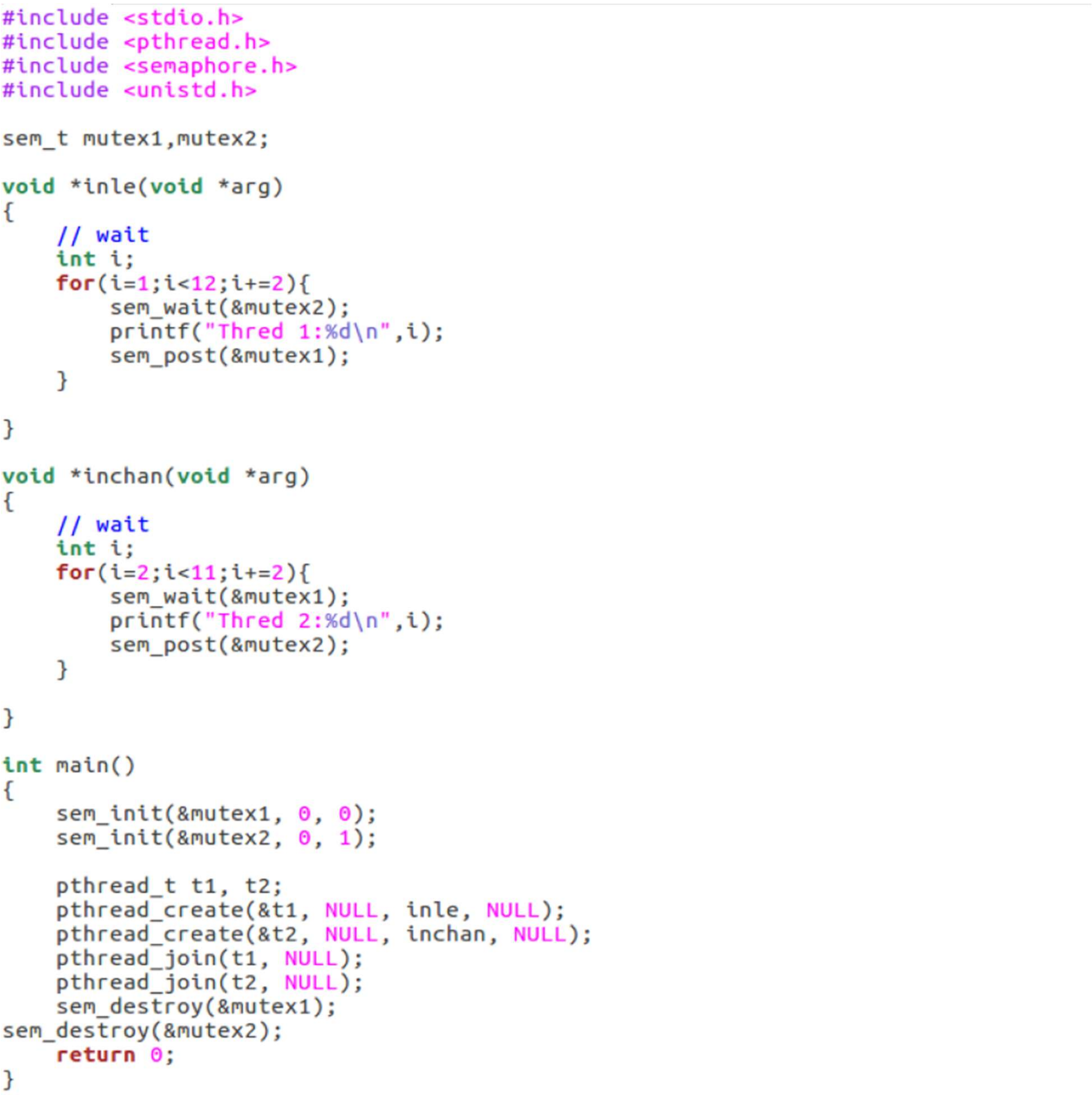
Họ và tên: Nguyễn Trọng Đạt

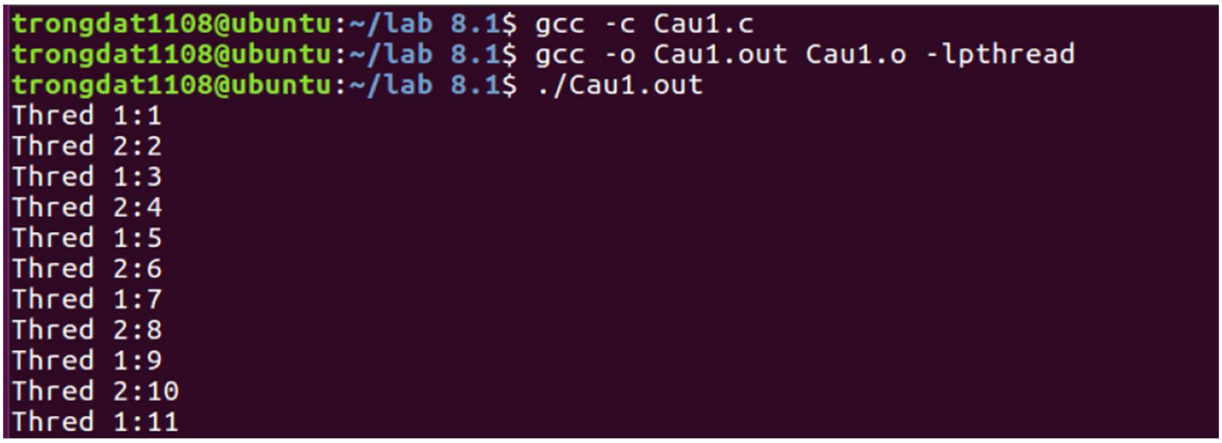
MSSV: 52100176

Lớp: 21050301

Câu 1:

