

מטלה 5 – מערכות הפעלה
תאריך הגשה: התרגיל יוגש עד ה 04/6/23 AM 8:00

שלב א:

ממשו פונקציה המקבלת unsigned int הבודקת אם המספר הוא ראשוני (רק אי זוגיים, עד השורש, *לא צריך להריץ מילר רבין* וכו') הפונקציה תחזיר 0 אם המספר לא ראשוני או ערך אחר אחרת.

שלב ב:

ממשו תור בסביבה מרובת טרדים.
התור צריך להיות עם הגנות של מיוטקס וגם לאפשר המתנה לאיבר בתור ללא ביזי וואיטינג (לדוגמא בעזרת (cond
התור יחזיק void *.

שלב ג:

ממש active object שתומך בפונקציות הבאות

א. CreateActiveObject
פונקציה שתאלקץ ותריץ טרד של active object.
בנוסף לכך הפונקציה תאלקץ תור וגם תקבל מצביע לפונקציה שתקרא עבור כל איבר בתור.

הביזי לופ של ה AO יראה כך

```
while (task=this->queue->dequeue())  
;this->func(task)
```

ב. getQueue
פונקציה שתחזיר מצביע לתור של activeObject שמתקבל כפרמטר (this).
(ניתן יהיה להשתמש בפונקציה הזאת כדי להכניס איבר לתור)

ג. stop
פונקציה שתעצור activeObject שמתקבל כפרמטר this.
בנוסף לכך הפונקציה תשחרר את כל מבני הזכרון של האובייקט.

שלב ד - המשימה:

השתמשו בפונקציות שבניתם עד כה:

כתבו תוכנה בשם `st_pipeline` אשר תקבל ארגומנט אחד או שניים (בשורת הקלט), הראשון הוא `N` (המייצג את כמות המשימות). השני הוא `random seed`. אם לא יתקבל `random seed` ניתן להגדיל אחד בעזרת `time(null)`

* להזכיר, `pipeline` הוא אוסף של `active object (AO)`.

בנו `pipeline` באופן הבא:

ה `AO` הראשון יבצע:

- (1) יאתחל את המחולל הרנדומי בעזרת הזרע שניתן ואז יחולל `N` מספרים בני 6 ספרות כל אחד (אפשר `3 + 3`) - בעזרת `rand`.
- (2) יעביר אחד אחד ל `AO` שאחריו, בהפרשים של אלפית השניה - `sleep(1)`.

ה `AO` השני:

- (1) ידפיס את המספר, יבדוק האם המספר הוא ראשוני, וידפיס אמת או שקר בהתאם.
- (2) יוסיף למספר 11, ויעביר אותו ל `AO` הבא.

ה `AO` השלישי:

- (1) ידפיס את המספר, יבדוק האם המספר הוא ראשוני, וידפיס אמת או שקר בהתאם.
- (2) יוריד מהמספר 13, ויעביר ל `AO` הרביעי.

ה `AO` הרביעי ידפיס את המספר שהתקבל, יוסיף למספר 2, וידפיס את המספר החדש.
(אם עשיתם הכל נכון, המספר האחרון יהיה שווה לראשון)

לאחר ביצוע `N` הגרלות וחישוביהן, התוכנית תסתיים.

דוגמה להרצה:

```
./st_pipeline 2
123456
false
123467
false
123454
false
123456
```

שים לב שימוש באותו זרע רנדומלי אמור לייצר את אותו רצף מספרים פסאודו אקראיים ולכן יאפשר בדיקה אוטומטית

בהצלחה

הנחיות כלליות:

- התוכנית תכתב בשפת C או ++C.
- עליכם להגיש את כל קבצי המקור, ובנוסף לכך Makefile.
- Makefile לא תקין, מכל סיבה שהיא, עלול להוריד את הציון דרמטית, עד ל 0.
- ה Makefile יכיל לפחות את פקודות all שמקמפלת את התוצרים, ואת פקודת clean שמוחקת את תוצאות הקימפול של פקודת ה all וכן את כל ה kraft.
- עליכם להגיש קובץ הסבר להפעלת המטלה, בטקסט חופשי ודוגמאות הרצה. שם הקובץ יהיה readme.txt - במידה ולא קיים, ירדו 10 נקודות.
- המטלה עלולה להיבדק אוטומטית, ולכן אין להוסיף קבצים מיותרים, בפרט קבצים מקומפלים.
- את המטלה יש להגיש בקובץ ZIP (לא RAR/TAR/7ZIP) – במידה ולא, ירדו 15 נקודות.
- שם קובץ ההגשה (ה ZIP) יהיה כמספר ת.ז. של המגיש. במידה ומגישים בזוג, יש להגיש רק פעם אחת, ושם הקובץ יהיה ת.ז_ת.ז – במידה ולא, ירדו 15 נקודות.
- לא ניתן לקבל הארכה. במקרה של סיבה מוצדקת (באישור המחלקה) ניתן להוריד את המטלה מחשבון כלל ההגשות של הסטודנט.
- יש להקפיד על שמות קבצי ההרצה, כפי שמופיע ב uasge עבור כל סעיף שבו נדרש קובץ הרצה.
- במידה ונדרשת פונקציות מעבר לדרישות בתרגיל (כגון ערוץ תקשורת נוסף, או API שלא צוין במפורש) , אתם רשאים לממש בכל דרך סבירה ויעילה, ולהסביר את הנעשה בקובץ ה readme.txt.
- מומלץ להשתמש בקודים קודמים, ומקורות אחרים באינטרנט.
- במידה והשתמשתם באתר כלשהו – צרפו את הקישור לכך (בקובץ ההסבר).
- במידה והשתמשתם ב ChatGPT (כן, מותר !), צרפו את מלל הבקשה המדויק שהעברתם לו.
- ניתן ליצר קבצי עזר כרצונכם, אך בסופו של דבר, אני דורש שיהיו קבצי הרצה ו/או ספריות כמו שהתבקשתם.
- במידה וחסר לכם פרט במטלה, (כדוגמת מספר התהליכים ב threadpoll) עליכם לקבל החלטה לבד, ולנמק זאת בקובץ ה readme.txt. מומלץ להביא אסמכתא לנימוק - למשל אתר שממליץ על אסטרטגיה מסוימת.