

מבחן 10010 שאלון Y

הוראות כלליות:

1. אין לשנות מבנים, הגדרות פונקציות .
2. יש לממש כל שאלה בקבצים השייכים לה בלבד, אין להוסיף קבצים לפרויקט
3. אין להשתמש במשתנים גלובלים, בפונקציה exit וב goto
4. אין להגדיר מטריצות בצורה דינאמית ללא הקצאה דינאמית.
5. יש להקפיד לסגור קבצים ולשחרר זיכרון בכל מקום רלוונטי.
6. בנושא bits אין להגדיר מבנה עם : וכמות Bits
7. כל סעיף במבחן נבדק באופן בלתי תלוי בסעיף אחר באותה שאלה. לכן ההמלצה לפתור כל סעיף גם אם לא הצלחתם לפתור את הסעיף שלפניו.
8. כל דבר שלא צוין שאסור אזי מותר

שאלה 1 (25 נקודות): (קבצים q1.c q1.h)

מוגדר המבנה הבא:

```
typedef struct
{
    int digit;
    int count;
} OCCUR;
```

סעיף א' (15 נקודות):

ממש/י פונקציה אשר מקבלת כקלט מספר שלם *num*. הפונקציה מחזירה כתובת תחילת מערך של איברים מהטיפוס OCCUR.

לכל ספרה אשר מופיעה במספר *num* קיים איבר במערך המוחזר. ב *digit* תשמר ערך הספרה וב *count* מספר מופיעה.

הפונקציה מחזירה את המערך ממוין בסדר עולה לפי הספרות (השדה *digit* ב-OCCUR). הפונקציה מעדכנת מצביע הניתן לה, *size*, במספר הספרות השונות ב-*num*.
דוגמא:

עבור *num* = 435536111 אזי יוחזר משמאל לימין:

(1,3), (3,2), (4,1), (5,2), (6,1)

(הספרה 1 מופיעה 3 פעמים וכדומה) ו- *size*=5.

חתימת הפונקציה:

OCCUR createOccurance(Long num, int *size);*

נתונה הפונקציה *Q1_A0* המאפשרת בדיקה של הפונקציה

סעיף ב' (10 נקודות):

ממש/י פונקציה אשר מקבלת כקלט מספר שלם *num*. הפונקציה מחזירה מספר אשר מורכב מספרות *num* בסדר עולה. יש להשתמש בפונקציה מהסעיף הקודם.
דוגמא:

עבור *num* = 435536111 אזי יוחזר 111334556

חתימת הפונקציה:

Long createSortedNum(Long num);

נתונה הפונקציה *Q1_B0* המאפשרת בדיקה של הפונקציה

שאלה 2 (35 נקודות): (קבצים q2.c q2.h)

סעיף א' (20 נקודות):

ממשי פונקציה אשר מקבלת כקלט מחרוזת אחת, *str*, המורכבת רק מאותיות בא"ב האנגלי (upper case ו-lower case). הפונקציה יוצרת מערך חדש *out* של מחרוזות. כל מחרוזת מורכבת רק מאותיות ב-lower case או רק מאותיות UPPER CASE אשר הופיעו ברצף במחרוזת.

לדוגמא:

- עבור מחרוזת "aaaBBB" אזי `out = {"aaa", "BBB"}`
- עבור מחרוזת "xYz" אזי `out = {"x", "Y", "z"}`
- עבור מחרוזת "CCC" אזי `out = {"CCC"}`

הפונקציה מחזירה את כתובת תחילת המערך *out* וכן מעדכנת מצביע שניתן לה, *pCount*, במספר האיברים האקטואליים במערך. כל איבר ב-*out* הוא מחרוזת. ניתן להניח שכל רצף אותיות לא יותר גדול מ-255 אך המחרוזות במערך *out* הן מחרוזות בגודל המדויק לא הנחת גודל.

