

מבוא למדעי מחשב מ' / ח' (234114 / 234117) סמסטר חורף תשס"ו

מבחן מסכם מועד ב', 22 מרץ 2006

_	שם פרטי	 שם משפחה	_		 		טודו	יר ס	วดก

משך המבחן: 3 שעות.

חומר עזר: אין להשתמש בכל חומר עזר בכתב, מודפס או אלקטרוני.

הנחיות והוראות:

- . מלאו את הפרטים בראש דף זה.
- בדקו שיש 20 עמודים (5 שאלות) במבחן, כולל עמוד זה.
- כתבו את התשובות על טופס המבחן בלבד, במקומות המיועדים לכך. שימו לב שהמקום המיועד לתשובה אינו מעיד בהכרח על אורך התשובה הנכונה.
- העמודים הזוגיים בבחינה ריקים. ניתן להשתמש בהם כדפי טיוטה וכן לכתיבת תשובותיכם. סמנו טיוטות באופן ברור על מנת שהן לא תיבדקנה.
 - יש לכתוב באופן ברור, נקי ומסודר. ניתן בהחלט להשתמש בעיפרון ומחק.
 - אין לכתוב הערות והסברים לתשובות אם לא נתבקשתם מפורשות לכך.
 - בכל השאלות, הינכם רשאים להגדיר (ולממש) פונקציות עזר כרצונכם.
 - אין להשתמש בפונקציות ספריה או בפונקציות שמומשו בכיתה אלא אם צוין אחרת בשאלה.

בודק	הישג	ערך	שאלה
		20	1
		20	2
		20	3
		20	4
		20	5
		100	סה"כ

	צוות הקורס 234114
פ' רון קימל (מרצה אחראי).	מרצים: סאהר אסמיר, פרו
, גיא סלע, ולנטין קרבצוב	מתרגלים : עידן בן-הרוש, ג

מתרגלים: עידן בן-הרוש, גיא סלע, ולנטין קרבצוב, מרק גינזבורג, רן רובינשטיין (מתרגל אחראי).

צוות הקורס 234117

מרצים: ארז חדד, ויטלי סקצ'ק, פרופ' רון קימל (מרצה אחראי).

מתרגלים: שיאון שחורי, רג'א ג'יריס, אסנת טל, מרק גינזבורג, ארקדי פיורו, רן רובינשטיין (מתרגל אחראי).

בהצלחה!





<u>שאלה 1 (20 נקודות)</u>

נתונים שלושה מערכים של מספרים שלמים c-l b a ו-c, ואורכיהם m n בהתאמה. כל מערך ממוין בסדר עולה-ממש, וללא חזרות. עליכם לחשב מערך פלט d שיכיל את החיתוך של שלושת המערכים; כלומר, ערך כלשהו יופיע ב-b אם ורק אם הוא מופיע בכל שלושת מערכי הקלט. על מערך הפלט d להיות ממוין אף הוא בסדר עולה וללא חזרות. כמו כן על הפונקציה להחזיר את מספר האיברים במערך d כערך ההחזרה שלה. לדוגמה, עבור מערכי הקלט:

a:	1 2 6 9	
b:	2 5 6 7 8 9 11	
c:	2 6 8 9	יוסדר 2 יוסד כל מווכר בסלט בבעי
q.	2 6 9	יוחזר 3, ויתקבל מערך הפלט הבא:

.O(1) אורכי שלושת מערכי הקלט), מקום נוסף (O(N) הוא סכום אורכי שלושת מערכי הקלט), מקום נוסף

<pre>int c[], int r, int d[]) {</pre>	int intersect3(int	a[],	int n,	int b[],	int m,
	int	c[],	int r,	<pre>int d[])</pre>	€



-	



שאלה 2 (20 נקודות)

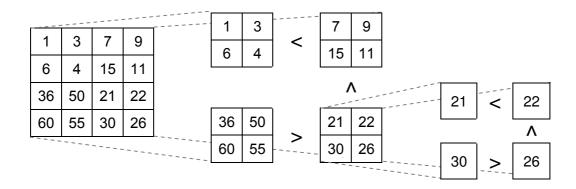
בשאלה זו נתייחס למערכים דו-ממדיים **ריבועיים** בגודל NxN (כאשר N הוא קבוע המוגדר כ-define#). לצורך השאלה, נניח כי N הוא חזקה שלמה של 2. עבור מערך כזה, נגדיר חלוקה פנימית שלו לארבעה רובעים בגודל N/2xN/2, ממוספרים מ-1 עד 4, באופן הבא:

