

Tên	Đinh Phương My
MSSV	52100703
Nhóm thực hành	N101

BÀI BÁO CÁO THỰC HÀNH LAB 8.2

Câu 1: Hãy thực hiện đồng bộ bằng semaphore, so sánh độ chính xác của số PI sau đó. Đồng thời đo thời gian chạy và cho biết sau khi đồng bộ, thời gian chạy đã tăng bao nhiêu %. Có thể cải tiến chương trình để giảm thiểu sự trả giá % thời gian chạy này không?

Code:

```

ex1.c (-/Lab8/Lab8.2) - gedit
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <string.h>
#include <pthread.h>
#include <semaphore.h>
#define SEED 35791246

sem_t mutex1, mutex2;

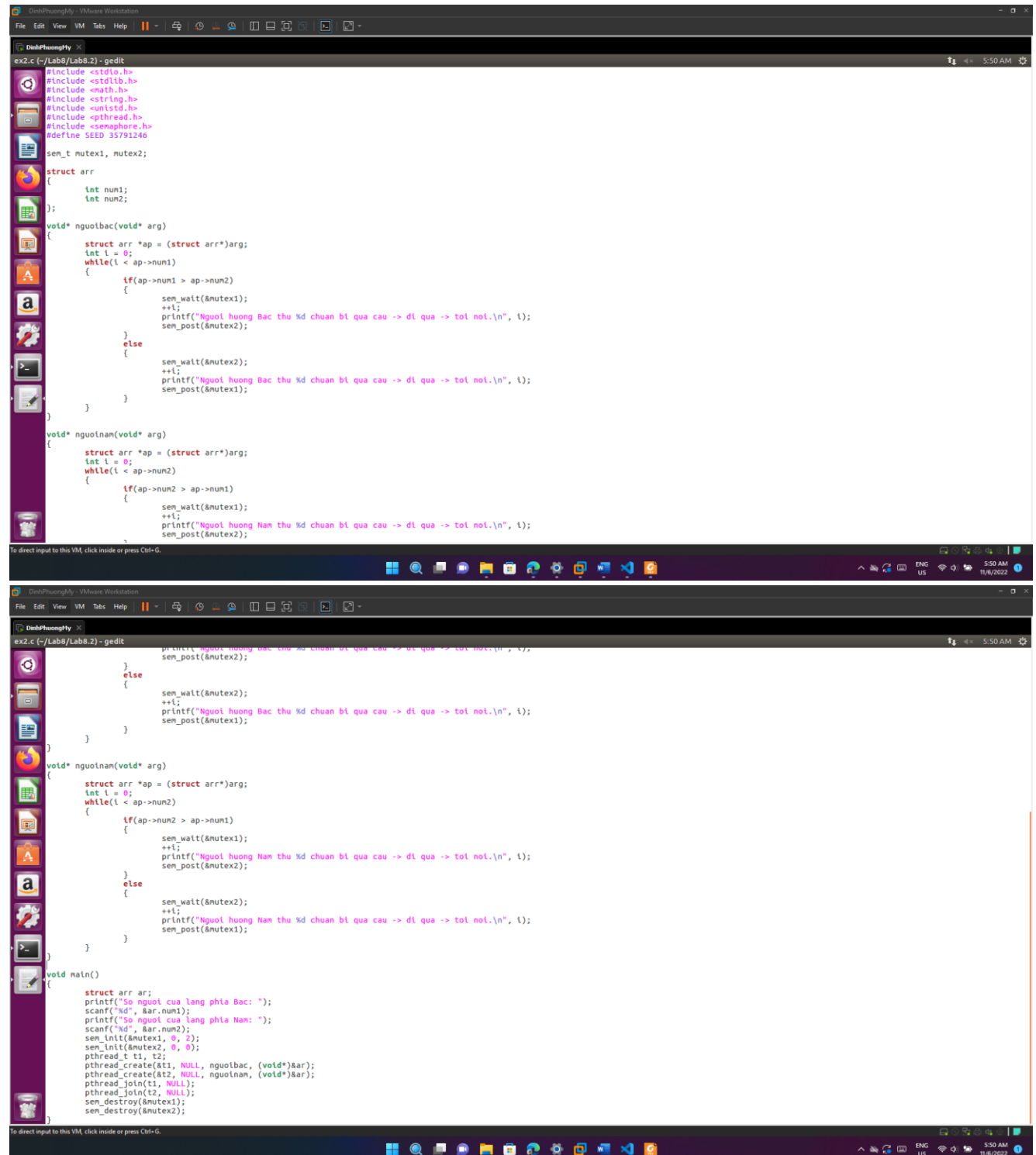
struct arr
{
    int count;
    int niter;
};

void* count1(void* arg)
{
    struct arr *ap = (struct arr*)arg;
    int i;
    double x, y, z;
    for(i = 0; i < (ap->niter)/2; i++)
    {
        sem_wait(&mutex1);
        x = (double) rand()/RAND_MAX;
        y = (double) rand()/RAND_MAX;
        z = x*x + y*y;
        if(z <= 1)
        {
            ap->count += 1;
        }
        sem_post(&mutex2);
    }
}

void* count2(void* arg)
{
    struct arr *ap = (struct arr*)arg;
    int i = 0;
    double x, y, z;
    for(i = (ap->niter)/2; i < ap->niter; i++)
    {
        sem_wait(&mutex2);
        x = (double) rand()/RAND_MAX;
        y = (double) rand()/RAND_MAX;
        z = x*x + y*y;
        if(z <= 1)
        {
            ap->count += 1;
        }
        sem_post(&mutex1);
    }
}
  