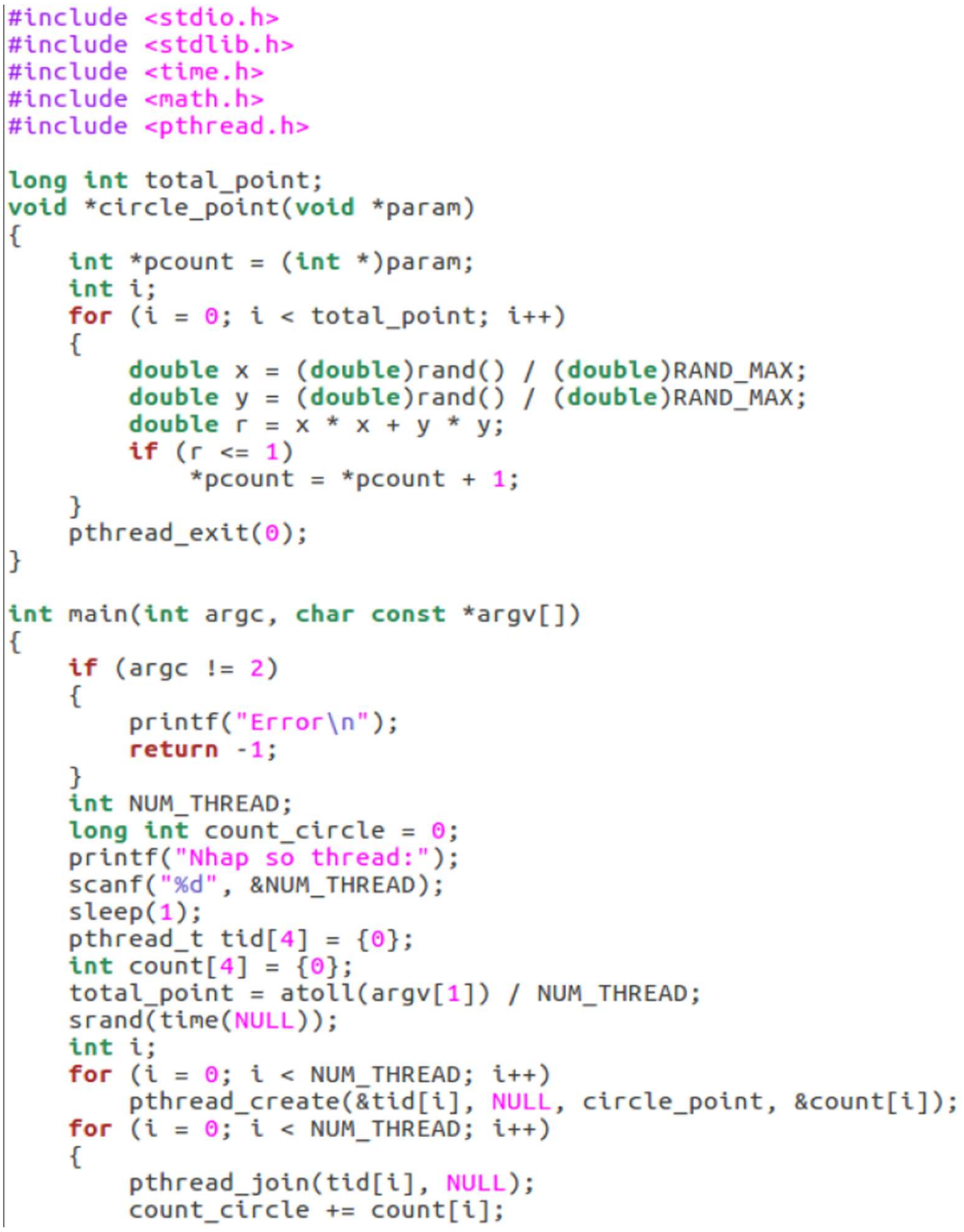


```
#include <stdio.h>
#include <pthread.h>
#include <semaphore.h>
#include <unistd.h>

sem_t mutex1,mutex2;

void *inle(void *arg)
{
    // wait
    int i;
    for(i=1;i<12;i+=2){
        sem_wait(&mutex2);
        printf("Thred 1:%d\n",i);
        sem_post(&mutex1);
    }

}

void *inchan(void *arg)
{
    // wait
    int i;
    for(i=2;i<11;i+=2){
        sem_wait(&mutex1);
        printf("Thred 2:%d\n",i);
        sem_post(&mutex2);
    }

}

int main()
{
    sem_init(&mutex1, 0, 0);
    sem_init(&mutex2, 0, 1);

    pthread_t t1, t2;
    pthread_create(&t1, NULL, inle, NULL);
    pthread_create(&t2, NULL, inchan, NULL);
    pthread_join(t1, NULL);
    pthread_join(t2, NULL);
    sem_destroy(&mutex1);
sem_destroy(&mutex2);
    return 0;
}
```

