

מבחן סדנה מתקדמת בתכנות-61108

ד"ר מרק קורנבליט, ד"ר לאוניד קוגל, מר טוביה דב רוזנברג, מר דוד אלגרלי

סמסטר ב', מועד ב', תשפ"א

תאריך: 13.07.21

הוראות:

- משך המבחן 3 שעות.
- אין להשתמש בחומרי עזר, פרט לדף המצורף לשאלון.

חלק 1. בחלק זה יש לענות על שתי שאלות מתוך שלוש.

שאלה 1 (40 נק')

כתוב פונקציה אשר מקבלת מחרוזות `infile_1`, `infile_2` ו-`outfile` המהוות שמות של שלושה קבצי טקסט.

תוכן של כל שורה בקבצים `infile_1` ו-`infile_2` הוא רצף של מספרים שלמים **ממוינים** (מתחילת השורה עד סופה) מופרדים ברווחים. התו של סוף השורה מופיע מיד אחרי המספר האחרון בכל שורה, כולל השורה האחרונה.

על הפונקציה למלא קובץ `outfile` כך שתוכן שורה מס' `i` בתוכו יהיה רצף **ממוין** (מתחילת השורה עד סופה) של כל המספרים ששייכים לשורות מס' `i` בקבצים `infile_1` ו-`infile_2`.
אם מספרי השורות בקבצים `infile_1` ו-`infile_2` הם שונים (נסמן אותם ב-`n` ו-`m` חסך שלמשל `m > n`) אז קובץ `outfile` יכלול `n` שורות כך ש-`m-n` השורות האחרונות יועתקו מהקובץ התואם.

דוגמא:

<code>outfile</code>	<code>infile_2</code>	<code>infile_1</code>
3 8 9 12 16 17 20	9 17	3 8 12 16 20
0 2 2 2 6 6 7 11	2 6 7 11	0 2 2 6
5 9 10 10 11 13 14 18 19 26	5 13 14 18	9 10 10 11 19 26
4 4 4 5		4 4 4 5

```
/* Auxiliary function. Copy a number from file to file
and input of a next number */
void next_number(FILE* in, FILE* out, int* num, char* ch)
{
    fprintf(out, "%d ", *num);
    if ((*ch = fgetc(in)) != '\n')
        fscanf(in, "%d", num);
}

/* Auxiliary function.
Merging 2 sorted rows of 2 files into a sorted row in a new file */
void rows_merge(FILE* in1, FILE* in2, FILE* out)
{
    char ch1=0, ch2=0;
    int num1, num2;

    /* Merging rows until one of the rows ends */
    fscanf(in1, "%d", &num1);
    fscanf(in2, "%d", &num2);
    while (ch1 != '\n' && ch2 != '\n')
    {
        if (num1 < num2)
            next_number(in1, out, &num1, &ch1);
        else
            next_number(in2, out, &num2, &ch2);

        /* Copying remainder of a longer row */
        while (ch1 != '\n')
            next_number(in1, out, &num1, &ch1);
        while (ch2 != '\n')
            next_number(in2, out, &num2, &ch2);

        fputc('\n', out);
    }

    /* Auxiliary function */
    void copy_file(FILE* in, FILE* out)
    {
        char ch;
        while ((ch = fgetc(in)) != EOF)
            fputc(ch, out);
    }
}
```

```
void filesMerge(char* infile1, char* infile2, char* outfile)
{
    FILE* in1 = fopen(infile1, "r");
    FILE* in2 = fopen(infile2, "r");
    FILE* out = fopen(outfile, "w");
    if (!in1 || !in2 || !out)
        exit(1);

    /* Merging file rows until one of the files ends */
    while (fgetc(in1) != EOF && fgetc(in2) != EOF)
    {
        fseek(in1, -1, SEEK_CUR);
        fseek(in2, -1, SEEK_CUR);
        rows_merge(in1, in2, out);
    }

    /* Copying remainder of a larger file */
    fseek(in1, -1, SEEK_CUR);
    fseek(in2, -1, SEEK_CUR);
    copy_file(in1, out);
    copy_file(in2, out);

    fclose(in1);
    fclose(in2);
    fclose(out);
}
```

שאלה 2 (40 נק')

נתונה הגדרה הבאה של הטיפוס **person** אשר מהווה שם משפחה, שם פרטי ומס' זהות של בן-אדם:

```
typedef struct {
    char last_name[30];
    first_name[30];
    unsigned id;
} person;
```

כתוב פונקציה בוליאנית אשר מקבלת מערך איברים מסוג **person** וגודלו.
על הפונקציה להחזיר **1** אם המערך ממין בסדר עולה ולהחזיר **0** אחרת.

