

מבוא למדעי המחשב (61101)

פרופ' אודי רוטיץ, מר טוביה רוזנברג

סמסטר ב', תשפ"א.

מועד ב, 11.7.2021

מדבקת הבחינה

ציון הבחינה – למילוי ע"י המרצה:

שאלה	ניקוד למילוי ע"י המרצה	ניקוד מקסימלי
1		
2		
3		
4		
סה"כ		

משך הבחינה: 3 שעות.

- אין חומר עזר.
- השימוש במחשבוניס אסור.

- במבחן 4 שאלות. יש לענות על כולם.
- כתבו תשובותיכם במחברות הבחינה ולא על גבי טופס הבחינה.
- בתחילת כל שאלה יש לרשום את מספר השאלה.
- בכל השאלות ניתן להניח שהפרמטרים שמועברים לפונקציות תקינים. אין צורך לבדוק את תקינותם.

שאלה 1 (25 נקודות)

נגדיר שפה שבה בכל מילה כל אות שונה משאר האותיות. לדוגמא המחרוזת הבאה היא משפט חוקי בשפה :

"pmtwxn9mtn pbr-%\$jpcuwfRjvcufgxn\$pbL"

המחרוזת הבאה איננה חוקית בשפה כי יש בה מילה אחת לפחות שאיננה עומדת בחוקי השפה:

"pmtwtn4mtn pbr-jfcuwfjrRjvcu3fggxn\$pbL"

בשפה יכולים להיות כל סוגי התווים. המילים בשפה מורכבות מאותיות קטנות (LowerCase) בלבד. התו המפריד בין המילים יכול להיות כל תו שאיננו אות קטנה (תו שהוא אות גדולה (UpperCase) או תו שהוא ספרה ייחשב למפריד בין המילים).

כתוב פונקציה בשם CheckLang המקבלת מחרוזת (משפט) מהשפה ובודקת האם המחרוזת עומדת בתנאי השפה. אם המחרוזת עומדת בתנאי השפה הפונקציה תחזיר 1 ואחרת היא תחזיר אפס.

- חתימת פונקציית CheckLang : `int CheckLang(char * sentence)` אין להוסיף פרמטרים לחתימת הפונקציה

- לשם פתרון השאלה מומלץ להשתמש בפונקציית עזר בשם CheckWord הבודקת האם כל האותיות במילה שונות האחת מהשנייה. חתימת הפונקציה: `int CheckWord(char * word)`

- לפונקציות אסור לשנות את מחרוזת הקלט

שאלה 2 (25 נקודות)

מספר במערך הוא מקסימום יחסי אם ערכו גבוה משני שכניו. ערך המקסימום הוא סכום ההפרשים בין המספר ובין שני שכניו. כתבו פונקציה **רקורסיבית** בשם MaxRelative המקבלת את המערך וגודלו, ומחזירה את ערך המקסימום היחסי הגדול ביותר המופיע במערך.

דוגמא (מערך בגודל 15):

24 28 22 4 6 7 8 6 5 11 10 21 9 4 13

הפונקציה תחזיר 23 משום ש- 21 (המופיע בשלשה המסומנת בקו תחתון) הוא המקסימום היחסי הגדול ביותר במערך. ערכו של המקסימום היחסי הגדול ביותר הוא 23 לפי החישוב הבא: $23 = (21-9) + (21-10)$

שימו לב :

- ניתן להניח שיש לפחות מקסימום יחסי אחד במערך, וכן שבמערך יש לפחות 3 איברים.
- לא קיים ערך יחסי לאיבר הראשון והאחרון במערך מכיוון שאין להם שני שכנים.
- על חתימת הפונקציה להיות `int MaxRelative(int *arr, int size)`. אין להוסיף פרמטרים נוספים לחתימת הפונקציה.

שאלה 3 (25 נקודות)

כתבו פונקציה שמקבלת כפרמטרים מערך שמכיל מספרים 0,1,2 בלבד בסדר כלשהו ואת גודלו n . על הפונקציה לשנות את סדר האיברים במערך כך שלאחר הקריאה לפונקציה המערך יהיה ממוין.

ניתן להניח שכל אחד מהמספרים 0,1,2 מופיע במערך לפחות פעם אחת.

על סיבוכיות זמן הריצה של הפונקציה להיות מסדר גודל של n .

אין להשתמש במערך עזר לפתרון השאלה.

על חתימת הפונקציה להיות:

```
void sort_arr(int a[],int n)
```

אין להוסיף פרמטרים נוספים לחתימת הפונקציה.

לדוגמה נניח שתוכן מערך a הוא: 0,1,0,0,1,2,1,2,0,2,1,0,0

לאחר הקריאה לפונקציה על ידי הפקודה:

```
sort_arr(a,13)
```

תוכן המערך a יהיה: 0,0,0,0,0,0,1,1,1,1,2,2,2

שאלה 4 (25 נקודות)

כתבו פונקציה שמקבלת כפרמטר מערך דו מימדי (מטריצה) בגודל $N \times N$ כאשר N מוגדר בעזרת `#define` ומשנה את סדר האיברים במטריצה באופן הבא:

בשורה הראשונה יופיעו כל האיברים הקטנים ביותר במטריצה, בסדר עולה משמאל לימין.

בשורה השנייה יופיעו כל האיברים הקטנים ביותר במטריצה, שגדולים מכל האיברים בשורה הראשונה, בסדר עולה משמאל לימין

בשורה השלישית יופיעו האיברים הקטנים ביותר במטריצה, שגדולים מכל האיברים בשורה השנייה (ולכן גם גדולים מכל אלה שבשורה הראשונה), בסדר עולה משמאל לימין.

וכן הלאה עד השורה האחרונה שבה יופיעו כל האיברים הגדולים ביותר במטריצה ממוינים בסדר עולה משמאל לימין. יש להניח שהאיברים במטריצה הם מסוג `int`.

על סיבוכיות זמן הריצה של הפונקציה להיות מסדר גודל של $n^2 \log n$

מותר להשתמש במערך עזר אחד בלבד (אין הגבלה על מספר האיברים במערך העזר).

על חתימת הפונקציה להיות:

`void sort_Mat(int a[][N])`

אין להוסיף פרמטרים נוספים לחתימת הפונקציה.

לדוגמה, נניח ש $N=4$ ותוכן המטריצה a לפני הקריאה לפונקציה הוא:

2	14	16	8
2	-7	17	7
-1	-4	-7	-9
0	15	3	20

לאחר הקריאה לפונקציה על ידי הפקודה:

`sort_Mat(a)`

תוכן המטריצה a יהיה:

-9	-7	-7	-4
-1	0	2	2
3	7	8	14
15	16	17	20

במבחן אפשר להעזר בפונקציות הבאות כקופסה שחורה:

```
void merge_sort(int *,int,int);
```

```
void quick_sort(int*,int,int);
```

```
int binary_search(int*,int,int);
```

```
int binary_search_last(int*,int,int);
```

```
void swap(int*,int*);
```

```
int partition(int*,int,int); //int split(int*,int,int);
```

```
int* merge_arrays(int*,int,int*,int);
```

```
int strlen(char *);
```

```
void strcpy(char *, char *);
```

```
int strcmp(char *, char *);
```



בהצלחה !