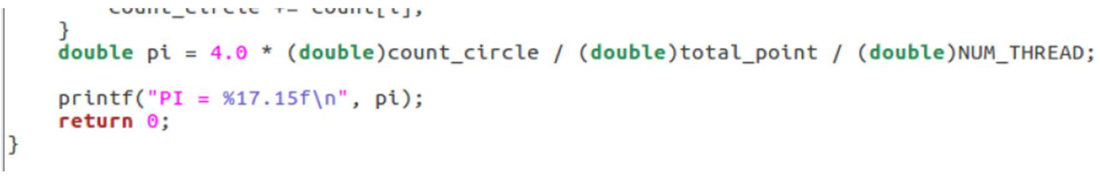


```
trongdat1108@ubuntu:~/lab 8.1$ gcc -c Cau1.c
trongdat1108@ubuntu:~/lab 8.1$ gcc -o Cau1.out Cau1.o -lpthread
trongdat1108@ubuntu:~/lab 8.1$ ./Cau1.out
Thred 1:1
Thred 2:2
Thred 1:3
Thred 2:4
Thred 1:5
Thred 2:6
Thred 1:7
Thred 2:8
Thred 1:9
Thred 2:10
Thred 1:11
```

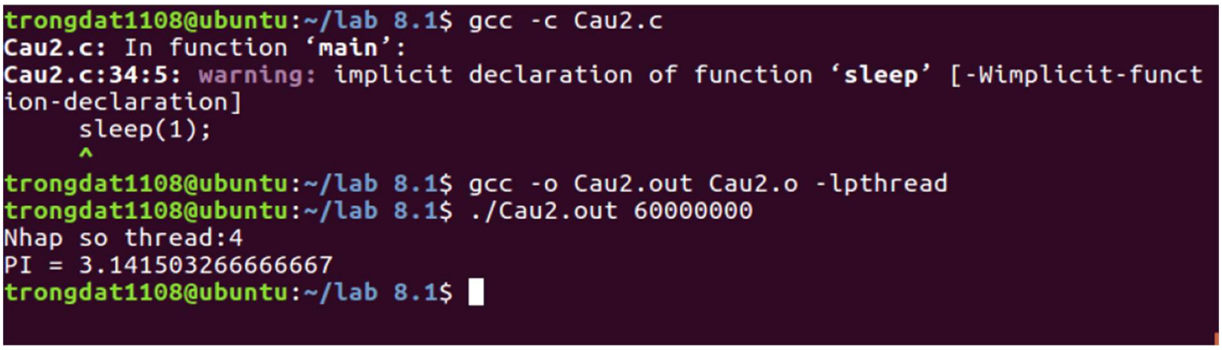
Câu 2:



```c
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
#include <pthread.h>

long int total_point;
void *circle_point(void *param)
{
    int *pcount = (int *)param;
    int i;
    for (i = 0; i < total_point; i++)
    {
        double x = (double)rand() / (double)RAND_MAX;
        double y = (double)rand() / (double)RAND_MAX;
        double r = x * x + y * y;
        if (r <= 1)
            *pcount = *pcount + 1;
    }
    pthread_exit(0);
}

int main(int argc, char const *argv[])
{
    if (argc != 2)
    {
        printf("Error\n");
        return -1;
    }
    int NUM_THREAD;
    long int count_circle = 0;
    printf("Nhap so thread:");
    scanf("%d", &NUM_THREAD);
    sleep(1);
    pthread_t tid[4] = {0};
    int count[4] = {0};
    total_point = atoll(argv[1]) / NUM_THREAD;
    srand(time(NULL));
    int i;
    for (i = 0; i < NUM_THREAD; i++)
        pthread_create(&tid[i], NULL, circle_point, &count[i]);
    for (i = 0; i < NUM_THREAD; i++)
    {
        pthread_join(tid[i], NULL);
        count_circle += count[i];
```