נגדיר מערך איברים מסוג **person** כממין בסדר עולה אם:

- איברי המערך ממוינים בסדר לקסיקוגרפי עולה ממש לפי שמות המשפחה (שדה **last_name**);
- כאשר בין איברי המערך יש איברים בעלי שדות **last_name** זהים, האיברים הללו ממוינים בסדר לקסיקוגרפי עולה ממש לפי שמות פרטיים (שדה **first_name**);



מכון טכנולוגי חולון Holon Institute of Technology

3. כאשר בין איברי המערך יש איברים בעלי שדות **last_name** זהים וגם שדות **first_name** זהים, האיברים הללו ממוינים בסדר עולה לפי מספרי הזהות (**id** שדה).

דוגמאות:

עבור המערך הבא

```
0 Asher Mike 231567809
1 Bern David 216578900
2 Borgman Joseph 354678903
3 Cohen Abraham 224567821
4 Cohen Den 206666431
5 Levi Ilan 290876441
6 Levi Omri 245689432
7 Levi Omri 245689436
8 Smith Jack 345389010
9 Smith John 311232111
```

הפונקציה תחזיר 1.

עבור המערך הבא

```
0 Asher Mike 231567809
1 Bern David 216578900
2 Borgman Joseph 354678903
3 Cohen Abraham 224567821
4 Cohen Den 206666431
5 Levi Ilan 290876441
6 Levi Omri 245689436
7 Levi Omri 245689432
8 Smith Jack 345389010
9 Smith John 311232111
```

הפונקציה תחזיר 0.

```
int ordered_person_array(person* person_arr, int n)
{
    int i, cmp;
    for (i = 1; i < n; i++)
        if ( (cmp = strcmp(person_arr[i].last_name, person_arr[i - 1].last_name)) < 0 )
            return 0;
    else
        if (cmp == 0)
            if ((cmp=strcmp(person_arr[i].first_name, person_arr[i-1].first_name))<0)
                return 0;
            else
                if (cmp == 0)
                    if (person_arr[i].id < person_arr[i - 1].id)
                        return 0;
    return 1;
}
```

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 3 (40 נק')

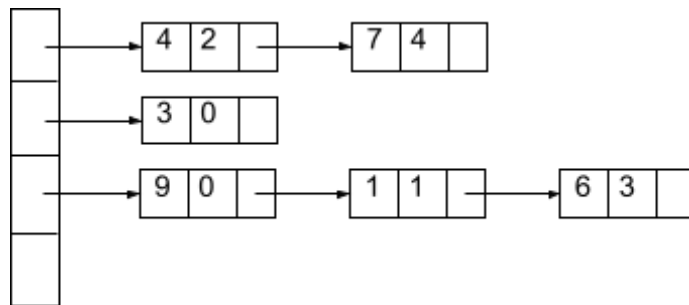
מטריצה דלילה הינה מטריצה שמרבית איבריה בעלי ערך אפס.

כתוב פונקציה אשר מקבלת מטריצה דלילה A המיושמת כמערך של רשימות מקושרות (היישום החסכוני, כאשר מופיעים ברשימה רק איברי המטריצה שאינם בעלי ערך אפס). רשימה מס' i של המערך תתאים לשורה מס' i של מטריצה A . איברי הרשימות יכילו ערך שהוא איבר המטריצה וגם ערך שהוא מס' העמודה בה הוא נמצא במטריצה. הפונקציה תקבל כפרמטרים את מערך הרשימות וגודלו וגם את מספר העמודות במטריצה.

על הפונקציה לשחזר את המטריצה כמערך דו-ממדי **דינאמי** שכל שורותיו באותו גודל. הפונקציה תחזיר כתובת של המערך הדו-ממדי הדינאמי הבנוי.

דוגמא:

$$\begin{bmatrix} 0 & 0 & 4 & 0 & 7 \\ 3 & 0 & 0 & 0 & 0 \\ 9 & 1 & 0 & 6 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



בדוגמא הנ"ל הפונקציה תקבל את A כמערך רשימות, את גודל המערך, 4, וגם את מספר העמודות ב- A .
5.

יש להגדיר את הטיפוס של איברי הרשימה מעל הפונקציה.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

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

int** lists_to_sparse_matrix(element** list_row, int rows, int cols)
{
    int** arr, i;
    element* ptr;
    arr = (int**)calloc(rows, sizeof(int*));
    for (i = 0; i < rows; i++)
        arr[i] = (int*)calloc(cols, sizeof(int));
    for (i = 0; i < rows; i++)
    {
        ptr = list_row[i];
        while (ptr)
        {
            arr[i][ptr->col] = ptr->data;
            ptr = ptr->ptr_next;
        }
    } // for i
    return arr;
}
```

חלק 2.

שאלה 4 (20 נק')

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

1. מהו פלט (משמאל לימין) של התוכנית הבאה?

