Name: Aditya Somani
Roll No: T1851061
Div: A
PRN NO. 71901204L

ASSIGNMENT NO. 5

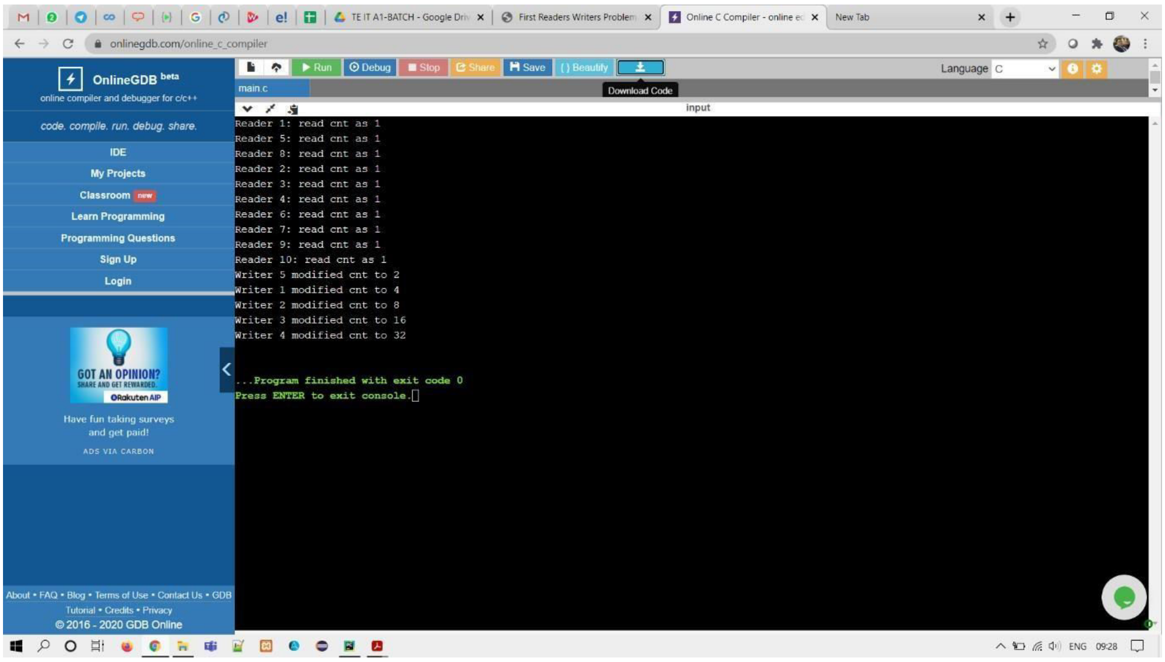
TITLE: Thread synchronization and mutual exclusion using mutex.

Program:

```
#include <pthread.h>

#include <semaphore.h>

#include <stdio.h>


/*

This program provides a possible solution for first readers writers problem
using mutex and semaphore.

I have used 10 readers and 5 producers to demonstrate the solution. You can
always play with these values.

*/


sem_t wrt;

pthread_mutex_t mutex;

int cnt = 1;

int numreader = 0;


void *writer(void *wno)

{

   sem_wait(&wrt);

   cnt = cnt*2;

   printf("Writer %d modified cnt to %d\n",(*((int *)wno)),cnt);

   sem_post(&wrt);
```

```
}
void *reader(void *rno)
{
    // Reader acquire the lock before modifying numreader
    pthread_mutex_lock(&mutex);
    numreader++;
    if(numreader == 1) {
        sem_wait(&wrt); // If this id the first reader, then it will block the writer
    }
    pthread_mutex_unlock(&mutex);
    // Reading Section
    printf("Reader %d: read cnt as %d\n",*((int *)rno),cnt);

    // Reader acquire the lock before modifying numreader
    pthread_mutex_lock(&mutex);
    numreader--;
    if(numreader == 0) {
        sem_post(&wrt); // If this is the last reader, it will wake up the writer.
    }
    pthread_mutex_unlock(&mutex);
}

int main()
{

    pthread_t read[10],write[5];
    pthread_mutex_init(&mutex, NULL);
    sem_init(&wrt,0,1);

    int a[10] = {1,2,3,4,5,6,7,8,9,10}; //Just used for numbering the producer and consumer

    for(int i = 0; i < 10; i++) {
        pthread_create(&read[i], NULL, (void *)reader, (void *)&a[i]);
    }
    for(int i = 0; i < 5; i++) {
        pthread_create(&write[i], NULL, (void *)writer, (void *)&a[i]);
    }

    for(int i = 0; i < 10; i++) {
        pthread_join(read[i], NULL);
    }
    for(int i = 0; i < 5; i++) {
        pthread_join(write[i], NULL);
    }

    pthread_mutex_destroy(&mutex);
    sem_destroy(&wrt);

    return 0;

}
```

Output:


TE IT A1-BATCH - Google Driv
First Readers Writers Problem
Online C Compiler - online ec
New Tab
onlinegdb.com/online_c_compiler
OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
GOT AN OPINION?
SHARE AND GET REWARDED.
Rakuten AIP
Have fun taking surveys and get paid!
ADS VIA CARBON
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2020 GDB Online
Run
Debug
Stop
Share
Save
{ } Beautify
Download Code
Language C
main.c
input
ENG
09:28

```
Reader 1: read cnt as 1
Reader 5: read cnt as 1
Reader 8: read cnt as 1
Reader 2: read cnt as 1
Reader 3: read cnt as 1
Reader 4: read cnt as 1
Reader 6: read cnt as 1
Reader 7: read cnt as 1
Reader 9: read cnt as 1
Reader 10: read cnt as 1
Writer 5 modified cnt to 2
Writer 1 modified cnt to 4
Writer 2 modified cnt to 8
Writer 3 modified cnt to 16
Writer 4 modified cnt to 32


...Program finished with exit code 0
Press ENTER to exit console.
```