

תרגיל 4, מבוא לתכנות מערכות, חורף 2022-2023

הגשה בזוגות או ביחידים דרך המודל
התרגיל הוא להגשה עד ליום שלישי, 3/1/2023 בשעה 23:59

Abstract Data Types (ADT)

בתרגיל זה תממשו מבנה נתונים בשם תור ישראלי. מבנה הנתונים הזה מממש פעולות של תור כפי שלמדנו בכתה ויש לו, בנוסף, פעולות שמאפיינות תור ישראלי. את ה-ADT תממשו בעזרת רשימה מקושרת, שהמימוש שלה נתון במודל (LinkedList.c ו-LinkedList.h). אין לשנות את מימוש (או ממשק) הרשימה המקושרת, אתם צריכים להשתמש בו כמו שהוא. אין להשתמש במערכים או במבנה נתונים אחר לצורך מימוש התור הישראלי.

א. מימוש התור הישראלי

- הגדירו ADT גנרי בשם `ISRQueue` שכולל את הפעולות הבאות. הקפידו להמנע משכפול קוד ככל שניתן.
- `ISRCreate` – יצירת תור ישראלי ריק.
 - `ISRDestruct` – מחיקת תור ישראלי. יש למחוק את כל האלמנטים שנותרו בתור
 - `ISREnqueue` – הוספת איבר לתור. האיבר יתווסף לסוף התור בהתאם לכללים של תור רגיל
 - `ISREnqueueFirst` – הוספת אלמנט לתחילת התור. מתאים למצבים של "יש לי רק שאלה לרופאה". כמובן שיש מצבים לגיטימיים שבהם אנשים ייכנסו לתחילת התור, למשל כאלו שיש להם פטור מתור.
 - `ISREnqueueAfter` – מקבלת שני אלמנטים, `newElement` - אלמנט להכנסה ו-`existingElement` - אלמנט שזהה לאלמנט שכבר נמצא בתור. הפונקציה תכניס את `newElement` לאמצע התור, מייד אחרי האלמנט שזהה ל-`existingElement`. אם קיימים כמה אלמנטים שזהים ל-`existingElement` האלמנט החדש יוכנס אחרי האלמנט הקרוב ביותר לתחילת התור. אם לא קיים אלמנט שזהה ל-`existingElement` האלמנט החדש יתווסף לסוף התור. מתאים למצבים של "הוא אח שלי ושמר לי מקום בתור".
 - `ISRDequeue` – הוצאה והחזרה של אלמנט מתחילת התור
 - `ISRQueueSize` – החזרת אורך התור
 - `ISRQueueSplit` – משאירה בתור המקורי את $\text{int}(\text{ISRQueueSize}(\text{queue})/2)$ האלמנטים הקרובים לתחילת התור ומחזירה תור חדש שיכיל את שאר האלמנטים בסדר אקראי (מתאים למצבים של "פתחתי עוד קופה, אתם יכולים לבוא לכאן"). לצורך יצירת תור אקראי בחרו בכל פעם בצורה אקראית את האלמנט הבא שייכנס לתור החדש, הוציאו אותו מהתור המקורי והכניסו אותו לסוף התור החדש. אם בתור המקורי יש אלמנט אחד או פחות יוחזר תור חדש ריק.

ב. שימוש בתור הישראלי

ממשו פונקציית `main` שעושה שימוש ב-`ISRQueue` בשביל לשמור מחרוזות שפעולת ההשוואה ביניהן מתבצעת ללא הבחנה בין אותיות uppercase ל-lowercase. למשל: המחרוזות "Hello" ו-"hello" תחשבנה זהות. אין לשנות את המחרוזות כאשר מייצרים עותק חדש שלהן. בצעו את הפעולות הבאות ב-`main`:

- `ISRQueueCreate`
- `ISREnqueue "Beth"`
- `ISREnqueue "Kylian"`
- `ISREnqueue "Zlatan"`
- `ISREnqueue "Mohamed"`
- `ISREnqueue "Ada"`
- `ISRDequeue`

- ISREnqueueFirst "Lionel"
- ISREnqueueAfter "Karim", after "Kylian"
- ISREnqueueAfter "Megan", after "Alex"
- ISRQueueSplit
- Print the contents of the two queues by the order of their elements. Remove the elements from the queues while doing so
- ISRQueueDestroy for the two queues

הגשה

הגישו את קבצי ISRQueue.h, ISRQueue.c ו-main.c שיצרתם בקובץ zip דרך המודל.

בהצלחה!