חתימת הפונקציה:

```
char** splitOneString(int* pCount, const char* str);
```

הפונקציה מקבלת את הנתונים הבאים:

1. `int* pCount` – מצביע לעדכון מספר המחרוזות במערך *out*.
2. `str` – מחרוזת הקלט

נתונה הפונקציה *Q2_A0* המאפשרת בדיקה של הפונקציה

סעיף ב' (15 נקודות):

(בעזרת הפונקציה שמימשת בסעיף א').

ממש/י פונקציה אשר מקבלת מספר לא ידוע של מחרוזות, מחרוזת אחר מחרוזת. סיום הרשימה ב NULL. ידוע כי כל מחרוזות מורכבת רק מאותיות בא"ב האנגלי (lower case ו- upper case). הפונקציה יוצרת מערך חדש **outFull** של מחרוזות. שהוא האיחוד של כל מערכי המחרוזות המתקבלים מהפעלת הפונקציה מסעיף א'. ניתן להניח שמספר המערכים לא גדול מ 10 אך **outFull**, המערך המוחזר צריך להיות בגודל המדויק ללא הנחת גודל.

לדוגמא

- עבור קלט: "aaaBBB", NULL אזי **outFull** = {"aaa", "BBB"} - 2 מחרוזות
- עבור קלט: "aaaBBB", "xYz", NULL אזי **outFull** = {"aaa", "BBB", "x", "Y", "z"} - 5 מחרוזות

הפונקציה מחזירה את כתובת תחילת המערך **outFull** וכן מעדכנת מצביע שניתן לה, **pCount**, במספר האיברים האקטואליים במערך. כל איבר ב-**outFull** הוא מחרוזת. ניתן להניח שכל רצף אותיות לא יותר גדול מ 255 אך המחרוזות במערך **outFull** הן מחרוזות בגודל המדויק ללא הנחת גודל.

חתימת הפונקציה:

```
char** splitUpperLower(int* pCount, const char *str1, ...);
```

הפונקציה מקבלת את הנתונים הבאים:

1. **int*** pCount – מצביע לעדכון מספר המחרוזות במערך **outFull**.
2. **str1** – מחרוזת הקלט
3. מחרוזות נוספות עד לסיום הרשימה ב NULL

נתונה הפונקציה **Q2_B0** המאפשרת בדיקה של הפונקציה

שאלה 3 (40 נקודות): (קבצים q3.c q3.h)

נתון קובץ בינארי. בקובץ ישנם *num* מערכים של ספרות. פורמט הקובץ:

- מספר המציין את מספר המערכים *num*.
 - עבור כל מערך:
 - מספר שלם המציין את מספר הספרות במערך, *size*.
 - לאחר מכן רצף באורך *size* של ספרות המייצגות את המערך עצמו.
- ידוע שבכל מערך יש לפחות סיפרה אחת.

לדוגמא:

46153274798765434123514

אם בקובץ הבינארי רשום:

אזי בקובץ ישנם 4 מערכים, המערך הראשון באורך 6 ספרות והוא 15324, המערך השני באורך 7 ספרות והוא 9876543, המערך השלישי באורך 4 ספרות והוא 1235, והמערך הרביעי הוא בן ספרה אחת - 4.

הנתונים בקובץ הבינארי דחוסים תוך שימוש בהנחות הבאות:

1. מספר המערכים המקסימאלי יהיה 255 ולכן Byte יספיק.
2. מספר הספרות המקסימאלי בכל מערך יהיה 15 ולכן 4 ביטים (bits) יספיקו.
3. כיוון שכל סיפרה יכולה להיות בין 0 ל 9 אזי 4 ביטים (bits) יספיקו לשמירת הספרה.

