

מבחן סדנה מתקדמת בתכנות-61108

ד"ר מרק קורנבליט, מר חיים שפיר, גב' אסתר אמיתי

סמסטר קיץ, מועד ב', תשע"ט

תאריך: 21.11.19

הוראות:

- משך המבחן 3 שעות.

- אין להשתמש בחומרי עזר, פרט לדף המצורף לשאלון

שאלה 1 (28 נק')

כתוב פונקציה אשר מקבלת מחרוזת **infile** וגם מחרוזות **outfile_true** ו- **outfile_false** המהוות שמות של שלושה קבצי טקסט.

תוכן של קובץ ששמו נכלל ב- **infile** הוא טקסט מוצפן שנוצר מהטקסט המקורי באופן הבא:
אחרי תו אחרון של כל שורה בטקסט המקורי מוסיפים רווח (ניתן להניח שאין רווחים בטקסט המקורי);
אחרי הרווח מוסיפים מספר ביקורת ששווה לסכום קודי ASCII של כל תווי השורה;
תו של סוף השורה אמור להופיע גם בסוף של השורה האחרונה של הקובץ.

על הפונקציה לבדוק נכונות של מספר ביקורת בכל שורה.
יש להעתיק כל שורה שבה מספר הביקורת נכון לקובץ **outfile_true** וכל שורה שבה מספר הביקורת לא נכון לקובץ **outfile_false**. לא להעתיק את מספרי הביקורת עצמם.
בנוסף הפונקציה תעביר במצביעים (by reference) את מספר השורות בקובץ **outfile_true** ואת מספר השורות בקובץ **outfile_false**.
לא להשתמש במחרוזות ומערכי עזר.

דוגמא:

	<u>outfile_false</u>	<u>outfile_true</u>	<u>infile</u>
הסבר: <u>שורה</u> <u>ראשונה.</u>	3+2+1=6	abc A	abc 294 3+2+1=6 352 A 65

קודי ASCII של 'a', 'b', 'c' הם 97, 98, 99 בהתאמה. $294 = 97 + 98 + 99$.
שורה שנייה. סכום קודי ASCII של כל תווי השורה (ספרות ותווים לפעולות אריתמטיות) הוא 351. לכן מספר ביקורת 352 יהיה לא נכון.
שורה שלישית. קוד ה-ASCII של 'A' שהוא התו היחיד בשורה שווה ל- 65.

הפונקציה תעביר 2 (מספר שורות ב-**outfile_true**) ו- 1 (מספר שורות ב-**outfile_false**).



מכון טכנולוגי חולון Holon Institute of Technology

שאלה 2 (28 נק')

נתונות הגדרות הבאות של הטיפוסים:

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

```
typedef struct ptr_item {  
    int *ptr;  
    struct ptr_item *next;  
} PtrItem;
```

כתוב פונקציה המקבלת מערך **AL** של רשימות מקושרות של מספרים שלמים וגודלו. הרשימות מורכבות מאיברים מסוג **DataItem**.

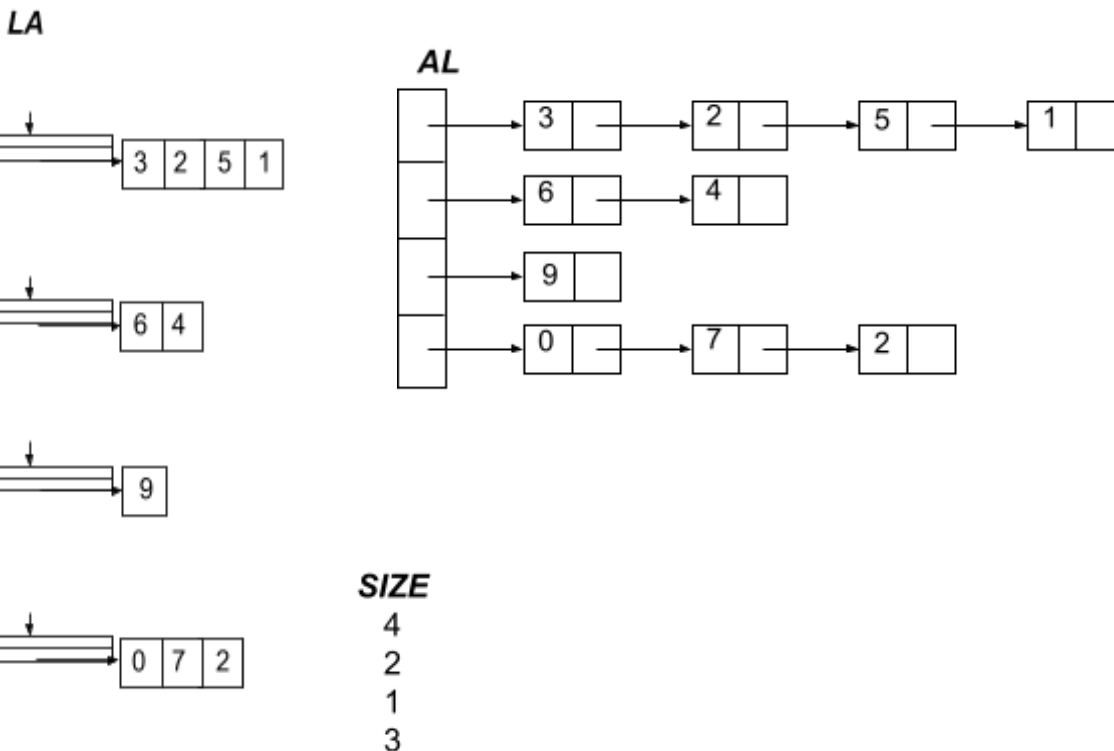
על הפונקציה לבנות רשימה מקושרת **LA** של מערכים של מספרים שלמים. הרשימה תהיה מורכבת מאיברים מסוג **PtrItem**. שדה **ptr** של איבר מס' **i** ב- **LA** יצביע לתחילת מערך מספרים מס' **i** ושדה **next** שלו יצביע לאיבר הבא ב- **LA**.

