

**COMPUTER SCIENCE DEPARTMENT המחלקה למדעי המחשב**

**סדנה מתקדמת בתכנות 61108**

סמסטר ב' תשע"ז

מועד א'

26.06.2017

**מרצים: ד"ר מרק קורנבליט ומר מרדכי חגיז**

משך המבחן: שלוש שעות

אין להשתמש בחומרי עזר, פרט, לדף המצורף לשאלון

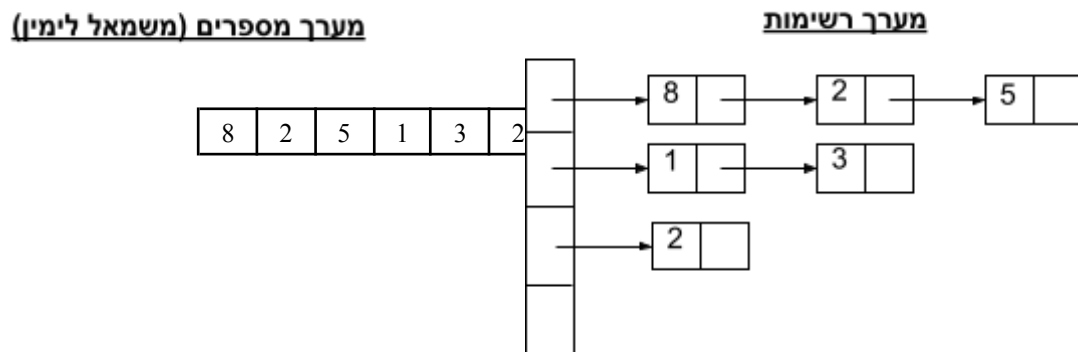
**שאלה 1 (28 נק')**

כתוב פונקציה המקבלת כפרמטרים **מערך רשימות מקושרות** של מספרים שלמים וגם **כמות רשימות** בתוכן.

על הפונקציה לבנות **מערך מספרים שלמים** חדש אשר מכיל בתוכו את כל האיברים של רשימות המערך המקורי (מהרשימה הראשונה עד לרשימה האחרונה, מהאיבר הראשון בכל רשימה עד לאיבר האחרון בתוכה).

הפונקציה תחזיר **כתובת** של המערך החדש ותעביר by reference את **גודלו**.

**דוגמא:**



הפונקציה תעביר **6** בדוגמא.

יש להגדיר את **הטיפוס של איברי הרשימה** לפני הפונקציה.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

```
typedef struct list_item
{
    int data;
    struct list_item *next;
} ListItem;

int *arr_of_lists_to_arr (ListItem **arr_of_lists,
                          int size_arr_of_lists, int *arr_size)
{
    int i, j=0, *arr;
    ListItem *curr_ptr;
    /* Evaluation of array's size */
    *arr_size = 0;
    for (i=0; i<size_arr_of_lists; i++)
    {
        curr_ptr = arr_of_lists[i];
        while (curr_ptr)
        {
            (*arr_size)++;
            curr_ptr = curr_ptr->next;
        }
    }
    if (*arr_size == 0)
        return NULL;
    /* Allocation of array */
    arr = (int *)malloc(*arr_size*sizeof(int));
    /* Filling of array */
    for (i=0; i<size_arr_of_lists; i++)
    {
        curr_ptr = arr_of_lists[i];
        while (curr_ptr)
        {
            arr[j++] = curr_ptr->data;
            curr_ptr = curr_ptr->next;
        }
    }
    return arr;
}
```

**שאלה 2 (28 נק')**

כתוב פונקציה אשר מקבלת מערך דו-ממדי סטטי (לא דינאמי)  $A$  של מספרים שלמים בעל  $n$  שורות ו- $m$  עמודות. המספר הפיסי של העמודות יינתן ע"י קבוע בשם COLS. יש להגדיר אותו לפני הפונקציה. ידוע שכל שורה ב- $A$  היא מערך ממוין בסדר עולה.

