

המחלקה למדעי המחשב COMPUTER SCIENCE DEPARTMENT

סדנה מתקדמת בתכנות 61108

סמסטר קיץ תשע"ד

מועד א'

30.09.2014

מרצים: ד"ר מרק קורנבליט ומר לאוניד קוגל

משך המבחן: שלוש שעות

אין להשתמש בחומרי עזר, פרט, לדפים המצורפים לשאלון

שאלה 1 (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 החסכונית 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.

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 ותחזיר את הערך של a_{ij} .

א.

```
#define MAX 5 //example

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

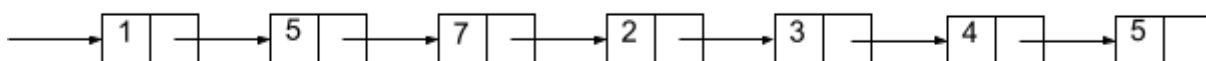
```
int sym_matr_element (int **a_dyn, int i, int j)
{
    if (i>=j)
        return a_dyn[i][j];
    else
        return a_dyn[j][i];
}
```

שאלה 2 (28 נק')

קטע סגור **[a, b]** הוא קבוצת כל הנקודות על ישר המספרים אשר נמצאות בין שתי הנקודות **a** ו-**b** (הנקראות קצוות הקטע) כולל **a** ו-**b** עצמם.

כתוב פונקציה המקבלת רשימה מקושרת של מספרים שלמים וגם שני מספרים שלמים **a** ו-**b** שהם קצוות של הקטע הסגור. על הפונקציה למצוא את איברי הרשימה הנמצאים בקטע **[a, b]** ולהעתיק אותם למערך דינאמי **A**. הפונקציה תעביר by reference את הכתובת של **A** ותחזיר את גודלו. במידה ו-**A** לא מכיל אף איבר, כתובתו תהיה NULL וגודלו יהיה 0.

לדוגמא, עבור הרשימה הבאה:



ומספרים **a** השווה ל-3 ו-**b** השווה ל-6

הפונקציה תיצור את המערך הבא:

5	3	4	5
---	---	---	---

ותחזיר 4.

יש להגדיר את הטיפוס של איברי הרשימה המקורית מעל הפונקציה.

ניתן להניח שיש בזיכרון מספיק מקום להקצאת המערך.

```
typedef struct item {
```

```

    int data;
    struct item *next;
} Item;

int closed_interval (Item *lst, int **arr, int a, int b)
{
    int size=0, i=0;
    Item *ptr = lst;
    /* Estimation of array's size */
    while (ptr)
    {
        if (ptr->data >= a && ptr->data <= b)
            size++;
        ptr = ptr->next;
    }
    if (size == 0)
    {
        *arr = NULL;
        return 0;
    }
    /* Allocation of array */
    *arr = (int *)calloc(size, sizeof(int));
    /* Filling of array */
    ptr = lst;
    while (ptr)
    {
        if (ptr->data >= a && ptr->data <= b)
        {
            (*arr)[i] = ptr->data;
            i++;
        }
        ptr = ptr->next;
    }
    return size;
}

```

שאלה 3 (28 נק')

בקובץ טקסט נמצא מידע עבור לקוחות בייצוג הבא:

- מס' ת.ז. (9 תווים),
- שם לקוח (30 תווים),
- סכום סה"כ קניות (מספר מסוג double שתופס 8 עמדות).

כתוב פונקציה אשר מקבלת מס' ת.ז. של הלקוח, סכום הקנייה החדשה שלו, ומחרוזת המכילה בתוכה את שם הקובץ. על הפונקציה לעדכן מידע עבור הלקוח בקובץ ע"י חיבור סכום הקנייה החדשה לסכום סה"כ הקניות של הלקוח.

לדוגמא, עבור הקובץ בעל התוכן הבא:

06842151	Moshe	71.25
330578795	Benjamin	1755.65
20051546	Dina	560.10

ומס' ת.ז. "330578795" וסכום הקנייה החדשה ששווה ל-200.1 התוכן המעודכן של הקובץ אחרי הקריאה לפונקציה יהיה:

06842151	Moshe	71.25
----------	-------	-------

330578795
20051546

Benjamin 1955.75
Dina 560.10

הפונקציה תחזיר אמת אם הפעולה בוצע בהצלחה. הפונקציה תחזיר שקר אם מס' ת.ז. הלקוח לא נמצא בקובץ או אם הקובץ לא נפתח.

לטיפול בנתוני הקובץ ניתן להשתמש בפורמטים הבאים: "%9s", "%30s", "%8.2lf".

שים לב: פונקציה fscanf מתעלמת מהרווחים השמאליים של המחרוזת הנקלטת ומתחילה את הקלט מהתו הראשון שאחריהם.

```
int UpdateItem(char *fName, char *idClient, double addTotal)
{
    char id[10], name[31];
    double total;
    FILE *f=fopen(fName, "r+");
    if( !f )
    {
        printf("File not found!\n");
        return 0;
    }
    fscanf(f,"%9s%30s%lf", id, name, &total);
    while (!feof(f))
    {
        if( strcmp( idClient, id ) == 0) // id equals idClient
        {
            total+=addTotal;
            // 8 positions to the left to beginning of the number
            fseek(f, -8, SEEK_CUR);
            fprintf(f,"%8.2lf\n", total);
            fclose(f);
            return 1;
        }
        fscanf(f,"%9s%30s%lf", id, name, &total);
    }
    fclose(f);
    return 0;
}
```

שאלה 4 (16 נק')

נתונות שתי פונקציות הבאות:

```
unsigned fun1( unsigned value )
{
    unsigned c, displayMask = 1, x=0;
    for ( c = 1; c <= 32; c++ )
    {
        x ^= value & displayMask ? 1 : 0;
        displayMask <<= 1;
    }
    return x;
}

unsigned fun2( unsigned value )
```

```

{
    unsigned c, displayMask = 1, x=0;
    for ( c = 1; c <= 32; c++ )
    {
        if( value & displayMask )
            x++;
        displayMask <<= 1;
    }
    return x & 1;
}

```

יש להניח שכל המספרים מסוג unsigned הם 32 סיביות.

1. האם שתי הפונקציות מבצעות אותה משימה בדרכים שונות או הן מבצעות משימות שונות?

שתי הפונקציות מבצעות אותה משימה בדרכים שונות.

2. אם הן מבצעות אותה משימה, מה הן מבצעות ומה ההבדל בין הדרכים שהן משתמשות?
אם הן מבצעות משימות שונות, מה מבצעת fun1 ומה מבצעת fun2?

הפונקציות בודקות האם כמות סיביות 1 בייצוג הבינארי של מספר value היא אי-זוגית ומחזירות 1 (אמת) אם היא אי-זוגית ו-0 (שקר) אם היא זוגית.

fun1 סורקת את הייצוג הבינארי של value כך שכל סיבית 1 חדשה הופכת את תשובת הביניים מ-0 ל-1 ומ-1 ל-0.

fun2 סופרת את כמות x של הסיביות 1 בייצוג הבינארי של value ומגלה את הסיבית הימנית ב-x. אם הסיבית היא 1 אז x הוא מספר אי-זוגי והפונקציה מחזירה 1. אם הסיבית היא 0 אז x הוא מספר זוגי והפונקציה מחזירה 0.

שאלה 5 (בנוס – 10 נק')

מהו הפלט של התוכנית הבאה?

```

#include <stdio.h>

void main()
{
    int d=1;
    printf("%d", (int) "printf("%d)");
}

```

0