

## תרגיל 2, מבוא לתכנות מערכות, חורף 2020-2021

הגשה בזוגות או ביחידים דרך המודל  
התרגיל הוא להגשה עד ליום חמישי, 19/11/2020 בשעה 23:59

### בניית מילון בעזרת פונקציית hash

בתרגיל הזה תיצרו תוכנית מילון של מילים. התוכנית תקלוט מילים ותשמור אותן במערך על ידי יצירת מספר ייחודי לכל מילה, ושמידת המספר במקום המתאים במערך. התוכנית תאפשר שתי פעולות: הוספת מילה ובדיקה האם מילה קיימת. כמו כן הפונקציה תאפשר יציאה מסודרת מהתוכנית.

### משימות

#### פונקציית hash

פונקציית hash היא פונקציה שמקבלת ערך (במקרה שלנו: מחרוזת) ויוצרת מספר ייחודי לכל ערך. פונקציית hash מושלמת יוצרת לכל ערך מספר שונה, ותמיד את אותו מספר לכל ערך. פונקציית hash-ה שאותה תממשו מממשת את האלגוריתם הבא:

**קלט:** מצביע למחרוזת מטיפוס `const char*`

**פלט:** מספר מטיפוס `unsigned long`

**אלגוריתם:**

- אתחלו משתנה בשם `h` מטיפוס `unsigned long` ל-5381.
- בלולאה, בצעו לכל תו `c` במחרוזת  $(h = h * 33 + c)$ : כאשר `c` הוא הערך ה-ASCII של התו `c`.
- החזירו את `h`.

משימות:

ממשו שתי גרסאות לפונקציית hash-ה:

- `hash_arr` – מממשת את האלגוריתם תוך שימוש בסוגריים מרובעים וכתוב של מערכים בלבד
- `hash_ptr` – מממשת את האלגוריתם בכתוב של מצביעים בלבד

#### פונקציה שבודקת חוקיות מילים

ממשו פונקציה בשם `is_legal_word` שמקבלת מחרוזת (`const char*`) ומחזירה `true` אם המחרוזת מכילה רק אותיות (גדולות או קטנות) או `false` אחרת.

#### פונקציה שמשנה מחרוזת למחרוזת ללא אותיות גדולות

ממשו פונקציה בשם `word_tolower` שמקבלת מצביע למחרוזת (`char*`) ומשנה את כל האותיות הגדולות במחרוזת לקטנות. אין לקרוא לאף פונקציה (בפרט לא ל-`tolower`) מתוך הפונקציה. יש להשתמש בכתוב מצביעים בלבד (ללא סוגריים מרובעים).

#### פונקציה main

ממשו את הפונקציה `main` שמבצעת:

- הגדירו מערך של 500,000 תווים (`char`) בשם `word_hash` אותו תאתחלו ל-0.
- בלולאה בצעו:
  - הדפיסו סימן \$ עם רווח אחריו למסך
  - אם המשתמש הכניס את הפקודה `ADD` ולאחריה מחרוזת, אז בידקו שהמחרוזת היא מילה חוקית (אם לא הדפיסו הודעת שגיאה וחזרו ל-א), הפכו את המחרוזת למילה עם אותיות קטנות בלבד, קראו לאחת מפונקציות ה-`hash` ושימרו את התוצאה במשתנה `i` והכניסו ל-`word_hash[i%500000]` את הערך 1.

- ג. אם המשתמש הכניס את הפקודה CHECK ולאחריה מחרוזת אז בידקו שהמחרוזת היא מילה חוקית (אם לא הדפיסו הודעת שגיאה וחזרו ל-א), הפכו את המחרוזת למילה עם אותיות קטנות בלבד, קראו לאחת מפונקציות ה-hash ושימרו את התוצאה במשתנה i והדפיסו exists אם word\_hash[i%500000] מכיל 1 או does not exist אחרת
- ד. אם המשתמש הכניס את הפקודה EXIT אז צאו מהלולאה
- ה. אחרת הדפיסו הודעת illegal command והשתמשו בפקודה fflush(stdin) כדי לנקות את ערוץ הקלט
- ב-ב ו-ג הניחו שהמחרוזות שמועברות הן ללא רווחים. הרווח היחיד שקיים הוא בין הפקודה למחרוזת.
3. החזירו 0

### דוגמאות הרצה

```
$ ADD hello
$ CHECK HELLO
exists
$ CHECK world
does not exist
$ ADD 123
illegal command
$ KUKU hello
illegal command
$ EXIT
```

ניתן למצוא מספר דוגמאות הרצה במודל

### הגשה ומימוש

אתם יכולים לממש את הקוד ב-onlinegdb.com או בכל סביבת עבודה שנוחה לכם. ודאו שהתוכנית מתקמפלט ורצה על שרת החוג. הגישו את קובץ ה-C שיצרתם דרך המודל בלבד, קראו לו ex3.c. כקובץ C (לא קובץ דחוס!) ודאו שהתוכנית שלכם מחזירה בדיוק את אותו הפלט כמו זה שבדוגמאות. הקפידו שהטקסט יהיה זהה לגמרי! חלק מהבדיקה יהיה אוטומטי.

### בהצלחה!