

מערכות הפעלה 31261

מבחן סופי, מועד ב', סמסטר ב' תשע"ג, 06/08/2013

הוראות לנבחן: משך הבחינה שלוש שעות. חומרי העזר המותרים הם השקפים של הקורס שנמסרו באתר הקורס ומחשבוני פשוטים בלבד. כל חומר נוסף מעבר למותר ייפסול את הבחינה! רשום את תשובותיך במחברת במצורפת לבחינה. ציין בבירור את מספר השאלה במחברת והשתדל להיות קצר וענייני (תשובות ארוכות מדי או נסיונות לתת מספר תשובות אפשריות ייפסלו את תשובתך!). הקפד על כתב יד ברור ומסודר, ומחק את כל חומר הטייטא. השימוש בשפה האנגלית מותר. השאלון מכיל 9 שאלות בשווי של 100 נקודות, שאלת בונוס בשווי 10 נקודות, ונפרש על פני 4 עמודים.
בהצלחה!

שאלה 1 [10%]

- תאר בקצרה את מנגנון הפסיקה (Interrupt) במערכת הפעלה מודרנית (באופן כללי)
- תן שלוש דוגמאות לשלושה סוגי פסיקות נפוצות

שאלה 2 [10%]

- תאר בקצרה מהן קריאות שרות (system calls) במערכת הפעלה מודרנית?
- תן שלוש דוגמאות לקריאות שרות במערכת ההפעלה Unix והסבר את תפקידיהן.
- תן שלוש דוגמאות לקריאות שרות במערכת ההפעלה Windows והסבר את תפקידיהן.

שאלה 3 [5%]

הסבר בקצרה מהן תוכניות מערכת (system programs) ותן שלוש דוגמאות לתוכניות מסוג זה במערכת ההפעלה Unix, ושלוש דוגמאות במערכת ההפעלה Windows.

שאלה 4 [12%]

- לפניך דיאגרמת חלקית של רשומת ה-PCB (Process Control Block) של תהליך במערכת הפעלה.
- תן הסבר קצר לכל אחד מהשדות המופיעים בדיאגרמה
 - תן דוגמא אחת בלבד לשדה משמעותי נוסף שאינו מופיע בדיאגרמה

process state
process number
program counter
registers
memory limits
list of open files
...

שאלה 5 [8%]

בתהליך P במערכת ההפעלה Linux :

- יש 200 חוטים (Threads) שונים A[0:200] אשר כל אחד מהם מבלה 30% מזמנו ביחידת העיבוד, ו-70% מזמנו בפעולות קלט ופלט שונות.
- 48 חוטים נוספים B[0:48] אשר כל אחד מהם מבלה 40% מזמנו ביחידת העיבוד, ו-60% בפעולות קלט/פלט.
- אף אחד מהחוטים A[100:124] לא יוכל להתחיל לפני שכל החוטים B[0:24] סיימו. כל שאר החוטים הם עצמאיים לחלוטין.
- במעבד שלנו יש 8 ליבות בעלות כוח חישוב זהה לחלוטין
- הזמן הריצה של כל אחד מהחוטים A[0:200] הוא 50 שניות, ושל כל חוט B[0:48] הוא 100 שניות

בהנחה שמערכת ההפעלה היא הוגנת (Fair), ומריצה רק את התהליך P (ולא שום תהליך נוסף) בתנאים האידאליים ביותר, מהו הזמן הקצר ביותר שבו ניתן לצפות שהתהליך P יסיים את עבודתו?

שאלה 6 [12%]

א. הסבר בקצרה את מה שקורה בתוכנית C הבאה במערכת ההפעלה Unix.

ב. מה יהיה הפלט המדויק של התוכנית?

ג. האם תשובתך לסעיף הקודם חייבת להיות יחידה או האם ייתכן יותר מפלט אחד אפשרי? נמק.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h> /* for pid_t */
#include <sys/wait.h> /* for wait */

int do_work(const int a) {
    int i, b=0 ;
    printf("Entered a=%d\n", a) ;
    fflush(stdout) ;
    for (i=0 ; i<3 ; i++)
        b = a+b ;
    printf("Leaving b=%d\n", b) ;
    return b ;
}

int main() {
    pid_t pid=fork() ;
    if (pid==0) {
        do_work(10) ;
        exit(0) ;
    } else {
        waitpid(pid,NULL,0) ;
        do_work(20) ;
        fork() ;
        do_work(0) ;
    }
    return 0 ;
}
```

שאלה 7 [12%]

רשום תוכנית shell קצרה בשפת C או בשפת Python אשר מקבלת מהמשתמש שם של פקודה (כמו ls או dir וכדומה) ומיד לאחר מכן מבצעת את הפקודה. התוכנית shell תחכה עד לסיים הפקודה ואז תהיה מוכנה לקבל את פקודה הבאה מהמשתמש (לולאה אינסופית).
להלן דוגמא לכיצד נראה פלט של התוכנית shell :

```
cmd> ls
Lab ncgi note.html notes.log proj4 pyser.py
cmd> whoami
jsmith
cmd> ps
PID  TTY      TIME    CMD
10927 pts/1    00:00:00 bash
11807 pts/1    00:00:00 shell
12508 pts/1    00:00:00 ps
```

שאלה 8 [10%]

עיין היטב בתוכנית Python הבאה. בשתי השורות האחרונות נוצרים 10 חוטים שונים של הפונקציה foo.

