

# מכללה אקדמית הדסה

## החוג למדעי המחשב

### תרגיל #5: בקורס תכנות מערכת ומבוא לתכנות מקבילי--

#### shared memory

##### תכנית a

נחזור לתכנית a בתרגיל #4.

התרגיל יכלול שני תהליכים שיהוו שרתים, יעבדו בשת"פ, כמעין 'מערכת', ויחזיקו את הנתונים הבאים:

א. שרת רישום: שומר בז"מ מספרי תהליכים 'שהתחברו למערכת'. השרת מייצר תור הודעות (לטובת לקוחות), וכן מקצה ז"מ שישמש אותו ואת שותפו: שרת האפליקציות.

השרת	מאפשר	שתי	פעולות:
1.	הוספת	תהליך	חדש.
2.	הסרת	תהליך	קיים.

עת הוא מקבל הודעה (דרך תור ההודעות) המבקשת להוסיף תהליך. הוא בודק האם מספר התהליך המופיע בה קיים בז"מ. במידה ומספר התהליך לא קיים, וכן יש מקום במבנה הנתונים (מערך לא ממוין המוחזק בז"מ) הוא מוסיף את התהליך.

הוא משיב את התשובה (מבחינת תוכן ההודעה): 0 התהליך הוסף בהצלחה, 1 התהליך לא הוסף שכן הוא קיים כבר, 2 התהליך לא הוסף שכן מ"נ מלא.

עת הוא מקבל הודעה המבקשת להסיר תהליך, הוא מסיר אותו מהז"מ (ואינו משיב).

שרת הרישום ירוץ לנצח. עת הוא מקבל את הסיגנל SIGTINT, הוא פונה ל-signal handler, משחרר את התור ואת הז"מ, ומסיים. שם קובץ המקור: ex5a1.c

ב. שרת אפליקציות: מאפשר למי שכבר התחבר למערכת לבצע פעולות: (א) לבדוק ראשוניות, (ב) לבדוק פלינדרומיות. השרת מייצר תור הודעות.

השרת	מאפשר	שתי	פעולות:
1.	בדיקת	ראשוניות	של מספר.
2.	בדיקת	פלינדרומיות	של מחרוזת.

עת הוא מקבל הודעה המבקשת לבצע פעולה, הוא בודק האם המבקש מוכר למערכת, כלומר האם הוא מופיע בז"מ. במידה וכן, הוא מבצע את הפעולה ומחזיר תשובה (0 או 1), במידה והמבקש לא רשום במערכת הוא מחזיר את הערך -1.

סיום: כמו שרת הרישום. שם קובץ המקור: ex5a2.c

מעבר לשני אלה ירוצו תהליכי לקוח (הם יורצו רק אחרי ששני התהליכים הקודמים כבר רצים, כלומר הרצת הפרויקט תהיה ע"י שיורצו, שרת הרישום, שרת אפליקציות, ורק אח"כ הלקוחות):

א. הלקוח מתחבר לשני תורי ההודעות, ונרשם אצל שרת הרישום (במידה ולא ניתן לרשמו הוא מוציא הודעת שגיאה, ומסיים).

ב. בלולאה:

1. הלקוח קורא מהמשתמש את התו n או s המציינים האם המשתמש רוצה לטפל במספר או במחרוזת. עתה הלקוח קורא את הנתון הדרוש (מספר טבעי או מחרוזת).

2. הלקוח שולח את ההודעה המתאימה לשרת האפליקציות, ממתין לתשובה, ומציג אותה

3. עת הלקוח קורא את התו e הוא שולח הודה לשרת הרישום כי יש להסירו מהמערכת, והוא מסיים.

שם קובץ המקור: ex5a3.c

### תכנית b

כתבו תכנית המבצעת את המשימה הבאה:

תהליך הורה מקצה ז"מ ומוליד שני תהליכים ילדים.

כל אחד מהילדים בלולאה מגריל מספרים, עד הגרלת מספר ראשוני.

עת התהליך (הילד) הגריל מספר ראשוני הוא מאחסן אותו בז"מ, הורג את אחיו, שולח סיגנל להורה, ומסיים.

ההורה מציג את המספר, ומסיים גם הוא.

הסבירו האם כן/לא קיימים מצבי מרוץ בתכנית שלכם.

### תזכורות חשובות:

א. הקפידו להרוג תהליכים שרצים בלולאה אינסופית. פקודות ה- Shell: ps, kill יעזרו לכם

ב. הקפידו לצאת מלינוקס בצורה מסודרת לא ע"י לחיצה על ה- X בפינת החלון, אלא ע"י

**תפריט: system/logout**

ג. זכרו את פקודות ה- Shell הרלוונטיות: **ipcs, ipcrm** ועשו בהן שימוש בעת הצורך...

ד. זכרו: בלינוקס אנו תלויים זה בזה! ואם לא נעבוד בצורה מסודרת כולנו מסבול מאוד!!