

המחלקה למדעי המחשב COMPUTER SCIENCE DEPARTMENT

סדנה מתקדמת בתכנות 61108

סמסטר ב' תשע"ב

מועד א'

25.06.2012

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 מערך דו-ממדי דינאמי בעל n שורות ומבצעת היסט מחזורי של שורותיו כך ששורה מס' 0 עוברת למקום 1, שורה מס' 1 עוברת למקום 2, ..., שורה מס' $n-2$ עוברת למקום $n-1$ ושורה מס' $n-1$ עוברת למקום 0.

דוגמא:

<u>מעריך לפני היסט</u>	<u>מעריך אחרי היסט</u>
0 5 12 6 9	0 2 20 4
1 0 3 1 29 101	1 5 12 6 9
2 13 14	2 0 3 1 29 101
3 2 20 4	3 13 14

זמן ריצת הפונקציה יהיה תלוי רק בכמות שורות במערך ולא יהיה תלוי בגדלי השורות!

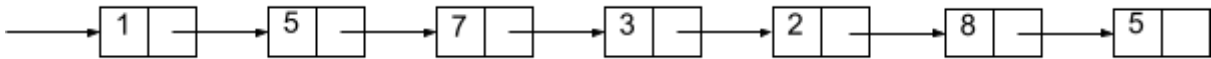
על הפונקציה להיות אוניברסאלית כלפי טיפוס איברי המערך, ז"א לבצע היסט מחזורי על המערך בעל איברים של טיפוס שרירותי.

```
void row_shift (void **arr, int rows)
{
    int i;
    void *temp = arr[rows-1];
    for (i = rows-1; i>0; i--)
        arr[i] = arr[i-1];
    arr[0] = temp;
}
```

}

שאלה 2 (28 נק')

כתוב פונקציה המקבלת רשימה מקושרת של מספרים שלמים ומספר שלם k .
על הפונקציה לבנות שני מערכים דינאמיים כך שהמערך הראשון יהיה מורכב מאיברי הרשימה הגדולים מ- k והמערך השני יהיה מורכב מאיברי הרשימה הקטנים מ- k .
הפונקציה תעביר את הכתובות של המערכים ואת הגדלים שלהם ותחזיר את מספר האיברים של הרשימה השווים ל- k .
לדוגמא, עבור הרשימה הבאה:



ומספר k השווה ל-5

הפונקציה תיצור את שני המערכים הבאים:

7	8
---	---

1	3	2
---	---	---

הפונקציה תעביר גם מספרים 2 ו-3 ותחזיר 2.

במידה ואחד מהמערכים לא יכיל אף איבר, כתובתו תהיה NULL וגודלו יהיה 0.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

```
typedef struct element
{
    int data;
    struct element *ptr_next;
} element;

int large_small_num (element *lst, int k, int **array_large,
                    int **array_small, int *size_large, int
                    *size_small)
{
    int count_large = 0, count_small = 0, equal_num = 0;
    element *ptr = lst;
    /* Estimation of array sizes */
    *size_large = *size_small = 0;
    while (ptr)
    {
        if (ptr->data > k)
            (*size_large)++;
        else if (ptr->data < k)
            (*size_small)++;
        else
            equal_num++;
        ptr = ptr->ptr_next;
    }
    /* Allocation of arrays */
    if (*size_large)
        *array_large
        = (int*) malloc (*size_large * sizeof(int));
```

```

else
    *array_large = NULL;

if (*size_small)
    *array_small =
        (int *)malloc(*size_small*sizeof(int));
else
    *array_small = NULL;
if (*size_large + *size_small == 0)
    return equal_num;
/* Filling arrays */
while (lst)
{
    if (lst->data > k)
    {
        (*array_large)[count_large] = lst->data;
        count_large++;
    }
    else
        if (lst->data < k)
        {
            (*array_small)[count_small] = lst->data;
            count_small++;
        }
    lst = lst->ptr_next;
}
return equal_num;
}

```

שאלה 3 (28 נק')

- כתוב פונקציה (void) אשר מקבלת מחרוזת המורכבת מזוגות של תווים כך שהתו הראשון בכל זוג הוא ספרה. על הפונקציה לעבד את המחרוזת כדלקמן:
- כל זוג שהתו השני בו הוא אות, יהפוך לסדרת תווים השווים לאות זאת. אורך הסדרה יהיה שווה למספר המיוצג ע"י הספרה שהיא התו הראשון בזוג;
 - כל זוג שהתו השני בו הוא גם ספרה יישאר במחרוזת ללא שינוי;
 - הפונקציה תסיר מהמחרוזת כל זוג שהתו השני בו הוא לא אות ולא ספרה.

דוגמא:

**3a6*0k4n901F7\$
aaannnnn90F**

מחרוזת לפני עיבוד:
מחרוזת אחרי עיבוד:

ניתן להניח שהגודל הפיסי של המחרוזת הנתונה יהיה מספיק למחרוזת המעובדת.

אפשר (אבל לא חובה) להשתמש במחרוזת עזר. במקרה זה ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

```

void string_transform (char *str)
{
    int i, j;
    char c, *temp = (char *)malloc(strlen(str)+1);
    strcpy (temp, str);
    for (i=0, j=0; temp[i]!='\0'; i+=2)
    {
        if (isalpha(temp[i+1]))
            for (c='1'; c<=temp[i]; c++, j++)
                str[j] = temp[i+1];
        else
            if (isdigit(temp[i+1]))
            {
                str[j] = temp[i];
                str[j+1] = temp[i+1];
                j+=2;
            }
    }
    str[j] = '\0';
    free (temp);
}

```

שאלה 4 (16 נק')
נתונה התוכנית הבאה:

```

#include <stdio.h>

#define ROWS 3
#define COLS 4

void input_array (int **, int, int);
void output_array (int[][COLS], int, int);

void main()
{
    int a[ROWS][COLS];
    // int ** is not an array and can not be initialized
    int **b = {{1,2,3,4}, {5,6,7,8}, {9,10,11,12}},
            **p, i;
    int (*c)[COLS];
    int *d[ROWS];
    p = b;
    printf ("%d\n", p[0][0]);
    input_array (a, ROWS, COLS); // different types for
                                formal and actual parameter 1

    c = a;
    output_array (c, ROWS, COLS);
    input_array (d, ROWS, COLS); // running error:
                                memory for rows was not allocated
}

```

```

    output_array (d, ROWS, COLS); // different types for
                                   formal and actual parameter 1
    d = b; // '=' : left operand must be l-value

    output_array (d, ROWS, COLS); // different types for
                                   formal and actual parameter 1
}

void input_array (int **arr, int rows, int cols)
{
    int i, j;
    for (i=0; i<rows; i++)
        for (j=0; j<cols; j++)
            scanf ("%d", &arr[i][j]);
}

// parameter can not be const
void output_array (int arr[][COLS], int rows, int COLS)
{
    int i, j;
    for (i=0; i<rows; i++)
    {
        for (j=0; j<COLS; j++)
            printf ("%d ", arr[i][j]);
        printf ("\n");
    }
}

```

התוכנית שגויה!
 'ש לתאר את כל שגיאות והזהרות הקומפילציה וגם שגיאות הריצה.

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

```

#include <stdio.h>

void main()
{
    printf ("%s\n", "?????":":");
}

```

?