

שאלה 3 (20 נקודות)

(סעיף א (10 נקודות)

נתון **מערך של מחרוזות** [strings] באורך N (כש-N מוגדר כ-define), **ומפתח** אשף הוא מחרוזת. key עליכם לממש פונקציה המחשבת לכל מחרוזת במערך strings כמה תווים משותפים יש לה עם המחרוזת cals[i] כאשר חזרות נספרות, ראו דוגמה בהמשך). על הפונקציה לרשום את התוצאות למערך vals, כש-[i] מכיל את מספר התווים המשותפים למפתח עם המחרוזת ה-i. **שימו לב:** אין לשנות את מחרוזות הקלט.

לדוגמה, למחרוזת "Good morning Israel!" ולמפתח "oops" יש 3 תווים משותפים: שני 'ס' ו-'s' אחד. לעומת זאת עם המפתח "!ooooops" יש לה 5 תווים משותפים: שלושה 'ס', 's' אחד ו-'!' אחד.

דרישות סיבוכיות: סיבוכיות מקום נוסף (O(1), וסיבוכיות זמן O(N*m+k) כש-m הוא אורך המחרוזת הארוכה סיבוכיות: סיבוכיות מקום נוסף (O(1), וסיבוכיות זמן k-l כשאלה זו ניתן להתייחס למספר התווים האפשריים (256) כקבוע.

<pre>void keycount(char* strings[N], char* key, int vals[N]) {</pre>					
	void keycount (char*	strings[N],	char* key,	int vals[N])	{



סעיף ב (10 נקודות)

בשאלה זו נתון מערך של מחרוזות ומפתח, כמתואר בסעיף א'. יש לממש פונקציה **שממיינת** את מערך המחרוזות בסדר עולה על פי ערכן של המחרוזות, כאשר ערכה של מחרוזת מוגדר כמספר התווים שמשותפים לה עם המפתח. בסעיף זה ניתן להשתמש בפונקציה מסעיף א' גם אם לא פתרתם אותו.

כפי שהוגדרו (ש-N, מש-N, כש-N, מש-N, כש-N, מיבוכיות מקום נוסף (מיבוכיות מקום נוסף (מיבוכיות מקום נוסף (מיבוכיות מקום נוסף (מיבוכיות מקום נוסף א'.

<pre>void keysort(char* strings[N], char* key) {</pre>