

מבחן סדנה מתקדמת בתכנות-61108

ד"ר מרק קורנבליט וד"ר נעמה קופלמן

סמסטר ב', מועד א', תשע"ט

תאריך: 10.07.19

הוראות:

- משך המבחן 3 שעות.
- אין להשתמש בחומרי עזר.

שאלה 1 (28 נק')

כתוב פונקציה אשר מקבלת מערך מחרוזות **str**, גודלו **n** וגם מחרוזת **fileName**.
תו מס' 0 של מחרוזת **[str[i]** הוא ספרה בעלת ערך **i+1** (ניתן להניח ש-**n** לא עולה על 9).

על הפונקציה ליצור קובץ טקסט ששמו נכלל ב- **fileName** ולהעתיק את תוכן המערך לקובץ באופן מחזורי כך שמחרוזת **[str[1]** תהיה שורה ראשונה בקובץ, מחרוזת **[str[2]** תהיה שורה שנייה בקובץ, ..., מחרוזת **[str[n-1]** תהיה שורה שלפני אחרונה בקובץ ומחרוזת **[str[0]** תהיה שורה אחרונה בקובץ.
יש לעדכן תו ראשון בתחילת כל שורה כך ששורות כמו מחרוזות המערך יהיו ממוספרות מ-1 עד **n**.

דוגמא:

תוכן הקובץ

1 Ron Smith, 367
2 Arthur Mitchell, 84
3 Eduard Stone, 17
4 Bill Johns, 28

str

1 Bill Johns, 28
2 Ron Smith, 367
3 Arthur Mitchell, 84
4 Eduard Stone, 17

```
void cyclic_shift_to_file (char **str, int n, char *fileName)
{
    int i;
    FILE *fptr = fopen(fileName, "w");
    if (!fptr)
        exit(1);
    for (i=1; i<n; i++)
    {
        fputc(i+'0', fptr);
        fputs(str[i]+1, fptr);
        fputc('\n', fptr);
    }
    fputc(n+'0', fptr);
    fputs(str[0]+1, fptr);
    fputc('\n', fptr);
    fclose(fptr);
}
```



מכון טכנולוגי חולון
Holon Institute of Technology

שאלה 2 (28 נק')

נתונות ההגדרות הבאות של טיפוסים:

```
typedef struct data_item {  
    int data;  
    struct data_item *next;  
} Datalitem;
```

```
typedef struct ptr_item {  
    Datalitem *ptr;  
    struct ptr_item *next;  
} Ptritem;
```

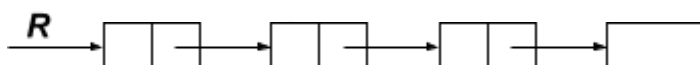
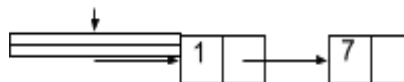
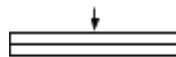
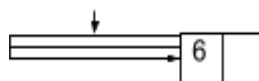
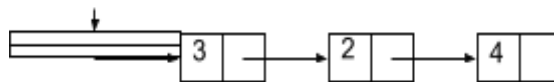
כתוב פונקציה המקבלת רשימה מקושרת L של רשימות מקושרות אשר מורכבת מאיברים מסוג **Ptritem**. השדה **ptr** של איבר מס' i ב- L יצביע לראש רשימת המספרים מס' i ושדה ה- **next** שלו יצביע לאיבר הבא ב- L . רשימות מספרים יהיו מורכבות מאיברים מסוג **Datalitem**.

על הפונקציה לבנות רשימה מקושרת חדשה R אשר מורכבת מאיברים מסוג **Datalitem**. כמות האיברים ב- R תהיה שווה לכמות הרשימות ב- L . שדה **data** של איבר מס' i ב- R יהיה שווה לסכום שדות **data** של איברי הרשימה מס' i של L .

הפונקציה תחזיר את כתובת האיבר הראשון של רשימת R .

L

דוגמא:



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

ניתן להניח שיש בזיכרון מספיק מקום להקצאות.

```
DataItem *list_of_lists_to_list (PtrItem *L)
{
    DataItem *data_item_list, *curr_ptr,
        *lst = (DataItem *)malloc(sizeof(DataItem)); //dummy
    PtrItem *temp = L;

    curr_ptr = lst;
    while (temp) // scan list of lists
    {
        data_item_list = temp->ptr;
        curr_ptr->next = (DataItem *)malloc(sizeof(DataItem));
        curr_ptr = curr_ptr->next;
        curr_ptr->data = 0;
        while (data_item_list) // scan internal list
        {
            curr_ptr->data += data_item_list->data;
            data_item_list = data_item_list->next;
        }
        temp = temp->next;
    }
    curr_ptr->next = NULL;
    /* deletion of dummy */
    curr_ptr = lst;
    lst = lst->next;
    free (curr_ptr);
    return lst;
}
```

שאלה 3 (28 נק')

כתוב פונקציה אשר מקבלת **מערך דו-ממדי סטטי** (לא דינאמי) של מספרים שלמים בעל **rows** שורות ו- **cols** עמודות. המספר הפיסי של העמודות יינתן ע"י קבוע בשם **COLS**. יש להגדיר אותו לפני הפונקציה.