```
from threading import Thread, Semaphore
x = y = 0
semx = Semaphore(2)
semy = Semaphore(2)

def foo():
    global x, y
    if x<y:
        semx.acquire()
        semy.release()
        x = y-1
    else:
        semx.release()
        semy.acquire()
        y = x+1

T = [Thread(target=foo) for i in range(10)]
for t in T: t.start()
```

- א. האם כל 10 החוטים שנוצרו בתוכנית יסיימו את עבודתם?
- ב. רשום את המספר המדויק של החוטים שיסיימו ומה יהיה הערך של y לאחר סיומם?

שאלה 9 [18%]

מערכת ניטור טמפרטורת בניין מתבססת על תהליך P שרץ באופן קבוע (על גבי מערכת Embedded Linux) ומקבל את נתוני הטמפרטורה באמצעות 80 חיישנים שונים המפוזרים על פני נקודות שונות ברחבי הבניין. כל חיישן $sensor[i]$ רושם את הטמפרטורה שדגם לתוך תור $que[i]$, כאשר $i=0,1,2,\dots,79$. לתהליך P יש גישה גלובלית לכל 80 התורים האלה. מטרת המערכת היא לחשב את הטמפרטורה הממוצעת בבניין $averageT$, הטמפרטורה המינימלית $minT$, והטמפרטורה מקסימלית $maxT$, בכל פרק זמן מחזורי של 5 שניות. החיישנים מייצרים כמויות ענק של נתוני טמפרטורה שאינן ניתנות לשליטה על ידי ליבה אחת במעבד, ונדרשות לכך שמונה ליבות.

רשום קוד Python מתאים לביצוע המשימה הזו על ידי 8 תהליכים מקבילים (למערכת יש מעבד מרובה ליבות שמאפשר זאת). המאמץ החישובי חייב להתחלק בין כל התהליכים השונים במידה מאוזנת לחלוטין.

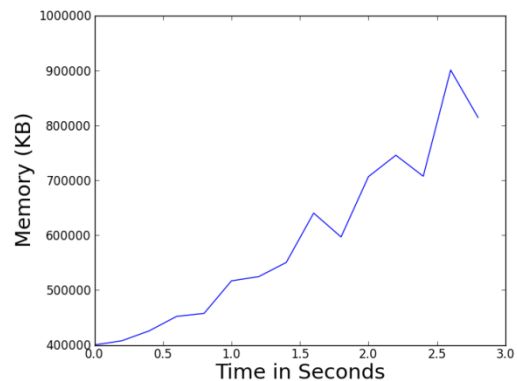
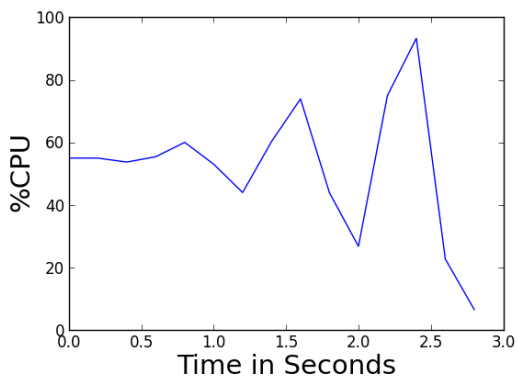
הכוונה: הנח שכל החיישנים פועלים באופן סדיר ואחיד, אינם נתקעים ואינם נעצרים. אל תטרח למדל את החיישנים עצמם ואת הקשר שלהם לתורים $que[i]$: פשוט הנח שהתורים $que[i]$ מספקים לך את הנתונים בצורה סדירה. השתמש בכל הטמפרטורות שהצטברו ב-5 השניות האחרונות מכל החיישנים. השתדל לרכז את כל החישובים הדרושים במקום אחד (סכום, ממוצע, מינימום, מקסימום).

שאלת בונוס [10%]

א. התוכנית `sysmon.exe` מדפיסה (לפלט הסטנדרטי `stdout`) את נתוני צריכת המעבד והזיכרון במערכת בכל 0.1 שניות בפורמט הבא:

```
cmd> sysmon.exe
time=2.1 cpu=38.6% mem=566123KB
time=2.2 cpu=42.1% mem=581703KB
time=2.3 cpu=45.2% mem=605893KB
...
```

- התוכנית `cpu_update.exe` מקבלת נתונים מהצורה "2.3,38.6%" מהקלט הסטנדרטי (`stdin`) ומעדכנת גרף מערכת המראה את צריכת המעבד (באחוזים) כפונקציה של זמן, כאשר $t=2.3$ הוא הזמן בשניות, ו-38.6% היא צריכת המעבד באחוזים (מופרדים על ידי פסיק).
- התוכנית `mem_update.exe` מקבלת נתונים מהצורה "2.3,581703KB" מהקלט הסטנדרטי (`stdin`) ומעדכנת גרף מערכת המראה את צריכת הזיכרון הפיזי כפונקציה של זמן. גם כאן הזמן ($t=2.3$) וכמות הזיכרון מופרדים על ידי פסיק.



כתוב תוכנית בשפת C או בשפת Python שמפעילה את שלושת התוכניות הנ"ל, ולאחר מכן מעבירה מהפלט של התוכנית `sysmon.exe`, את המידע הדרוש לתוכניות `cpu_update.exe` ו-`mem_update.exe`, בכדי לאפשר את עידכון הגרפים.

הכוונה: הנח שהגרפים נוכחים תמיד על ידי מערכת ההפעלה ומטופלים על ידיה. בכדי לפשט את התמונה, צא מהנחה ששלושת התוכניות מתחילות לפעול בזמן $t=0.0$ (על ידי התוכנית שתכתוב כמובן). ייתכן ותזדקק למנגנון ה-Pipe, אך אתה רשאי להשתמש בכל רעיון שנראה לך המתבסס על החומר שנלמד בקורס.