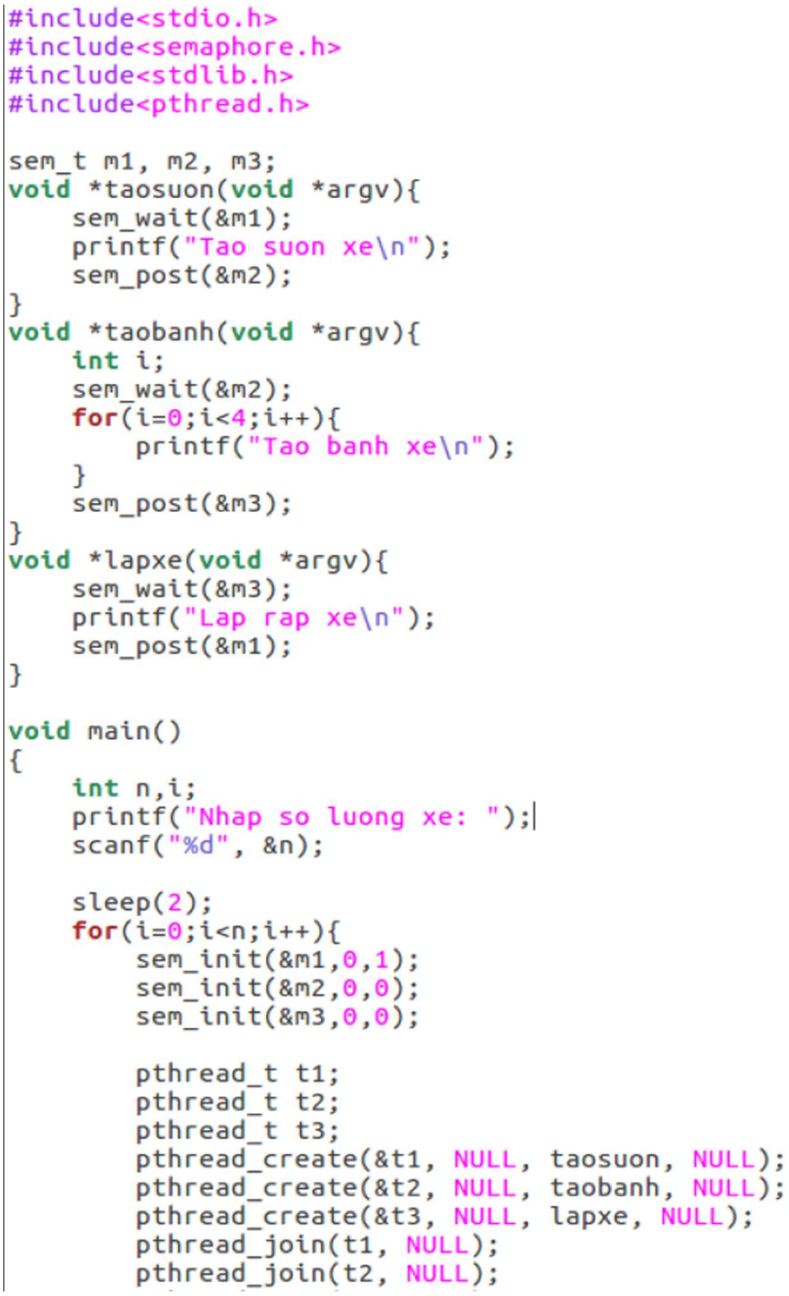


```
        count_circle += count[i];
    }
    double pi = 4.0 * (double)count_circle / (double)total_point / (double)NUM_THREAD;

    printf("PI = %17.15f\n", pi);
    return 0;
}
```

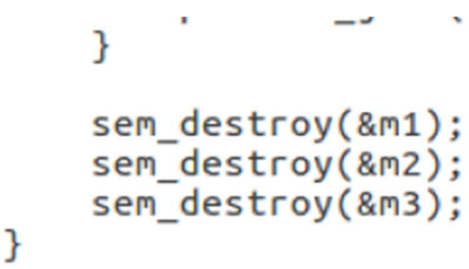


```
trongdat1108@ubuntu:~/lab 8.1$ gcc -c Cau2.c
Cau2.c: In function ‘main’:
Cau2.c:34:5: warning: implicit declaration of function ‘sleep’ [-Wimplicit-funct
ion-declaration]
     sleep(1);
     ^
trongdat1108@ubuntu:~/lab 8.1$ gcc -o Cau2.out Cau2.o -lpthread
trongdat1108@ubuntu:~/lab 8.1$ ./Cau2.out 60000000
Nhap so thread:4
PI = 3.141503266666667
trongdat1108@ubuntu:~/lab 8.1$ 
```

