

## עבודה להגשה מס' 2

### ~Processes and Pipes~

תאריך הגשה: 11.04.19

- קראו היטב את השאלות.
- ניתן להגיש עבודה בזוגות.
- הגשת העבודה תהיה דרך אתר הקורס במודל.
- בעמוד הראשון של העבודה יש לרשום את שמות ומספרי ת.ז. של המגישים.
- יש להגיש את העבודה בקובץ zip בתוכו יהיו קבצים:  
ex1.c, ex2.txt, ex3.c, ex4.c, ex5.c
- שם הקובץ שיוגש למערכת ההגשה יהיה מורכב מת"ז של המגיש/ים. לדוגמה:  
111111111.zip עבור הגשה ביחיד  
111111111\_222222222.zip עבור הגשה בזוג
- במקרה של הגשה בזוגות, רק אחד מבני הזוג יגיש את העבודה במודל.
- איחור במועד ההגשה יגרור הורדה של ציון, 10 נק' לכל יום איחור או חלק ממנו.  
בכל מקרה לא יהיה ניתן להגיש מעבר לשבוע ימי איחור ממועד ההגשה המקורי.  
במקרים חריגים בלבד יש לפנות למרצה כדי לקבל אישור על הגשה באיחור.
- שאלות לגבי העבודה יש לשאול בפורום באתר הקורס ("מודל") או בשעות קבלה של המתרגל/ת האחראי/ת בלבד. אין לשלוח שאלות במייל המתרגלים או המרצה.

## עבודה נעימה!

## שאלה 1 (5 נקודות)

כתבו תוכנית בשפת C, אשר יוצרת שני תהליכים.  
תהליך האב ידפיס על המסך שורה:

```
It's a parent process- My pid is: ----. My child Pid is: ----
```

ותהליך הבן ידפיס את השורה:

```
It's a child Process- My pid is: ----. My parent pid is: ----
```

לדוגמא, במחשב מסוים אלו המספרים שיודפסו:

```
It's a parent process- My pid is: 7789. My child Pid is: 7790
It's a child Process- My pid is: 7790. My parent pid is: 7789
```

את הפתרון של שאלה זו יש להגיש בקובץ `ex1.c`.

## שאלה 2 (20 נקודות)

נתונה התוכנית הבאה :

```
01 int main() {
02     int ipc[2];
03     pipe(ipc);
04     write(2, "start\n",6);
05     if (fork()) {
06         sleep(1);
07         char c;
08         do {
09             read(ipc[0], &c,1);
10             write(1, &c,1);
11         } while (c !='\n');
12         write(1,"if\n", 3);
13         waitpid(-1, NULL, 0);
14     }
15     else {
16         write(ipc[1],"else1\n", 6);
17         write(1, "else2\n", 6);
18     }
19     write(1, "done\n", 5);
20     return 0;
21 }
```

### Explanation:

**sleep()** - The Linux kernel uses the sleep() function, which takes a time value as a parameter that specifies the minimum amount of time (in seconds that the process is set to sleep before resuming execution). This causes the CPU to suspend the process and continue executing other processes until the sleep cycle has finished.

1. הסבירו, וציינו מקור (מס' שורת קוד + האם פלט התקבל מתהליך אב או מתהליך בן) של כל שורת פלט. הניחו כי ל- pipe יש buffer פנימי בגודל מספק.
2. (4 נקודות) מה יהיה השינוי בפלט, או למה לא יהיה שינוי, אם נמחק את שורה 13?
3. (4 נקודות) תארו את אופן הפעולה של פונקציה pipe בקוד הנ"ל.

את הפתרון של שאלה זו יש להגיש בקובץ טקסט ex2.txt.

## שאלה 3 (25 נקודות)

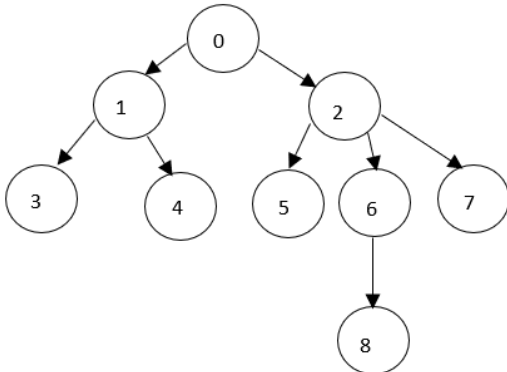
נתון מערך של מספרים שלמים חיוביים.  
כתבו תוכנית אשר לכל צומת תיצור מספר בנים התואם לערך הנמצא במערך.

לדוגמא: אם המערך הוא  $[2,2,3,0,0,0,1,0,0]$

אזי המבנה המתאים הוא:

לצומת מס' 0 יש ליצור 2 בנים, לצומת מס' 1 יש ליצור 2 בנים, לצומת מס' 2 יש ליצור 3 בנים, וכך הלאה.

הספירה של הצמתים היא בצורה של level order.



על כל תהליך יש לבצע שתי פעולות הדפסה:

- (1) יש להדפיס את מס' הסידורי של התהליך ואת pid שלו בפורמט:

Process number N has pid= pidnum

כאשר N ו- pidnum הם מספר סידור ו- pid של התהליך בהתאמה.

