**Assignment\_12 (Home Assignment)**

1. **Simulate Reader-Writer Problem using multiple processes**.

**//Program**

#include<semaphore.h>

#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<pthread.h>

sem\_t x,y;

pthread\_t tid;

pthread\_t writerthreads[100],readerthreads[100];

int readercount = 0;

void \*reader(void\* param)

{

sem\_wait(&x);

readercount++;

if(readercount==1)

sem\_wait(&y);

sem\_post(&x);

printf("%d reader is inside\n",readercount);

usleep(3);

sem\_wait(&x);

readercount--;

if(readercount==0)

{

sem\_post(&y);

}

sem\_post(&x);

printf("%d Reader is leaving\n",readercount+1);

return NULL;

}

void \*writer(void\* param)

{

printf("Writer is trying to enter\n");

sem\_wait(&y);

printf("Writer has entered\n");

sem\_post(&y);

printf("Writer is leaving\n");

return NULL;

}

int main()

{

int n2,i;

printf("Enter the number of readers:");

scanf("%d",&n2);

printf("\n");

int n1[n2];

sem\_init(&x,0,1);

sem\_init(&y,0,1);

for(i=0;i<n2;i++)

{

pthread\_create(&writerthreads[i],NULL,reader,NULL);

pthread\_create(&readerthreads[i],NULL,writer,NULL);

}

for(i=0;i<n2;i++)

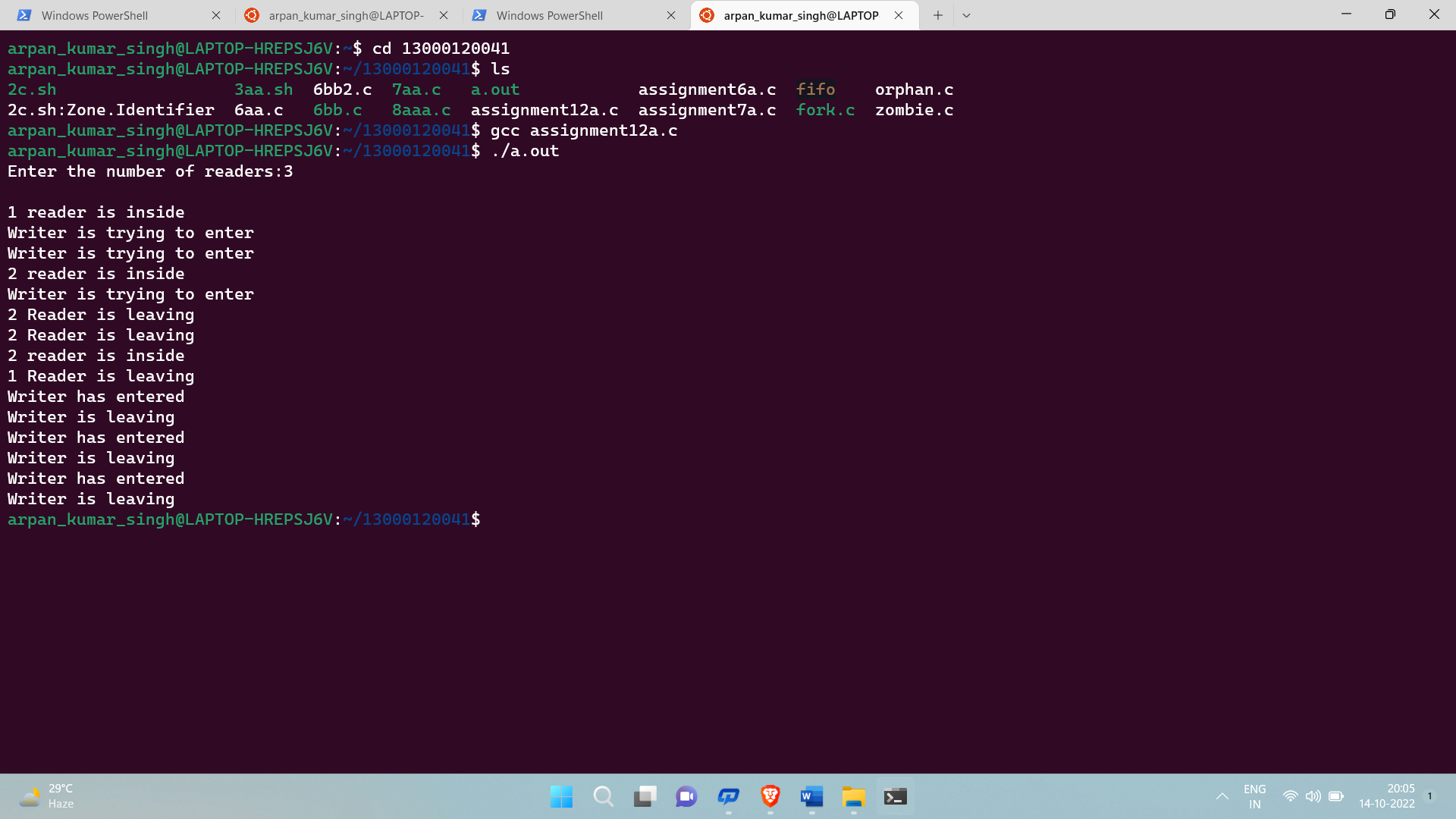
{

pthread\_join(writerthreads[i],NULL);

pthread\_join(readerthreads[i],NULL);

}

}



**2. Simulate the Dining Philosopher problem using multiple processes.**

**//Program**

#include<stdio.h>

#include<stdlib.h>

#include<pthread.h>

#include<semaphore.h>

#include<unistd.h>

sem\_t room;

sem\_t chopstick[5];

void \* philosopher(void \*);

void eat(int);

int main()

{

int i,a[5];

pthread\_t tid[5];

sem\_init(&room,0,4);

for(i=0;i<5;i++)

sem\_init(&chopstick[i],0,1);

for(i=0;i<5;i++){

a[i]=i;

pthread\_create(&tid[i],NULL,philosopher,(void \*)&a[i]);

}

for(i=0;i<5;i++)

pthread\_join(tid[i],NULL);

}

void \* philosopher(void \* num)

{

int phil=\*(int \*)num;

sem\_wait(&room);

printf("\nPhilosopher %d has entered room",phil);

sem\_wait(&chopstick[phil]);

sem\_wait(&chopstick[(phil+1)%5]);

eat(phil);

sleep(2);

printf("\nPhilosopher %d has finished eating",phil);

sem\_post(&chopstick[(phil+1)%5]);

sem\_post(&chopstick[phil]);

sem\_post(&room);

}

void eat(int phil)

{

printf("\nPhilosopher %d is eating",phil);

}

