**מעבדה 8 - סמפורים**

**מגישים:** 1. חוטמליאנסקי דמיטרי 334017415

2. שיח אחמד מוחמד 209158120

תרגיל 1

#include <pthread.h>

#include <stdio.h>

#include <string.h>

#include <semaphore.h>

sem\_t sem1; //semaphore declaration

sem\_t sem2;

void\* create\_message(void\* str){

int i = 0;

for(i = 0; i < 10; i++){

sem\_wait(&sem1); //block thread and wait

printf("I've wrote a message #%d. %s\n", i+1, (char\*)str);

sem\_post(&sem2); //wake up thread

}

}

int main(){

pthread\_t thread;

sem\_init(&sem1, 0, 1); //semaphore initialization

sem\_init(&sem2, 0, 0);

int i = 0;

pthread\_create(&thread, NULL, create\_message, (void\*)"Thread A");

for(i = 0; i < 10; i++){

sem\_wait(&sem2); //block thread and wait

printf("The message #%d was printed. Thread B \n", i+1);

sem\_post(&sem1); //wake up thread

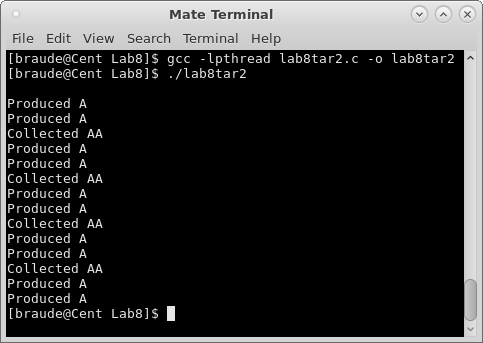
}

pthread\_join(thread, NULL);

sem\_destroy(&sem1); //destroy the semaphore

sem\_destroy(&sem2);

}



תרגיל 2

#include <stdlib.h>

#include <stdio.h>

#include <unistd.h>

#include <pthread.h>

#include <semaphore.h>

sem\_t sem; //semaphore declaration

void\* producer(); //function producing A product

void\* collector(); //function collect AA and produce new product

int main(){

int ans[2];

pthread\_t th1, th2; //two threads for each function

sem\_init(&sem, 0, 0); //initializing the semaphore

ans[0] = pthread\_create(&th1, NULL, producer, NULL); //first thread

if(ans[0] != 0){ //check if thread creating was successful

printf("Creating thread 1 ERROR");

exit(1);

}

ans[1] = pthread\_create(&th1, NULL, collector, NULL); //second thread

if(ans[1] != 0){ //check if thread creating was successful

printf("Creating thread 2 ERROR");

exit(1);

}

sleep(20); //the program will close after 20 seconds

sem\_destroy(&sem); //destroy the semaphore

return 0;

}

//\*\*\*\*\*\*\*Produce A product\*\*\*\*\*\*\*\*

void\* producer(){

while(1){

sem\_post(&sem); //wake up the thread for producing

printf("\nProduced A");

sleep(2); //wait 2 seconds

}

}

//\*\*\*\*\*\*\*wait for producing 2 A products and create new AA product\*\*\*\*\*\*\*\*

void\* collector(){

while(1){

sem\_wait(&sem); //wait for product A

sem\_wait(&sem); //wait for product A

printf("\nCollected AA");

}

}

