



Instituto Politécnico Nacional Escuela Superior de Cómputo



Sistemas Operativo Tarea

Semáforos EN LENGUAJE C

ENFOCADOS A LINUX

Integrantes:

Morales Blas David Israel

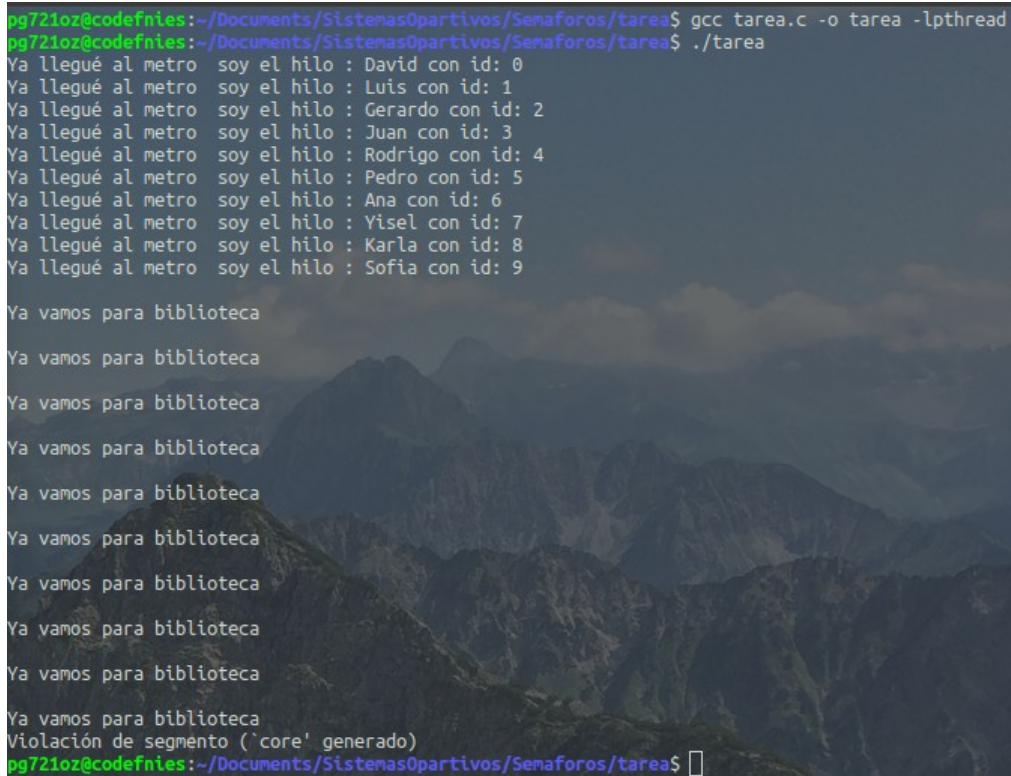
Profesora:

JUÁREZ MÉNDEZ ANA BELEM

Grupo: 2CM8

Fecha de entrega: 31 de marzo del 2020

Fotos de la práctica:



```
pg721oz@codefnies:~/Documents/Sistemas0partivos/Semaforos/tarea$ gcc tarea.c -o tarea -lpthread
pg721oz@codefnies:~/Documents/Sistemas0partivos/Semaforos/tarea$ ./tarea
Ya llegué al metro soy el hilo : David con id: 0
Ya llegué al metro soy el hilo : Luis con id: 1
Ya llegué al metro soy el hilo : Gerardo con id: 2
Ya llegué al metro soy el hilo : Juan con id: 3
Ya llegué al metro soy el hilo : Rodrigo con id: 4
Ya llegué al metro soy el hilo : Pedro con id: 5
Ya llegué al metro soy el hilo : Ana con id: 6
Ya llegué al metro soy el hilo : Yisel con id: 7
Ya llegué al metro soy el hilo : Karla con id: 8
Ya llegué al metro soy el hilo : Sofia con id: 9

Ya vamos para biblioteca
Ya vamos para biblioteca
Ya vamos para biblioteca
Ya vamos para biblioteca
Ya vamos para biblioteca
Ya vamos para biblioteca
Ya vamos para biblioteca
Ya vamos para biblioteca
Ya vamos para biblioteca
Ya vamos para biblioteca
Violación de segmento ('core' generado)
pg721oz@codefnies:~/Documents/Sistemas0partivos/Semaforos/tarea$
```

Código:

```

#include <stdio.h>
#include <pthread.h>
#include <unistd.h>
#include <semaphore.h>
#include <stdlib.h>
#define NHILOS 10
#define NP 10
int counter = 0;
void rutinahilo(void *);
void sincronizarProceso();

typedef struct data{
    char *cadena;
    int id;
}parametro;

sem_t semaforo;
sem_t barrera;

int main(void){
    int i, id[NHILOS], status, *salida;
    char *nombres[] = {"David", "Luis", "Gerardo", "Juan", "Rodrigo", "Pedro", "Ana", "Yisel", "Karla", "Sofia"};
    pthread_t hilos[NHILOS];
    sem_init(&semaforo, 0, 1);
    sem_init(&barrera, 0, 0);
    parametro p[NHILOS];

    for(i=0; i<NHILOS; i++){
        p[i].id = i;
        p[i].cadena = nombres[i];
    }
    for(i=0; i<NHILOS; i++){
        id[i]=i;
        status=pthread_create(&hilos[i], NULL, (void *) rutinahilo, &p[i]);
        sleep(1);
        if(status){
            printf("Error en el hilo %i \n", status);
            exit(-1);
        }
    }
    for(i = 0; i< NHILOS; i++){
        pthread_join(hilos[i], (void *) &salida);
        printf("\n Hilo %i termino con código %i \n", i, *salida);
    }
    return 0;
}

void rutinahilo(void *args){
    parametro *par = (parametro *) args;
    //printf("Ya llegué al metro soy el hilo : %s con id: %i\n", *(int *)args);
    printf("Ya llegué al metro soy el hilo : %s con id: %i\n", par->cadena, par->id);
    sleep(1);
    sincronizarProceso();
    printf("\nYa vamos para biblioteca\n");
    pthread_exit(0);
}

void sincronizarProceso(){
    int i;
    sem_wait(&semaforo);
    counter+=1;
    if(counter<NP){
        sem_post(&semaforo);
        sem_wait(&barrera);
    }else{
        for(i=0; i<NP-1; i++ )
            sem_post(&barrera);
        counter=0;
        sem_post(&semaforo);
    }
}

```