על הפונקציה לבנות מערך דו-ממדי דינאמי  $B$  חדש בעל  $n$  שורות עם גדלים שונים. איברי  $B$  יהיו מוגדרים באופן הבא:

```
typedef struct item {
    int data;
    int count;
} Item;
```

מערך  $B$  יישם את  $A$  בדרך חסכונית המאפשרת להציג את  $A$  ללא איברים חוזרים בשורות באופן הבא:

כל סדרת איברים זהים בשורה מס'  $i$  של  $A$  תוחלף באיבר בודד בשורה מס'  $i$  של  $B$ . שדה data של איבר ב- $B$  יהיה שווה לערך איברים זהים בסדרה התואמת ושדה count יציג את מספר האיברים הזהים בסדרה. איברים בכל שורה של  $B$  יהיו ממויינים בסדר עולה על פי הערך של data.

הפונקציה תיצור גם מערך נוסף חדש ששמו **SIZE** המורכב מגדלי שורות של מערך  $B$  כך שאיבר מס'  $i$  של **SIZE** יהיה שווה לגודל שורה מס'  $i$  של  $B$ .

**דוגמא:**

<u>SIZE</u>	<u>B</u>	<u>A</u>
4	{2,1} {3,2} {6,4} {9,1}	2 3 3 6 6 6 6 9
3	{0,4} {1,2} {7,2}	0 0 0 0 1 1 7 7
1	{5,8}	5 5 5 5 5 5 5 5
2	{3,3} {4,5}	3 3 3 4 4 4 4 4

כל מבנה Item הוצג כזוג מספרים בסוגריים מסולסלים (המספר השמאלי הוא data והמספר הימני הוא count).

על הפונקציה להחזיר את הכתובת של מערך  $B$  ולהעביר by reference את הכתובת של מערך **SIZE**.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.



מכון טכנולוגי חולון  
Holon Institute of Technology

```
#define COLS 8 //example

Item **compress_matrix_to_dyn_array (int A[][COLS], int rows, int cols,
                                     int **pSIZE)
{
    int i, j, ser_size, ser_num;
    Item **B = (Item **)malloc(rows*sizeof(Item *));
    *pSIZE = (int *)malloc(rows*sizeof(int));
    for (i=0; i<rows; i++)
    {
        ser_size = 1; //number of equal elements in the current
series
        ser_num = 0; //number of series of equal elements
        B[i] = (Item *)malloc(cols*sizeof(Item ));
        for (j=1; j<cols; j++)
            if (A[i][j] == A[i][j-1]) //the same series
                ser_size++;
            else //new series
            {
                B[i][ser_num].count = ser_size;
                B[i][ser_num++].data = A[i][j-1];
                ser_size = 1;
            }
        /* last series in row i */
        B[i][ser_num].count = ser_size;
        B[i][ser_num++].data = A[i][j-1];

        B[i] = (Item *)realloc(B[i], ser_num*sizeof(Item));
        (*pSIZE)[i] = ser_num;
    }
    return B;
}
```

### שאלה 3 (28 נק')

כתוב פונקציה (void) שמקבלת אות קטנה בשם **letter**, מחרוזת **str** ומחרוזת **fileName**. המחרוזת **str** כוללת מספר **מילים** (רצפים של אותיות קטנות וגדולות) המופרדות ברווח אחד או יותר.

על הפונקציה ליצור **קובץ טקסט** ששמו נכלל ב- **fileName** ולהעתיק את כל המילים ב- **str** שמתחילות באות **letter** או באות הגדולה התואמת ל- **letter** לשורה נפרדת בקובץ.

שים לב: המחרוזת **str** יכולה להתחיל או להסתיים לא רק עם אותיות אלא גם עם רווחים.