1	2
4	3

נאמר שהמערך הוא **סיבובי** אם כל האיברים ברובע 1 קטנים ממש מכל אלו שברובע 2, אלו שברובע 2 קטנים ממש מכל אלו שברובע 3, ואלו שברובע 3 קטנים ממש מכל אלו שברובע 3, ואלו שברובע 3, ואלו שברובע 3 קטנים ממש מכל אלו שברובע 3, ואלו שברובע 5, ואלו שברובע 5, ואלו שברובע 3, ואלו שברובע 5, ואלו שברובע 5,

1	5
9	7

לשם הנוחות, נגדיר גם כל מערך בגודל 1x1 כמערך סיבובי. כעת, נאמר שמערך NxN הוא **ממוין-סיבובית** אם הוא סיבובי, ארבעת הרובעים שלו סיבוביים, וכן הלאה עד לרובעים בגודל 1x1. לדוגמה, המערך הבא ממוין-סיבובית:



עליכם לממש פונקציה לחיפוש במערך ממוין-סיבובית (בעמוד הבא). הפונקציה מקבלת כפרמטר את המערך הדו-ממדי [N][N], ואת הערך לחיפוש x. במידה ו-x נמצא במערך, הפונקציה תחזיר 1 ותכתוב את הקואורדינאטות שלו למשתנים i,j הניתנים כפרמטרים. במידה ו-x אינו במערך, הפונקציה תחזיר 0 ואין חשיבות לתוכן של i,j.

דרישות סיבוכיות: על הפונקציה לעבוד בסיבוכיות זמן טובה ככל הניתן. פתרון לא אופטימאלי יזכה בניקוד חלקי לכל היותר. כמו כן על הפונקציה לעבוד בסיבוכיות מקום נוסף (O(1).





int	find2d(int	a[N][N],	int x,	int* i	, int*	i)	f
	•	/	•		•	٠.	·



-	



שאלה 3 (20 נקודות)

(סעיף א (10 נקודות)

נתון **מערך של מחרוזות** [strings] באורך N (כש-N מוגדר כ-define), **ומפתח** אשף הוא מחרוזת. key עליכם לממש פונקציה המחשבת לכל מחרוזת במערך strings כמה תווים משותפים יש לה עם המחרוזת cals[i] כאשר חזרות נספרות, ראו דוגמה בהמשך). על הפונקציה לרשום את התוצאות למערך vals, כש-[i] מכיל את מספר התווים המשותפים למפתח עם המחרוזת ה-i. **שימו לב:** אין לשנות את מחרוזות הקלט.

לדוגמה, למחרוזת "Good morning Israel!" ולמפתח "oops" יש 3 תווים משותפים: שני 'ס' ו-'s' אחד. לעומת זאת עם המפתח "!ooooops" יש לה 5 תווים משותפים: שלושה 'ס', 's' אחד ו-'!' אחד.

דרישות סיבוכיות: סיבוכיות מקום נוסף (O(1), וסיבוכיות זמן O(N*m+k) כש-m הוא אורך המחרוזת הארוכה סיבוכיות: סיבוכיות מקום נוסף (O(1), וסיבוכיות זמן k-l כשאלה זו ניתן להתייחס למספר התווים האפשריים (256) כקבוע.

void keycount(char*	strings[N],	char* key,	<pre>int vals[N])</pre>	{



D	



סעיף ב (10 נקודות)

בשאלה זו נתון מערך של מחרוזות ומפתח, כמתואר בסעיף א'. יש לממש פונקציה **שממיינת** את מערך המחרוזות בסדר עולה על פי ערכן של המחרוזות, כאשר ערכה של מחרוזת מוגדר כמספר התווים שמשותפים לה עם המפתח. בסעיף זה ניתן להשתמש בפונקציה מסעיף א' גם אם לא פתרתם אותו.

כפי שהוגדרו (ש-N, m , N-טיבוכיות מקום נוסף (O(N). סיבוכיות ממן (יש-N, m , N-טיבוכיות מקום נוסף (בסעיף א'.