```
#include <stdio.h>

int x;

void f() {
    static int x;
    printf("%d", x);
    printf("%d", ++x);
}

void main() {
    f();
    printf("%d", x);
    printf("%d", ++x);
    f();
}
```

010101 (1)

000000 (2)

000011 (3)

010112 (4)

011223 (5)

התוכנית אינה תקינה (6)

אף תשובה אינה נכונה (7)

כברירת מחדל ערך התחלתי של `x` static הוא 0. לכן בקריאה ראשונה של `f()` מדפיסים 0, מקדמים `x` ומדפיסים 1. אחרי מדפיסים ערך התחלתי של `x` גלובלי ששווה ל-0, מקדמים `x` גלובלי ומדפיסים 1. בקריאה שנייה של `f()` מדפיסים `x` static שערכו נישאר 1, מקדמים `x` ומדפיסים 2.

2. כמה פעמים יודפס Hi עקב ביצוע התוכנית הבאה?

```
#include <stdio.h>
void main() {
    int i = 64;
    for (; i; i >>= 1)
        printf("Hi \n");
}
```

(1) פעם אחת

(2) 7 פעמים

(3) 64 פעמים

(4) אף פעם

(5) תהיה לולאה אינסופית

(6) אף תשובה אינה נכונה

ייצוג בינארי של 64 הוא 1000000 (6 אפסים, 7 סיביות). כל פעולה $i \gg= 1$ מביאה להזזה של 1 בסיבית אחת ימינה. לכן תוך 7 בדיקות (6 הזזות) סיבית 1 תהיה עוד במספר i , ערך של i יהיה שונה מ-0 ותנאי המשך הלולאה יתקיים. אחרי הזזה שביעית 1 יצאה מייצוג בינארי של i אשר יהיה מורכב רק מאפסים, ולכן ערך עשרוני שלו יהיה 0 כך שתנאי המשך הלולאה בבדיקה שמינית לא יתקיים.

3. נתון פוינטר ל-`int p` שמצביע למערך דינאמי. למה יצביע `p` אחרי ביצוע פעולת

```
realloc(p, 0);
```

(1) לאותו מערך

(2) לאיזשהו מקום חדש

(3) `p` יהיה שווה ל-NULL

(4) הפעולה תגרום לשגיאת ריצה

(5) אף תשובה אינה נכונה

לפי כללי שפת C כאשר גודל חדש של בלוק זיכרון ב-`realloc` שווה ל-0, הפונקציה מחזירה NULL.

4. בתוכנית בה `a` מוגדר כמשתנה מסוג `int` ו-`fptr1` ו-`fptr2` הם פוינטרים לקבצים בינאריים קיימות שתי השורות הבאות:

```
fread(&a, sizeof(a), 1, fptr1);
fwrite(&a, sizeof(a), 1, fptr2);
```

מהי הטענה הנכונה?

(1) ברשימת פרמטרי `fread` בשורה הראשונה יש סימן & מיותר והשורה השנייה תקינה

(2) ברשימת פרמטרי `fwrite` בשורה השנייה יש סימן & מיותר והשורה הראשונה תקינה

(3) ברשימות פרמטרי `fread` ו-`fwrite` יש סימן & מיותר

(4) שתי השורות תקינות

(5) באחת מהשורות או בשתי השורות סימן & נמצא במקום לא נכון

(6) אף טענה אינה נכונה

גם `fread` וגם `fwrite` מקבלות כפרמטר מצביע לבלוק הזיכרון. לכן צריך להציב כתובת של המשתנה.

חלק 3.

שאלה 5 (בנוס – 10 נק')

מהו הפלט של התוכנית הבאה?

```
#include <stdio.h>
void main()
{
    printf("%d",printf("%d%d",printf("%d",11),printf("%d",11)));
}
```

יש לנמק את התשובה.

1111222

כל אחת מהפונקציות `printf("%d",11)` מדפיסה 11, סה"כ 1111. פונקציית `printf` מחזירה מספר בתיים כתובים. לפן כל אחת מהפונקציות הנ"ל מחזירה 2. אז `printf` שלישית (מימין) מדפיסה 22 (שני תווים) ומחזירה 2. לכן `printf` חיצונית (רביעית מימין) מדפיסה 2.