גודל מערך מס' **i** ב- **LA** יהיה שווה לגודל רשימה מס' **i** ב- **AL**. איבר מס' **j** של מערך מס' **i** ב- **LA** יהיה שווה לשדה **data** של איבר מס' **j** ברשימה מס' **i** ב- **AL**. אם רשימה מס' **i** ב- **AL** ריקה אז שדה **ptr** של איבר מס' **i** ב- **LA** יהיה שווה ל-NULL.

הפונקציה תחזיר את כתובת ראש הרשימה **LA**. בנוסף הפונקציה תיצור ותעביר במצביעים (by reference) מערך דינאמי **SIZE** המורכב מגדלי המערכים ב- **LA** כך שאיבר מס' **i** של **SIZE** יהיה שווה לגודל מערך מס' **i** ב- **LA**.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

דוגמא:



שאלה 3 (28 נק')

כתוב פונקציה (void) אשר מקבלת מחרוזת **str** שמורכבת מכמה **מילים** (**מילה** היא רצף תווים שאינם רווח המופרדים ממילים אחרות ע"י רווחים). נתון כי **str** אינה מתחילה ואינה מסתיימת ברווח. מספר רווחים בין מילים רצופות ב-**str** הוא שרירותי. יתכן ש-**str** מורכבת ממילה יחידה בלבד.

על הפונקציה לבצע היסט מחזורי של מילים ב-**str** כך שמילה ראשונה (מתחילת המחרוזת) תהיה שנייה, מילה שנייה תהיה שלישית, ..., מילה שלפני אחרונה תהיה אחרונה, מילה אחרונה תהיה ראשונה.

דוגמא:

one two three four	str לפני עיבוד:
four one two three	str אחרי עיבוד:

מספר רווחים בין מילים רצופות ב-**str** אחרי עיבוד יכול להיות שרירותי אבל על מספר הרווחים הכולל ב-**str** לא לעלות על מספר הרווחים הכולל לפני עיבוד.

אפשר להשתמש במחרוזת עזר. במקרה זה ניתן להניח שיש בזיכרון מספיק מקום להקצאתה.

שאלה 4 (16 נק')

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

- מה ההבדל בין מבנה לאיגוד?
 - (1) לשדה של מבנה ניתן לפנות דרך מצביע למשתנה ולשדה של איגוד ניתן לפנות רק דרך שם המשתנה
 - (2) לשדה של איגוד ניתן לפנות דרך מצביע למשתנה ולשדה של מבנה ניתן לפנות רק דרך שם המשתנה
 - (3) לעומת מבנה, שדות איגוד משתפים אותו מקום בזיכרון
 - (4) לעומת איגוד, שדות מבנה משתפים אותו מקום בזיכרון
 - (5) אף תשובה אינה נכונה

2. מהי הטענה הנכונה?
 - (1) גודל מצביע ל-double עולה על גודל מצביע ל-char
 - (2) גודל מצביע ל-double שווה לגודל מצביע ל-char
 - (3) גודל מצביע ל-double קטן מגודל מצביע ל-char
 - (4) לא ניתן להשוות גדלי מצביעים ל-double ול-char
 - (5) אף טענה אינה נכונה

3. מהו פלט של `printf("%d", ~0)` ?

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

4.

`int *const *q;`

- מה הוא `q`?
- (1) פוינטר לפוינטר קבוע שמצביע למשתנה
 - (2) פוינטר לפוינטר לקבוע
 - (3) פוינטר קבוע לפוינטר שמצביע למשתנה
 - (4) אף תשובה אינה נכונה

שאלה 5 (בונים – 10 נק')

מהו הפלט של התוכנית הבאה?

```
#include <stdio.h>

void main()
{
    int a[100];
    printf("%d", sizeof a + sizeof sizeof a + sizeof sizeof sizeof a);
}
```

יש לנמק את התשובה.

בהצלחה!