- ב Byte הראשון יהיו רשומים מספר המערכים
- לאחר מכן עבור כל מערך יהיו הנתונים הבאים:
- ב Byte הראשון יופיע מספר הספרות (4 ביטים, bits, גבוהים (שמאליים)) והמספר הראשון במערך (4 ביטים, bits, תחתונים (ימניים))
- לאחר מכן בכל Byte יופיעו 2 ספרות של המערך. אם תישאר ספרה אחרונה בודדת היא תשמר ב Byte נפרד ב 4 הביטים, bits, גבוהים, (שמאליים).

Num of arrays	arr1 length	arr1[0]	arr1[1]	arr1[2]
1 Byte	4 bits	4 bits	4 bits	4 bits	4 bits

Byte Byte

סעיף א' (20 נקודות):

ממש/י פונקציה, **createArr**, המקבלת כקלט שם של קובץ בינארי. הפונקציה מחזירה כתובת תחילת מערך המכיל **num** איברים, ובו כל איבר הוא כתובת תחילת מערך של ספרות, כאשר האיבר הראשון במערך הספרות יציין כמה ספרות יש בו. כמו כן הפונקציה תעדכן מצביע שניתן לה במספר מערכי הספרות הקיימים בקובץ.

עבור הדוגמא הקודמת, התכנית תייצר מערך בו ימצאו כתובות תחילת ארבעת המערכים הבאים:

המערך 1: [6, 1, 5, 3, 2, 7, 4]

המערך 2: [7, 9, 8, 7, 6, 5, 4, 3]

המערך 3: [4, 1, 2, 3, 5]

המערך 4: [1, 4]

חתימת הפונקציה:

```
int** createArr(const char* filename, int* pSize);
```

הפונקציה מקבלת את הנתונים הבאים:

1. filename – שם הקובץ בו נתוני הספרות. הקובץ נכתב בפורמט בינארי המפורט למעלה.
2. int* pSize – מצביע לעדכון מספר המערכים בקובץ.

שם הקובץ הבינארי שנמצא בתיקייה הוא: **"numbers_comp.bin"**

נמצא גם בתיקייה קובץ בו האינפורמציה לא דחוסה בשם **"numbers.bin"** בו הנתונים נשמרים באותו סדר כ **int** עבור כל מספר. ניתן לעבוד עם הקובץ הזה בהורדה של **10** נקודות.

נתונה הפונקציה **Q3_A0** המאפשרת בדיקה של הפונקציה

סעיף ב' (20 נקודות):

נניח כי קיימת פונקציה כללית בשם *isGood* המקבלת מצביע לאיבר כללי ומחזירה 0 או 1 על פי הערך שהתקבל ולוגיקה מסוימת הממומשת בפונקציה.

חתימת הפונקציה: `int isGood(void* element)`

עבור מערך ספרות בפורמט של סעיף א', כלומר שבאינדקס 0 מצוין כמה ספרות במערך, רוצים לאפשר יצור מערך חדש בו יהיו רק חלק מהספרות שהיו במערך המקורי וזאת לפי מה שהפונקציה *isGood* תחזיר. כלומר במערך החדש יהיו רק ספרות בהן הפונקציה *isGood* החזירה 1.

הגדיר/י וממש/י את הפונקציה *createSpecialArr* אשר מקבלת את מערך הספרות בפורמט של סעיף א', ועוד משתנים לפי החלטתך. ומחזירה מערך חדש אותו היא מייצרת בו נמצאות רק הספרות מהמערך המקורי שהפונקציה *isGood* החזירה עבורן 1.

השלם את הפונקציה *Q3_B()* כך שעבור המערך הנתון:

1. יוחזר מערך בו יש רק ספרות זוגיות.
2. יוחזר מערך בו יש רק את הספרות המתחלקות ב 3.

דוגמא:

למקרה 1 עבור המערך (6 מציין את אורך המערך): [6,5,6,3,2,7,4] יודפס [6,2,4]

למקרה 2 עבור המערך (6 מציין את אורך המערך): [6,5,6,3,2,7,4] יודפס [6,3]