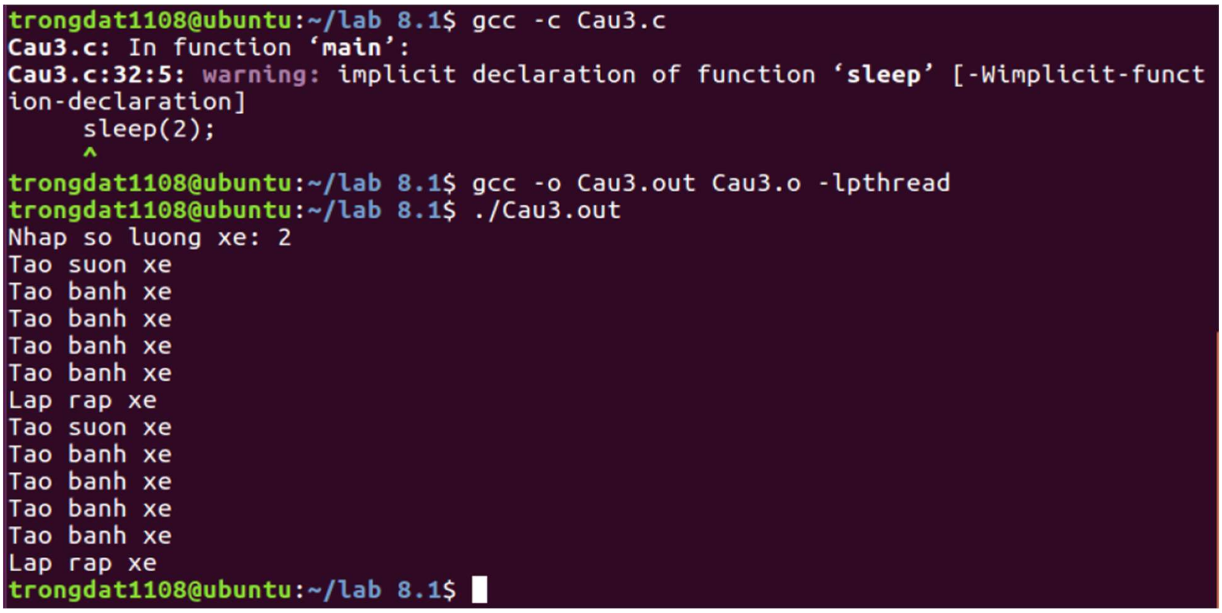
Câu 3:



```
#include<stdio.h>
#include<semaphore.h>
#include<stdlib.h>
#include<pthread.h>

sem_t m1, m2, m3;
void *taosuon(void *argv){
    sem_wait(&m1);
    printf("Tao suon xe\n");
    sem_post(&m2);
}
void *taobanh(void *argv){
    int i;
    sem_wait(&m2);
    for(i=0;i<4;i++){
        printf("Tao banh xe\n");
    }
    sem_post(&m3);
}
void *lapxe(void *argv){
    sem_wait(&m3);
    printf("Lap rap xe\n");
    sem_post(&m1);
}

void main()
{
    int n,i;
    printf("Nhap so luong xe: ");
    scanf("%d", &n);

    sleep(2);
    for(i=0;i<n;i++){
        sem_init(&m1,0,1);
        sem_init(&m2,0,0);
        sem_init(&m3,0,0);

        pthread_t t1;
        pthread_t t2;
        pthread_t t3;
        pthread_create(&t1, NULL, taosuon, NULL);
        pthread_create(&t2, NULL, taobanh, NULL);
        pthread_create(&t3, NULL, lapxe, NULL);
        pthread_join(t1, NULL);
        pthread_join(t2, NULL);
```



```
    }

    sem_destroy(&m1);
    sem_destroy(&m2);
    sem_destroy(&m3);
}
```

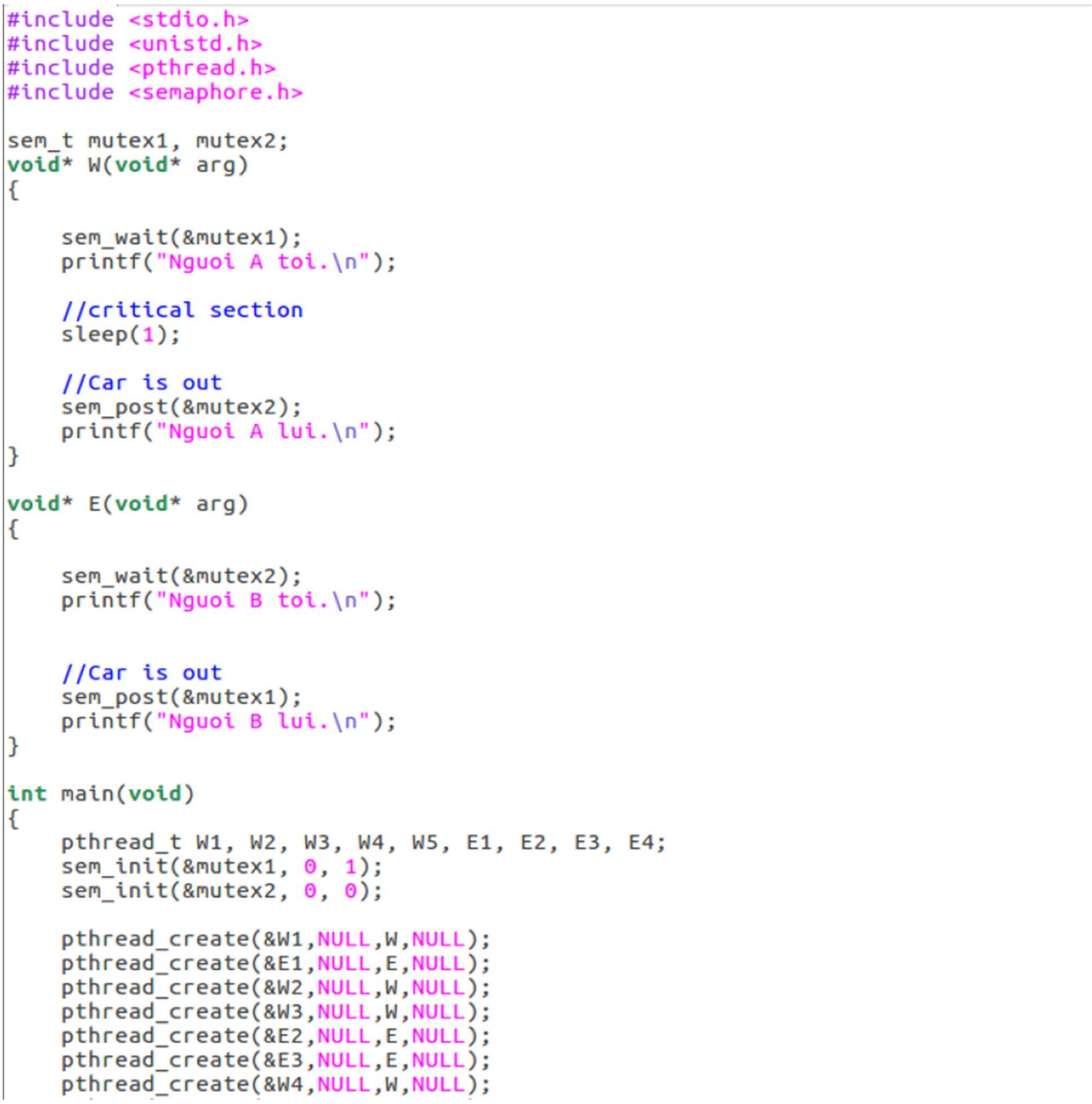


```
trongdat1108@ubuntu:~/lab 8.1$ gcc -c Cau3.c
Cau3.c: In function ‘main’:
Cau3.c:32:5: warning: implicit declaration of function ‘sleep’ [-Wimplicit-funct
ion-declaration]
     sleep(2);
     ^
trongdat1108@ubuntu:~/lab 8.1$ gcc -o Cau3.out Cau3.o -lpthread
trongdat1108@ubuntu:~/lab 8.1$ ./Cau3.out
Nhap so luong xe: 2
Tao suon xe
Tao banh xe
Tao banh xe
Tao banh xe
Tao banh xe
Lap rap xe
Tao suon xe
Tao banh xe
Tao banh xe
Tao banh xe
Tao banh xe
Lap rap xe
trongdat1108@ubuntu:~/lab 8.1$ 
```