<pre>void keysort(char* strings[N], char* key) {</pre>



Table Tabl	
Table Tabl	



<u>שאלה 4 (20 נקודות)</u>

כתבו את סיבוכיות הזמן והמקום הנוסף של הפונקציות f1 ו-f2:

```
void f1(int n)
{
  int i, k=0;

  for (i=0; i<n; i++)
     k += g1(i+1)-g1(i);

  while (k>0) {
     printf("k= %d", k);
     k--;
  }
}

int g1(int n)
{
  int i, j=0;
  for (i=n; i>0; i--)
     j += i;
  return (int)sqrt(2*j-n);
}
```

```
void f2(int n)
{
   if (n<=1)
      return;
   g2(n, n/3);
}

void g2(int n, int m)
{
   int i=1;
   while (m < n) {
      m += i;
      i++;
   }
   f2(n/2);
}</pre>
```

```
\Theta( ______ ) סיבוכיות זמן: (______ ) \Theta( סיבוכיות מקום נוסף: \Theta(
```



Table Tabl	
Table Tabl	



שאלה 5 (20 נקודות)

מעוניינים להרכיב צוות שמירה למוזיאון. ברחבי המוזיאון פזורות עמדות שמירה, שבכל אחת ניתן לשים שומר אחד בלבד. הדרישה היא שלכל פסל במוזיאון יהיו לפחות שני שומרים המסוגלים לראות אותו. ברצוננו למצוא אחד בלבד. הדרישה היא שלכל פסל במוזיאון יהיו לפחות על המוזיאון. לשם כך, נתונה מטריצה a בגודל את צוות השומרים המינימאלי הדרוש על מנת לשמור על המוזיאון. לשם כך, נתונה מטריצה a מכיל a את מספר הפסלים ו-a את מספר עמדות השמירה. a [i][i] מכיל i אם נקודת השמירה i .

השלימו את הפונקציה findGuardTeam למטה, ואת פונקצית העזר בה היא משתמשת (בעמוד הבא), אשר מקבלת כפרמטר את המטריצה a ומחזירה את גודל צוות השמירה המינימלי המקיים את הדרישות. אם לא קיים כזה, הפונקציה מחזירה את הערך המיוחד 1+M. כמו כן, במידה וניתן למצוא צוות שמירה העומד בדרישות, על הפונקציה לכתוב את מיקומי השומרים למערך result, שיכיל 1 במקום ה- i אם יש למקם שומר בעמדה ה- i, ו- 0 אחרת.

לנוחותכם, ניתן להשתמש בפונקצית העזר הבאה, המקבלת מערך באורך n של מספרים שלמים וערך x, ומחזירה 1 אם כל איברי המערך גדולים או שווים ל-x, ו-0 אחרת.

```
int atleastX(int arr[], int n, int x)
```

הערה: מספר השורות הריקות לא בהכרח מעיד על מספר שורות הקוד שיש לכתוב, ואורך הקו אינו בהכרח מעיד על אורך השורה שיש להשלים. ניתן להניח כי N ו-M הם קבועים המוגדרים כ-define.

```
int findGuardTeam (int a[M][N], int result[M])
{
   int aux[_____]= {0}, i;

for ( i = 0 ; i < M ; ++i )
    result[i] = _____;

return guardTeamAux(A, result, aux, 0);
}</pre>
```



_



```
int guardTeamAux (int A[M][N], int result[M], int aux[], int k)
   int min_with_curr, min_without_curr, i;
   if ( ______ ) return 0;
   if ( ______ ) return M+1;
   for (i = 0; i < N; ++i) {
       aux[i] += _____;
   }
   min_with_curr =
       guardTeamAux ( A, _____, ____, ____);
   for (i = 0; i < N; ++i) {
       aux[i] -= _____;
   min_without_curr =
       guardTeamAux ( A, _____, ____, ____);
   if ( min_with_curr +1 < min_without_curr) {</pre>
       return _____;
   return _____;
}
```



הפקולטה למדעי המחשב סמסטר חורף תשס"ו (2006)





הפקולטה למדעי המחשב סמסטר חורף תשס"ו (2006)





הפקולטה למדעי המחשב סמסטר חורף תשס"ו (2006)

