**מעבדה 9 - סמפורים – בעיות נוספות**

To initialize a semaphore, use [sem\_init](http://pubs.opengroup.org/onlinepubs/009695399/functions/sem_init.html):

int [sem\_init](http://pubs.opengroup.org/onlinepubs/009695399/functions/sem_init.html)(sem\_t \*sem, int pshared, unsigned int value);

* sem points to a semaphore object to initialize
* pshared is a flag indicating whether or not the semaphore should be shared with fork()ed processes. In Linux only 0;
* value is an initial value to set the semaphore to

Example of use:  sem\_init(&sem\_name, 0, 10);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

To wait on a semaphore, use [sem\_wait](http://pubs.opengroup.org/onlinepubs/009695399/functions/sem_wait.html):

int [sem\_wait](http://pubs.opengroup.org/onlinepubs/009695399/functions/sem_wait.html)(sem\_t \*sem);

Example of use:  sem\_wait(&sem\_name);

* If the value of the semaphore isn't positive, the calling process blocks; one of the blocked processes wakes up when another process calls sem\_post.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

To increment the value of a semaphore, use [sem\_post](http://pubs.opengroup.org/onlinepubs/009695399/functions/sem_post.html):

int [sem\_post](http://pubs.opengroup.org/onlinepubs/009695399/functions/sem_post.html)(sem\_t \*sem);

Example of use:

sem\_post(&sem\_name);

* It increments the value of the semaphore and wakes up a blocked process waiting on the semaphore, if any.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**תרגיל 1.**

עליכם לממש קוד למחסנית משותפת ל K חוטים.

כלומר, עליכם להגדיר מערך גלובלי: int stk[N] לייצוג המחסנית ואינדקס int idx ועליכם לכתוב את הפונקציות:

void stkPush(int num);

int stkPop();

שימו לב: חוט המבצע POP ממחסנית ריקה, יחכה שיהיה מה להוציא.

חוט המבצע PUSH למחסנית מלאה, יחכה עד שיתבצע POP ויהיה מקום

ה main יהיה:

#define K 5

#define N 10

void stkPush(int num);

int stkPop();

...//ניתן להוסיף קוד כרצונך

int main(int argc, char \* argv[]){

pthread\_t tArr[K];

int i=0 , ans[K];

...//ניתן להוסיף קוד כרצונך

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

ans[i]=pthread\_create(&tArr[i], NULL, tFunc, NULL);

sleep(10);

return 0;

}

void\* tFunc(void\* p) {

int num;

while(1){

num = rand()%100;

if(rand()%2) {

stkPush(num);

printf(“Push %d\n”, num);

} else {

printf(“Pop %d\n”, stkPop());

}

sleep(1);

}

}

**תרגיל 2**

בתרגיל זה נכתוב תכנית לניהול N אולמות עם מסכים פרטיים להקרנת סרט בקניון (N יוגדר בתוכנית כרצונכם ע"י define).

**בכל אולם יש מקום ל 5 אנשים.** (עם מסכים להקרנה פרטית :)

התנאי לפתיחת אולם הקרנה הוא שלפחות שני אנשים מעוניינים להיכנס לאולם ושאין אולם אחר שמאויש חלקית. הסרט יוקרן במסכים אישיים. כל איש יכול להיכנס לאולם ולצפות בסרט במסך אישי.

האנשים יכנסו לאולמות לפי סדר הגעתם. מי שרוצה להיכנס לאולם ולראות סרט יפעיל את הפונקציה:

void \* watchMovie (void \* p)

הפונקציה מדפיסה:

Person #i want to watch movie

הפונקציה תדאג להכניס את האיש לאולם פנוי על פי החוקים הנ"ל. לאחר כניסת איש לאולם, יודפס:

Person #i watch movie in room j

לאחר מכן הפונקציה תבצע sleep(2) לדמות צפייה בסרט ולבסוף יודפס:

Person #i END watch movie

כתשובה לשאלה זאת יש לכתוב תכנית מתועדות בשפת C.

חובה להשתמש בסמפורים. נדרש להוסיף קוד לאתחול הסמפורים ב main. ב main נדרש להפעיל 20 חוטים לדמות אנשים בקניון שמעוניינים לראות סרט במסך פרטי. לכל איש מספר סידורי. כל איש (חוט) יפעיל את הפונקציה watchMovie .

N יוגדרו כרצונכם. על התוכנית להיות נכונה לכל N.

לדוגמא, תחילת פלט אפשרי:

Person #0 wants to watch movie

Person #1 wants to watch movie

Person #0 watch movie in room 1

Person #1 watch movie in room 1

Person #2 wants to watch movie

Person #2 watch movie in room 1

Person #3 wants to watch movie

….