- (2) במידה והתליך אינו שורש, יש להדפיס את מספר המזהה של התהליך ומספר מזהה של תהליך האב בפורמט

I am process with pid= pidnum and my parent pid= ppidnum

כאשר ppidnum ו- pidnum הם מזהים של תהליך ותליך האב שלו.

למשל, פלט התוכנית עבור המערך שבדוגמה יראה באופן הבא (בעמוד הבא):

```
Process number 0 has pid=16706
Process number 2 has pid=16708
Process number 1 has pid=16707
Process number 7 has pid=16711
I am process with pid=16711 and my parent pid=16708
Process number 6 has pid=16710
Process number 5 has pid=16709
I am process with pid=16709 and my parent pid=16708
Process number 4 has pid=16713
I am process with pid=16713 and my parent pid=16707
Process number 3 has pid=16712
I am process with pid=16712 and my parent pid=16707
I am process with pid=16707 and my parent pid=16706
Process number 8 has pid=16714
I am process with pid=16714 and my parent pid=16710
I am process with pid=16710 and my parent pid=16708
I am process with pid=16708 and my parent pid=16706
I am process with pid=16706 and my parent pid=16187
```

את הפתרון של שאלה זו יש להגיש בקובץ **ex3.c**.

## שאלה 4 (25 נקודות)

כתבו תוכנית אשר תקלוט מהמשתמש מספר חיובי שלם.

לאחר מכן, התוכנית תיצור שרשרת תהליכים בגודל המתאים.

ברגע ששרשרת מוכן, תהליך השורש יבדוק האם כל תהליכי הבנים באמת נוצרו, ולכן הוא ינסה

לספור את הבנים שלו על ידי שליחת הודעות (דרך pipes) באופן הבא:

האב ישלח הודעה לבן שלו,

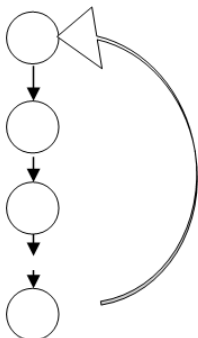
הבן ישלח בתורו הודעה לבן שלו,

וכך הלאה..

ברגע שההודעה תגיע לצומת האחרון, הוא ישלח אותה חזרה לשורש.

כמובן שבמהלך שליחת ההודעות תתבצע ספירה ותהליך האב ידפיס

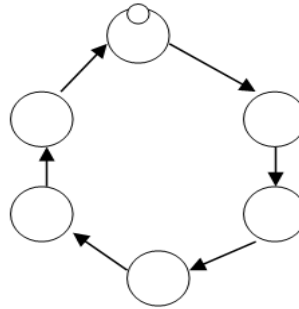
הודעה מתאימה.



את הפתרון של שאלה זו יש להגיש בקובץ **ex4.c**.

## שאלה 5 (25 נקודות)

בהינתן  $n$  תהליכים נרצה לבנות אלג' מתייצב עצמית אשר בכל רגע נתון מאפשר לכתוב אך ורק לתהליך אחד על המסך. דבר ראשון נסדר את התהליכים במעגל:



נבחר מנהיג (תהליך ראשי) ונבצעה אלג' הבא:  
עבור תהליך מנהיג:

1. קרא ערך של התהליך שלפניך (האחרון במעגל)
2. אם הערך של התהליך זהה לערך שלך אזי
  - א. יש עדכן את הערך של המנהיג ל-  $(\text{ערך} + 1) \pmod n$
  - ב. יש להדפיס את הערך של המנהיג על המסך

עבור תהליך שאינו מנהיג:

1. קרא ערך של התהליך שלפניך
2. אם הערך שונה משלך אזי
  - א. יש עדכן את הערך של התהליך לערך של התהליך הקודם
  - ב. יש להדפיס את הערך של התהליך על המסך

כתבו תוכנית אשר תקלוט מהמשתמש מספר שלם חיובי, תיצור מעגל של  $n$  תהליכים, לכל אחד מהתהליכים יוגרל ערך התחלתי שרירותי ותיתן תפריט הבא למשתמש:

- |   |   |
|---|---|
| 0 | הצג ערכים של התהליכים                                   |
| 0 | בצע צעד (כל תהליך מבצע צעד אחד בדיוק "מערכת סינכרונית") |
| 0 | יציאה   |

לתשומת ליבכם- אם מימשתם נכון את האלגוריתם, אזי לאחר מספר צעדים בכל צעד תודפס בדיוק הודעה אחת ובמעגל יהיו לכל היותר 2 מספרים שונים עוקבים.

ראו דוגמה לפלט בעמוד הבא.

את הפתרון של שאלה זו יש להגיש בקובץ `ex5.c`.

```

Please enter number of processes to create: 6
-----
Menu
1. Show values
2. Pulse
0. Exit
>>1
The process 5 has value 5.
The process 4 has value 4.
The process 3 has value 3.
The process 2 has value 2.
The process 1 has value 1.
The process 0 has value 0.
-----
Menu
1. Show values
2. Pulse
0. Exit
>>2
Permission to write at process 4
Permission to write at process 5
Permission to write at process 3
Permission to write at process 2
Permission to write at process 1
-----
Menu
1. Show values
2. Pulse
0. Exit
>>0
    
```

עבודה נעימה!