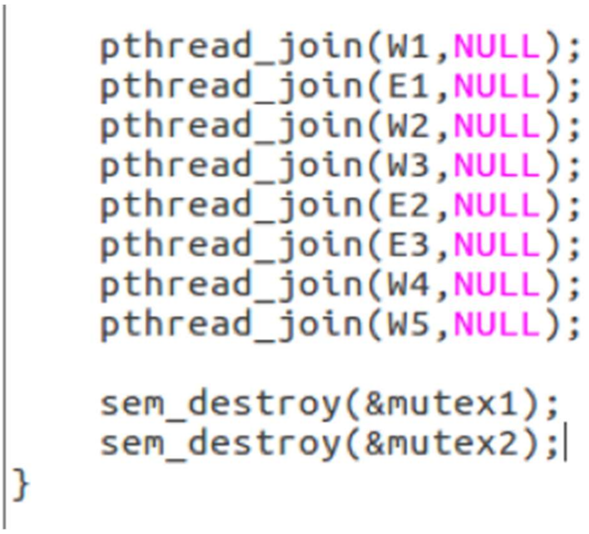
Bài tập thêm:

Lab 8.2:

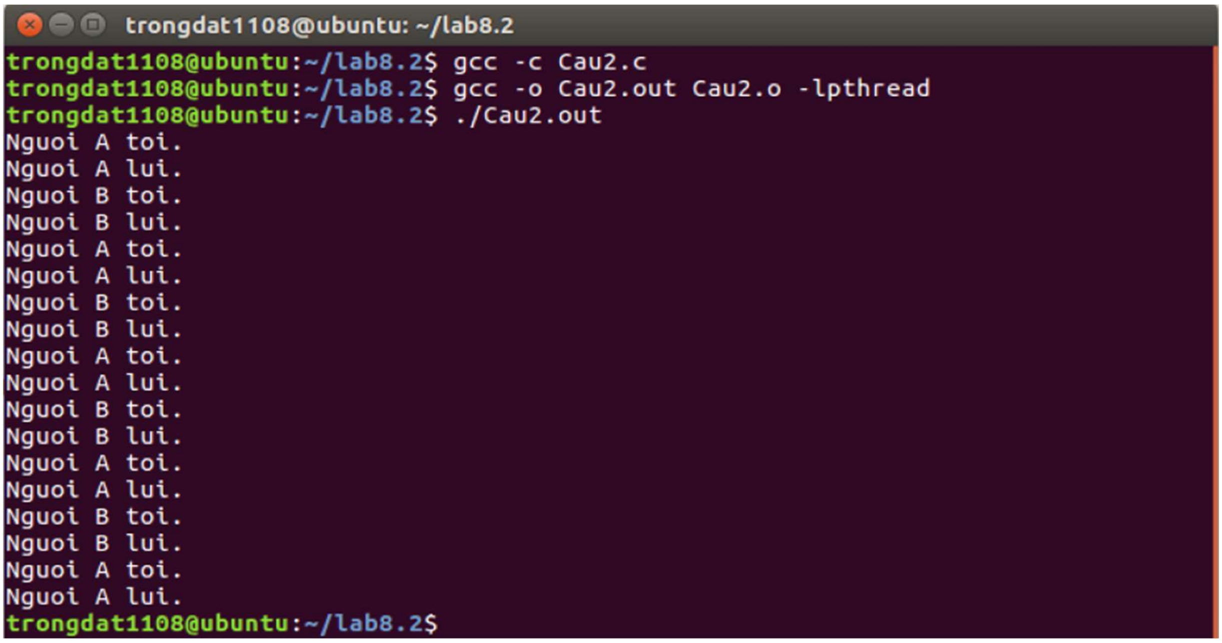
Câu 2:



```
#include <stdio.h>
#include <unistd.h>
#include <pthread.h>
#include <semaphore.h>

sem_t mutex1, mutex2;
void* W(void* arg)
{

    sem_wait(&mutex1);
    printf("Nguoi A toi.\n");

    //critical section
    sleep(1);

    //Car is out
    sem_post(&mutex2);
    printf("Nguoi A lui.\n");
}

void* E(void* arg)
{

    sem_wait(&mutex2);
    printf("Nguoi B toi.\n");


    //Car is out
    sem_post(&mutex1);
    printf("Nguoi B lui.\n");
}

int main(void)
{
    pthread_t W1, W2, W3, W4, W5, E1, E2, E3, E4;
    sem_init(&mutex1, 0, 1);
    sem_init(&mutex2, 0, 0);

    pthread_create(&W1,NULL,W,NULL);
    pthread_create(&E1,NULL,E,NULL);
    pthread_create(&W2,NULL,W,NULL);
    pthread_create(&W3,NULL,W,NULL);
    pthread_create(&E2,NULL,E,NULL);
    pthread_create(&E3,NULL,E,NULL);
    pthread_create(&W4,NULL,W,NULL);
```

