time: 4.55s
Activate Windows
Go to Settings to activate Windows.
Line: 1 Col: 1 Sel: 0 Lines: 105 Length: 3429 Insert Done parsing in 0.688 seconds
Type here to search
4:01 PM
3/29/2020