```


Câu 2: Xem xét bài toán “Nông dân qua cầu Vermont”, (tạm gọi bài tập này là phiên bản 2), chúng ta không kiểm soát tải trọng, mà giả định rằng người nông dân đến từ một trong hai hướng: Bắc và Nam, và không thể có 2 người cùng hướng đi lên cầu. Hãy thiết kế semaphore để giải quyết vấn đề này.

Code:



```
ex2.c (~\Lab8\Lab8.2) - gedit
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <string.h>
#include <unistd.h>
#include <pthread.h>
#define SEED 35791246

sem_t mutex1, mutex2;

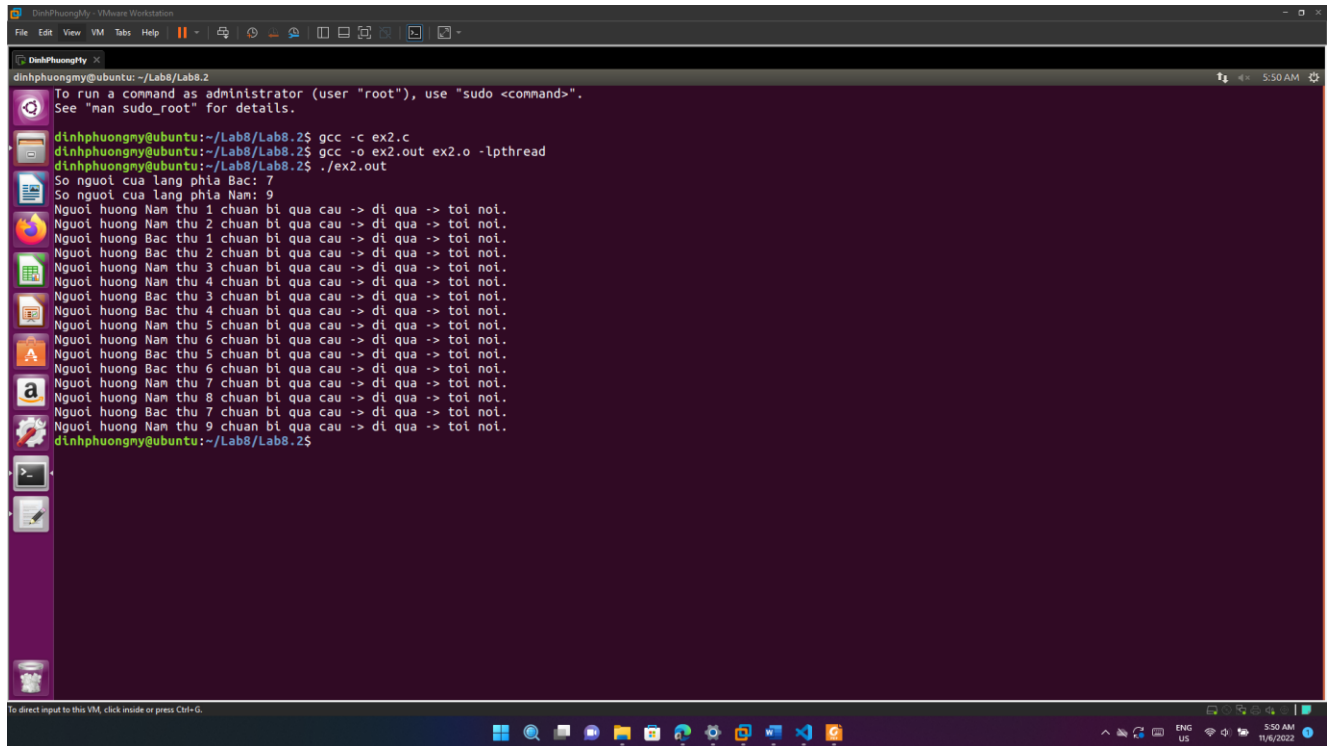
struct arr
{
    int num1;
    int num2;
};

void* nguoibac(void* arg)
{
    struct arr *ap = (struct arr*)arg;
    int l = 0;
    while(l < ap->num1)
    {
        if(ap->num1 > ap->num2)
        {
            sem_wait(&mutex1);
            ++l;
            printf("Nguoi huong Bac thu %d chuan bi qua cau -> di qua -> toi noi.\n", l);
            sem_post(&mutex2);
        }
        else
        {
            sem_wait(&mutex2);
            ++l;
            printf("Nguoi huong Bac thu %d chuan bi qua cau -> di qua -> toi noi.\n", l);
            sem_post(&mutex1);
        }
    }
}

void* nguoinam(void* arg)
{
    struct arr *ap = (struct arr*)arg;
    int l = 0;
    while(l < ap->num2)
    {
        if(ap->num2 > ap->num1)
        {
            sem_wait(&mutex1);
            ++l;
            printf("Nguoi huong Nam thu %d chuan bi qua cau -> di qua -> toi noi.\n", l);
            sem_post(&mutex2);
        }
        else
        {
            sem_wait(&mutex2);
            ++l;
            printf("Nguoi huong Nam thu %d chuan bi qua cau -> di qua -> toi noi.\n", l);
            sem_post(&mutex1);
        }
    }
}

void main()
{
    struct arr ar;
    printf("So nguoi cua lang phia Bac: ");