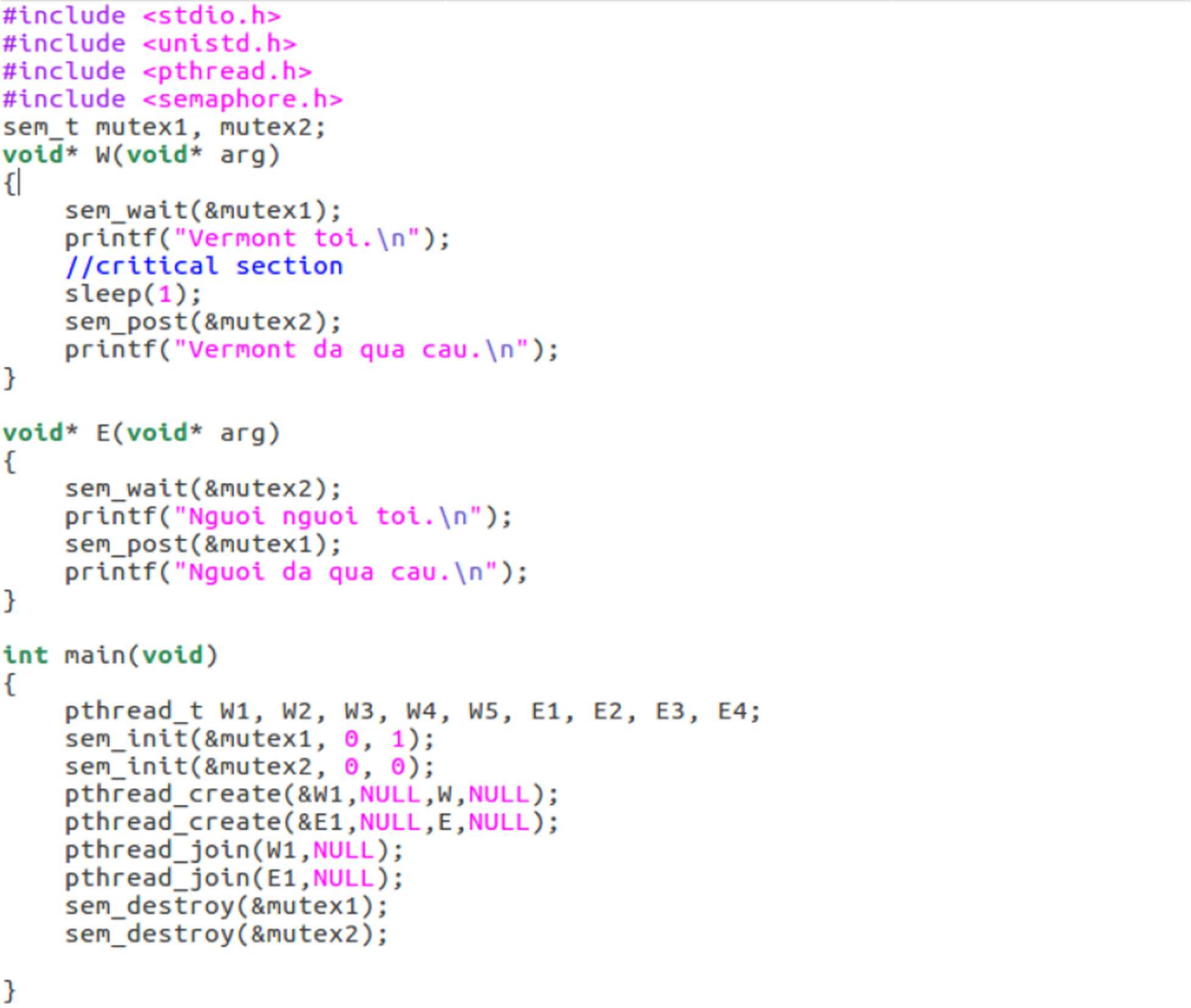


```
    pthread_join(W1,NULL);
    pthread_join(E1,NULL);
    pthread_join(W2,NULL);
    pthread_join(W3,NULL);
    pthread_join(E2,NULL);
    pthread_join(E3,NULL);
    pthread_join(W4,NULL);
    pthread_join(W5,NULL);

    sem_destroy(&mutex1);
    sem_destroy(&mutex2);
}
```

