

## מטלה 5 תהליכים – Readme

במטלה זו מימשנו את הפונקציונאליות של מטלה רביעית רק בעזרת fork והקצאת זיכרון בתצורה של מפה mmap ועבודה עם מיוטקס בשפת C בספריית fcntl.h, מימשנו STACK אשר שומר קריאות ובקשות מסוג:

- PUSH
- POP
- TOP
- EXIT

אשר מטופלים על ידי SWITCH CASE שמקבל קלט מהמשתמש .

### **הסבר קטן על מערכת התקשורת שמתמשת בפרוטוקול TCP :**

עשינו תקשורת מרובת משתמשים בעזרת fork לכל משתמש בשפת

משלו להיכנס אלי PORT | IP של סוקט וכל משתמש קיבל HEADER מימשנו

וגם אפשר להשתמש שוב באותו הפורט כדי שיהיה אפשר לנצל כל פורט שישתחרר

- מימוש fork:

בנינו פונקציה אשר בונה סוקט ומקשיב לכל חיבור , ברגע שהקליינט רוצה להתחבר מוקצה לו פורט ואז לחיצת היד הושלמה וניתן להמשיך את החיבור או לנתק , יצרנו טרד וצירפנו את הפונקציה הזאת לשם כדי שתעבוד במיין.

- טיפול בזיכרון:

כל אובייקט שפתחנו דאגנו שישחרר את הזיכרון שהוא תופס , כל סוקט שפתחנו ווידאנו שהוא נכבה כאשר הסתיימה התקשורת גם מצד הקליינט וגם מצד השרת

טיפול בזיכרון בוצע על ידי ספריית mmap אשר מקצה זיכרון באופן דינמי למגוון רכב של איברים בצורה יחסית אוטומטית , כמו כן שימוש במיוטקס הביא לנו שליטה על נעילת תהליכים עד שלא נגמרים לאותו המשתמש

### **בדיקות:**

ביצענו בדיקה לא טריויאלית על המחסנית האם היא טוענת נכון בקשות ומציצה לתוכם , כאשר הבדיקה לא צלחה תהיה הודעת שגיאה של EQUAL ASSERT ואם היא צלחה הבדיקה אז לא יוצגו שום שגיאות ויודפסו הערכים שהוכנסו