    scanf("%d", &ar.num1);
    printf("So nguoi cua lang phia Nam: ");
    scanf("%d", &ar.num2);
    sem_init(&mutex1, 0, 2);
    sem_init(&mutex2, 0, 0);
    pthread_t t1, t2;
    pthread_create(&t1, NULL, nguoibac, (void*)&ar);
    pthread_create(&t2, NULL, nguoinam, (void*)&ar);
    pthread_join(t1, NULL);
    pthread_join(t2, NULL);
    sem_destroy(&mutex1);
    sem_destroy(&mutex2);
}
```

Run:



```
dinhphuongmy@ubuntu: ~/Lab8/Lab8.2
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

dinhphuongmy@ubuntu:~/Lab8/Lab8.2$ gcc -c ex2.c
dinhphuongmy@ubuntu:~/Lab8/Lab8.2$ gcc -o ex2.out ex2.o -lpthread
dinhphuongmy@ubuntu:~/Lab8/Lab8.2$ ./ex2.out
So nguoi cua lang phia Bac: 7
So nguoi cua lang phia Nam: 9
Nguoi huong Nam thu 1 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Nam thu 2 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Bac thu 1 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Bac thu 2 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Nam thu 3 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Nam thu 4 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Bac thu 3 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Bac thu 4 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Nam thu 5 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Nam thu 6 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Bac thu 5 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Bac thu 6 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Nam thu 7 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Nam thu 8 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Bac thu 7 chuan bi qua cau -> di qua -> toi noi.
Nguoi huong Nam thu 9 chuan bi qua cau -> di qua -> toi noi.
dinhphuongmy@ubuntu:~/Lab8/Lab8.2$
```