על הפונקציה ליצור מערך דו-ממדי דינאמי **B** אשר יהיה מורכב אך ורק מהשורות של **A** שסכומי האיברים שלהן הם מספרים זוגיים. עם זאת רק איברים אי-זוגיים מועתקים מ- **A** ל- **B**. יש לשמור את סדר השורות מעותקות מ- **A** ל- **B** וגם את סדר האיברים מעותקים מ- **A** ל- **B** בכל שורה. אם סכום האיברים בשורה של **A** הוא מספר זוגי אבל גם כל האיברים בשורה הם זוגיים אז לא להעתיק את השורה מ- **A** ל- **B**.

הפונקציה תחזיר את הכתובת של מערך **B** ותעביר by reference את מספר השורות בתוכו. בנוסף הפונקציה תעביר by reference מערך דינאמי **S** המורכב מגדלי השורות של מערך **B**. איבר מס' *i* של מערך **S** יהיה שווה לגודל שורה מס' *i* של **B**.

ניתן להניח שיש בזיכרון מספיק מקום להקצאות.

<u>S</u>	<u>B</u>	<u>A</u>
2	13 33	20 14 10 13 33 80 sum=170
4	15 17 19 21	40 10 55 11 12 13 sum=141
6	41 43 53 41 17 15	15 16 17 18 19 21 sum=106
		41 43 53 41 17 15 sum=210
		10 20 30 40 50 62 sum=212

הפונקציה גם תעביר 3 by reference כמספר השורות במערך **B**.

```
#define COLS 6 //example

int **even_sums_odd_numbers (int A[][COLS], int rows, int cols, int **S,
int *prows)
{
    int i, j, k, sum, odd_count, **B;
    *prows = 0;
    B = (int **)calloc(rows, sizeof(int *));
    *S = (int *)calloc(rows, sizeof(int));
    for (i=0; i<rows; i++)
    {
        sum = odd_count = k = 0;
        for (j=0; j<cols; j++)
        {
            sum += A[i][j];
            odd_count += A[i][j]%2;
        }
        if (sum%2 == 0 && odd_count>0)
        {
            B[*prows] = (int *)calloc(odd_count, sizeof(int));
            for (j=0; j<cols; j++)
                if (A[i][j]%2 == 1)
                    B[*prows][k++] = A[i][j];
            (*S)[*prows] = odd_count;
            (*prows)++;
        }
    }
    B = (int **)realloc(B, *prows*sizeof(int *));
    *S = (int *)realloc(*S, *prows*sizeof(int));
    return B;
}
```

שאלה 4 (16 נק')

בחר את התשובה הנכונה ונמק בכל אחד מהסעיפים הבאים (חובה לרשום את הבחירה והנימוק במחברת):

1.

```
int (*p) (int *);
```

מה זה p?

- (1) מצביע לפונקציה שמקבלת מצביע ל-int ומחזירה int
- (2) פונקציה שמקבלת מצביע ל-int ומחזירה מצביע ל-int
- (3) מצביע לפונקציה שמקבלת מצביע ל-int ומחזירה מצביע ל-int
- (4) אף תשובה אינה נכונה

(1) – לפי התחביר של שפת C

2.

```
z = x^y;
```

מה יהיה ערך של z?

- (1) x בחזקת y
- (2) המספר המתקבל מהזזת y סיביות שמאלה ב-x
- (3) המספר המתקבל מהזזת y סיביות ימינה ב-x
- (4) אף תשובה אינה נכונה

(4) – כי $z = x \oplus y$

3.

```
static int c;
```

מה הוא ערך התחלתי של c?

- (1) c לא מאותחל
- (2) 0
- (3) 1
- (4) אף תשובה אינה נכונה

(2) – כי ברירת המחדל ערך התחלתי של משתנה static הוא 0

4.

```
const int* q;
```

מה הוא q?

- (1) מצביע קבוע למשתנה
- (2) מצביע שיכול להשתנות לקבוע
- (3) מצביע קבוע לקבוע
- (4) אף תשובה אינה נכונה

(2) – לפי התחביר של שפת C

שאלה 5 (בונוס – 10 נק')

מהו הפלט של התוכנית הבאה?

```
#include <stdio.h>
void main()
{
    int i;
    for (i=0; i<sizeof"sizeof"[i]; i++)
        printf("%d",sizeof"sizeof"[i]);
}
```

יש לנמק את התשובה.

1

sizeof"sizeof" זה גודל של תו מס' i במחרוזת "sizeof". גודל של כל תו הוא 1 (בית אחד). לכן באיטציה הראשונה i=0 והתנאי 1>0 מתקיים. נכנסים לגוף הלולאה ומדפיסים 1. באיטציה השנייה i=1 והתנאי 1>1 לא מתקיים. יוצאים מהלולאה.