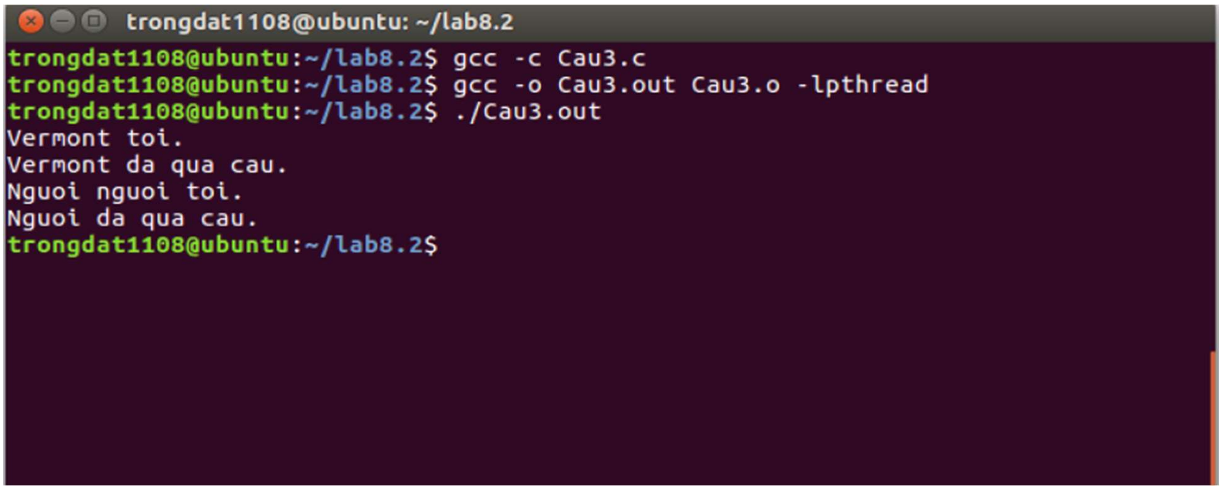


```
trongdat1108@ubuntu:~/lab8.2$ gcc -c Cau2.c
trongdat1108@ubuntu:~/lab8.2$ gcc -o Cau2.out Cau2.o -lpthread
trongdat1108@ubuntu:~/lab8.2$ ./Cau2.out
Nguoi A toi.
Nguoi A lui.
Nguoi B toi.
Nguoi B lui.
Nguoi A toi.
Nguoi A lui.
Nguoi B toi.
Nguoi B lui.
Nguoi A toi.
Nguoi A lui.
Nguoi B toi.
Nguoi B lui.
Nguoi A toi.
Nguoi A lui.
Nguoi B toi.
Nguoi B lui.
Nguoi A toi.
Nguoi A lui.
trongdat1108@ubuntu:~/lab8.2$
```

Câu 3:



```c
#include <stdio.h>
#include <unistd.h>
#include <pthread.h>
#include <semaphore.h>
sem_t mutex1, mutex2;
void* W(void* arg)
{
    sem_wait(&mutex1);
    printf("Vermont toi.\n");
    //critical section
    sleep(1);
    sem_post(&mutex2);
    printf("Vermont da qua cau.\n");
}

void* E(void* arg)
{
    sem_wait(&mutex2);
    printf("Nguoi nguoi toi.\n");
    sem_post(&mutex1);
    printf("Nguoi da qua cau.\n");
}

int main(void)
{
    pthread_t W1, W2, W3, W4, W5, E1, E2, E3, E4;
    sem_init(&mutex1, 0, 1);
    sem_init(&mutex2, 0, 0);
    pthread_create(&W1,NULL,W,NULL);
    pthread_create(&E1,NULL,E,NULL);
    pthread_join(W1,NULL);
    pthread_join(E1,NULL);
    sem_destroy(&mutex1);
    sem_destroy(&mutex2);

}
```



```
trongdat1108@ubuntu:~/lab8.2$ gcc -c Cau3.c
trongdat1108@ubuntu:~/lab8.2$ gcc -o Cau3.out Cau3.o -lpthread
trongdat1108@ubuntu:~/lab8.2$ ./Cau3.out
Vermont toi.
Vermont da qua cau.
Nguoi nguoi toi.
Nguoi da qua cau.
trongdat1108@ubuntu:~/lab8.2$
```