

**המחלקה למדעי המחשב COMPUTER SCIENCE DEPARTMENT**

**סדנה מתקדמת בתכנות 61108**

סמסטר ק' תשע"א  
מועד ב'  
7.11.2011

**מרצה: ד"ר מרק קורנבליט**

משך המבחן: שלוש שעות

אין להשתמש בחומרי עזר, פרט, לדף מצורף לשאלון

**שאלה 1 (28 נק')**

כתוב פונקציה המקבלת מערך  $A$  של מספרים שלמים לא שליליים. המערך מורכב מסדרות של מספרים חיוביים המובדלות באפסים (הסדרה יכולה להיות ריקה, ז"א אפסים ב- $A$  יכולים להופיע גם ברציפות). ידוע כי האיבר הראשון והאיבר האחרון של  $A$  הם לא אפסים.

על הפונקציה לבנות מערך דו-ממדי **דינאמי** בעל שורות עם גדלים שונים כדלקמן:

- מספר שורות במערך דו-ממדי יהיה שווה למספר סדרות של מספרים חיוביים ב- $A$ ;
- איבר מס'  $0$  בשורה מס'  $i$  במערך דו-ממדי יהיה שווה למספר איברים בסדרה מס'  $i$  (משמאל לימין) במערך  $A$ ;
- תוכן של שאר האיברים בשורה מס'  $i$  במערך דו-ממדי יהיה שווה לתוכן של סדרה מס'  $i$  (משמאל לימין) במערך  $A$  (איבר מס'  $j$  בשורה מס'  $i$  יהיה שווה לאיבר מס'  $j-1$  בסדרה מס'  $i$ ).

הפונקציה תחזיר כתובת של המערך הדו-ממדי הדינאמי הבנוי ותעביר את מספר השורות בתוכו.

**דוגמא:**

מערך חדש

$A = 5, 7, 9, 0, 12, 11, 0, 1, 3, 10, 87, 0, 6, 5, 0, 0, 31$

3 5 7 9  
2 12 11  
4 1 3 10 87  
2 6 5  
0  
1 31

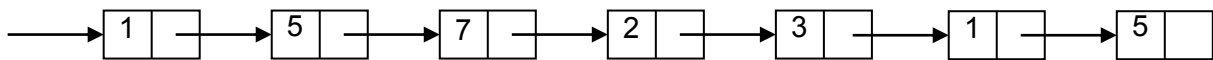
הפונקציה תעביר 6 בדוגמא.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

### שאלה 2 (28 נק')

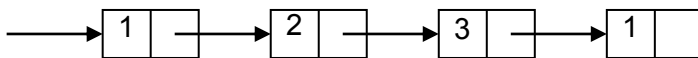
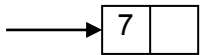
כתוב פונקציה המקבלת רשימה מקושרת של מספרים שלמים ומספר שלם  $k$ .  
על הפונקציה לבנות שתי רשימות מקושרות חדשות כך שהרשימה הראשונה תהיה מורכבת מאיברי הרשימה המקורית הגדולים מ- $k$  והרשימה השנייה תהיה מורכבת מאיברי הרשימה המקורית הקטנים מ- $k$ .  
הפונקציה תעביר את המצביעים לאיברים הראשונים של שתי הרשימות החדשות ותחזיר את מספר האיברים של הרשימה המקורית השווים ל- $k$ .

**לדוגמא**, עבור הרשימה הבאה:



ומספר  $k$  השווה ל-5

הפונקציה תיצור את שתי הרשימות החדשות הבאות:



ותחזיר 2.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

### שאלה 3 (28 נק')

כתוב פונקציה המקבלת מספר שלם (חיובי או שלילי או אפס) ומחזירה מצביע למחרוזת מספרית **דינאמית** שהיא ההצגה של המספר הנתון.  
כל תוויי המחרוזת פרט לתו הראשון יהיו ספרות. עבור המספר הנתון השלילי התו הראשון יהיה '-'.  
אחרת הוא יהיה ספרה.

**לדוגמא**, עבור המספרים 0, -569, 1896 הפונקציה תחזיר את המחרוזות "0", "-569", "1896" בהתאמה.

יש להקצות את המחרוזת בתוך הפונקציה.  
ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

לא להשתמש בפונקציה סטנדרטית **itoa** המממשת את האלגוריתם הנ"ל!

**שאלה 4 (16 נק')**  
נתונה התוכנית הבאה:

```
#include <stdio.h>
#include <string.h>

void fun (char *, const char *);

void main ()
{
    char *str="abc", msg[]="xyz";
    str = "hello";
    puts (str);
    strcpy (str, msg);
    puts (str);
    strcpy (msg, "hello");
    puts (msg);
    msg = "good bye";
    puts (msg);
    fun (msg, str);
    puts (msg);
    str = "good bye";
    fun (str, msg);
    puts (str);
}

void fun (char *s1, const char *s2)
{
    s2++;
    *s1 = *s2;
}
```

התוכנית שגויה!  
'ש לתאר את כל השגיאות (קומפילציה וריצה).

**שאלה 5 (בנוס – 10 נק')**  
מהו פלט התוכנית הבאה?

```
#include <stdio.h>
#include <ctype.h>

void main()
{
    char c = getchar();
    printf ("%d", isdigit(c)&&isalpha(c) ||
                islower(c)&&isupper(c) ||
                isalnum(c)&&isspace(c));
}
```

**בהצלחה!**