דוגמא:

עבור **str** "Rony Goodman got a good mark" ו- **letter**='g' תוכן הקובץ אחרי קריאה לפונקציה יהיה

Goodman

גולומב 52, ת.ד. 305, חולון 5810201  
טלפקס: 03-5026528

52 Golomb St., Holon 5810201 Israel

**www.hit.ac.il** Tel/Fax: 972-3-502-6528

הפקולטה למדעים  
המחלקה למדעי המחשב

Faculty of Sciences  
Department of Computer Science



מכון טכנולוגי חולון  
Holon Institute of Technology

got  
good

```
void words_to_file (char *str, char letter, char *fileName)
{
    int i, in=0;
    FILE *fptr = fopen(fileName, "w");
    if (!fptr)
        exit(1);
    if (*str == '\0')
    {
        fclose(fptr);
        return;
    }
    if (*str == letter || *str == toupper(letter))
    {
        in = 1; //mode: copying appropriate word
        fputc(str[0], fptr);
    }
    for (i=1; str[i]!='\0'; i++)
        if (in) //in copying appropriate word
            if (str[i] == ' ') //end of appropriate word
            {
                fputc('\n', fptr);
                in = 0; //mode: out of copying appropriate word
            }
            else //continuation of appropriate word
                fputc(str[i], fptr);
        else //out of copying appropriate word
            if (str[i-1] == ' ' && (str[i] == letter || str[i] ==
toupper(letter))) //beginning of appropriate word
            {
                in = 1; //mode: copying appropriate word
                fputc(str[i], fptr);
            }
        if (in)
            fputc('\n', fptr);
    fclose(fptr);
}
```

גולומב 52, ת.ד. 305, חולון 5810201  
טלפקס: 03-5026528

52 Golomb St., Holon 5810201 Israel

**www.hit.ac.il** Tel/Fax: 972-3-502-6528

הפקולטה למדעים  
המחלקה למדעי המחשב

Faculty of Sciences

Department of Computer Science

**שאלה 4 (16 נק')**

נתונה התוכנית הבאה:

```
#include <stdio.h>
#define SIZE 5

unsigned fun();
void sod(unsigned *, unsigned);

void main()
{
    unsigned arr[SIZE], i;
    sod(arr, SIZE);
    for (i=0; i<SIZE; i++)
        printf("%u ", arr[i]);
}

unsigned fun()
{
    static unsigned value = 0;
    value = !value ? 1 : value<<1;
    value = !value ? 1 : value;
    return value;
}

void sod(unsigned *arr, unsigned n)
{
    unsigned i;
    for(i=0; i<n; i++)
        arr[i] = fun();
}
```

1. מהו פלט התוכנית ?

**1 2 4 8 16**

2. מה מבצעת הפונקציה **sod** עבור **n** שאינו עולה על 32 (גודל של טיפוס **unsigned**) ?

ממלאת מערך **arr** כך שאיבר **i** שלו ( $i \geq 0$ ) יהיה שווה ל-  $2^i$ .

3. מה מבצעת הפונקציה **sod** במקרה הכללי (עבור **n** שרירותי) ?

ממלאת מערך **arr** כך שאיבר **i** שלו ( $i \geq 0$ ) יהיה שווה ל-  $2^{i \bmod 32}$ .

4. האם יהיה תקין מבחינת המהדר (הקומפיילר) למחוק את המילה **static** בפונקציה **fun**? נמק אם

לא, הראה איך ישתנה הפלט אם כן.

תקין. הפלט יהיה

**1 1 1 1 1**

שאלה 5 (בנוס – 10 נק')

מהו הפלט של התוכנית הבאה?

```
#include <stdio.h>

void main()
{
    char *s="WOW!";
    puts( (char *) ( ((long long)s*(long long)s-4)/(long long)(s-2) ) );
}
```

יש לנמק את התשובה.

**!W**

הסבר:

Casting ל- long long ממיר כתובת (ערך s) למספר שלם.  
מקבלים

$$(s^2-4)/(s-2)=(s-2)(s+2)/(s-2)=s+2$$

Casting ל- char \* קובע ערך s+2 ככתובת של תו מס' 2 במחרוזת.  
פלט מתו מס' 2 עד סוף המחרוזת:

**W!**