

המחלקה למדעי המחשב COMPUTER SCIENCE DEPARTMENT

מעבדה למחשבים אישיים 61105

סמסטר ק' תשס"ט

מועד א'

30.09.2009

מרצה: ד"ר מרק קורנבליט

משך המבחן: שלוש שעות

אין להשתמש בחומרי עזר, פרט, לדף מצורף לשאלון

שאלה 1 (28 נק')

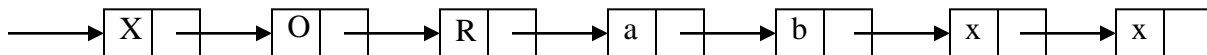
כתוב פונקציה אשר מקבלת מחרוזת ובונה שתי רשימות מקושרות. הרשימה הראשונה תהיה מורכבת מאותיות של המחרוזת המקורית. הרשימה השנייה תכלול בתוכה רק ספרות של המחרוזת. על הפונקציה להחזיר הכמות הכוללת של האיברים בשתי הרשימות.

דוגמא:

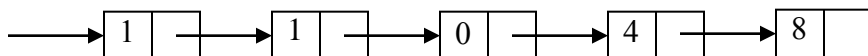
מחרוזת:

$1 \text{ XOR } 1 = 0$; $a+b=?$; $4*x=8$; $x - ?$

רשימה ראשונה:



רשימה שנייה:



הפונקציה תחזיר 12 בדוגמא.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

```

typedef struct element
{
    char data;
    struct element *ptr_next;
} element;

int two_lists (char *str, element **let_list, element **dig_list)
{
    element *let_ptr, *dig_ptr;
    int count = 0, i;

    /* allocation of dummy elements */
    *let_list = malloc(sizeof(element));
    *dig_list = malloc(sizeof(element));
    /* ----- */

    let_ptr = *let_list;
    dig_ptr = *dig_list;
    for (i=0; str[i] != '\0'; i++)
        if (isalpha(str[i]))
        {
            let_ptr->ptr_next = malloc(sizeof(element));
            let_ptr = let_ptr->ptr_next;
            let_ptr->data = str[i];
            count++;
        }
        else
            if (isdigit(str[i]))
            {
                dig_ptr->ptr_next = malloc(sizeof(element));
                dig_ptr = dig_ptr->ptr_next;
                dig_ptr->data = str[i];
                count++;
            }
    let_ptr->ptr_next = dig_ptr->ptr_next = NULL;

    /* deletion of dummy elements */
    let_ptr = *let_list;
    *let_list = (*let_list)->ptr_next;
    free (let_ptr);
    dig_ptr = *dig_list;
    *dig_list = (*dig_list)->ptr_next;
    free (dig_ptr);
    /* ----- */

    return count;
}

```

שאלה 2 (28 נק')

א. (20 נק')

כתוב פונקציה שמקבלת מטריצה סימטרית A ($a_{ij}=a_{ji}$) כמערך דו-ממדי סטטי. על הפונקציה לבנות מערך דו-ממדי דינאמי בעל שורות עם גדלים שונים אשר יישם מטריצה A בדרך חסכונית באמצעות אחסון רק משולש אחד של מטריצה A .

דוגמא:

מערך חדש

1
2 6
3 4 5

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 6 & 4 \\ 3 & 4 & 5 \end{bmatrix}$$

הפונקציה תחזיר כתובת של המערך החדש.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

```
#define MAX 5 //example

int **stat_to_dyn_sym_matrix (int a[][MAX], int size)
{
    int **a_dyn, i, j;
    a_dyn = calloc (size, sizeof(int *));
    for (i=0; i<size; i++)
        a_dyn[i] = calloc (i+1, sizeof(int));
    for (i=0; i<size; i++)
        for (j=0; j<=i; j++)
            a_dyn[i][j] = a[i][j];
    return a_dyn;
}
```

ב. (8 נק')

כתוב פונקציה המממשת פנייה לאיבר a_{ij} של מטריצה סימטרית A המיושמת באמצעות מערך דינאמי בסעיף (א). הפונקציה תקבל כתובת של המערך הדינאמי וקואורדינטות האיבר.

```
int sym_matr_element (int **a_dyn, int i, int j)
{
    if (i>=j)
        return a_dyn[i][j];
    else
        return a_dyn[j][i];
}
```

שאלה 3 (28 נק')

א. (20 נק')

כתוב פונקציה אשר מקבלת מערך בעל m מחרוזות ומבצעת היסט מחזורי של המחרוזות כך שמחרוזת מס' 0 עוברת למקום 1, מחרוזת מס' 1 עוברת למקום 2, ..., מחרוזת מס' $m-2$ עוברת למקום $m-1$ ומחרוזת מס' $m-1$ עוברת למקום 0.

דוגמא:

מערך אחרי היסט

0 ddd
1 aaaa
2 bbbbbbb
3 cc

מערך לפני היסט

0 aaaa
1 bbbbbbb
2 cc
3 ddd

```
void point_shift (char **str, int n)
{
    int i;
    char *temp = str[n-1];
    for (i=n-1; i>0; i--)
        str[i] = str[i-1];
    str[0] = temp;
}
```

ב. (8 נק')

תשנה (קצת) את פונקציה סעיף (א), כך שהיא תהיה אוניברסאלית כלפי טיפוס ותבצע היסט מחזורי על מערך דו-ממדי דינאמי בעל איברים של טיפוס שרירותי.

```
void point_shift (void **str, int n)
{
    int i;
    void *temp = str[n-1];
    for (i=n-1; i>0; i--)
        str[i] = str[i-1];
    str[0] = temp;
}
```

שאלה 4 (16 נק')
נתונה התוכנית הבאה:

```
#include <stdio.h>

void fun (const int *);

void main()
{
    int b[] = {1, 2, 3}, *c = b, *const d = b, i;

    for (i=0; i<3; i++)
        b[i]++;
    b++; error C2105: '++' needs l-value
    for (i=0; i<3; i++)
        c[i]++;
    c++;
    for (i=0; i<3; i++)
        d[i]++;
    d++; error C2166: l-value specifies const object
    fun (b);
    fun (c);
    fun (d);
    for (i=0; i<3; i++)
        c[i]++; may be runtime error for i==2
    for (i=0; i<3; i++)
        printf ("%d %d %d \n", b[i], c[i], d[i]);
}

void fun (const int *a)
{
    int i;
    for (i=0; i<3; i++)
        a[i]++; error C2166: l-value specifies const object
    a++;
}
```

התוכנית שגויה!

'ש לתאר כל שגיאות קומפילציה אשר ייתן המהדר וגם שגיאות ריצה אשר יכולות להתגלות אחרי תיקון שגיאות הקומפילציה והרצת התוכנית.

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

```
#include <stdio.h>

void main()
{
    static int x;
    int *xx = &x, **xxx = &xx;
    printf ("%d", '*'**xxx);
}
```

0

בהצלחה!