



3/10/11

N/A

 שאלה מס' 1 (25 נקודות)

שים לב שבשאלה זו שני סעיפים.

א (15 נקודות).

כתוב פונקציה בשם `legal` אשר מקבלת כפרמטר מחרוזת המכילה
ביטויים מתמטיים ובודקת חוויקות הופעת סוגרים עגולים בו.
הfonקציה מחזירה 1 אם הסוגרים מופיעים באופן חוקי, אחרת מחזירה 0.

הגדרה:

על שני התנאים הבאים להתקיים על מנת שביטוי יהיה חוקי:

1. כל סוגר שמאלי חייב להסגר עליו סוגר ימני.
2. סוגר ימני יכול להופיע רק אם הוא סוגר את הופעת סוגר שמאלי.

לדוגמא:

- ביטויים לא חוקיים:
- $a + b, (a - b, ((c-b$
- $a + b, ((a) - b), ((c - b)$

int is_legal(char str) {

int cnt = 0, i = 0,

char

while (str[i] != '\0') {

if (str[i] == '(') cnt++;

if (str[i] == ')') cnt--;

if (cnt < 0) {

return 0;

break;

}

i++;

}

if (cnt == 0) return 1;

}

המשך השאלה בעמוד הבא



11

שאלה מס' 1 (20 נקודות).

כתבו פונקציה

```
void print_longest_sentence(char *str)
```

המקבלת מחירות (לא מוגבלת באורך) ומדפיסה את המשפט ה"תיקין" הארוך ביותר. אם יש כמה כליה, על הponeקציה להציג את המשפט ה"תיקין" הארוך ביותר האחרון.

הנדרה: משפט הוא סדרה רצופה של תווים ש谟פרדת משאר הטקסט ע"י אחד מסימני הפיסוק הבאים: נקודה (.), סימן קראיה (!) או סימן שאלה (?). (שים לב שהמשפט הראשון מתחילה בתחילת המשפט הקודם)

הגדירה: משפט "תקין" הוא משפט שהתו הראשו בו שאיננו רוח הוא אות גזולה
הגדירה: אורץ משפט הוא מספר התווים שאינם רוח במשפט (כולל כל סימני

הערה: מותר לסימני פיסוק אחרים להופיע באמצע המשפט (למשל פסיק (,), גשודותיים ())).

דוגמה: במחרזות הבאה:

"This is a valid sentence. this is not a valid sentence! Is this a valid sentence?"

יש שני משפטים תקינים, שניהם באורך 21 והפונקציה תדפיס:

Is this a valid sentence?

שאלה מס' 3 (20 נקודות)

כתבו במקומות המסומנים מה ידפיס הקוד הבא

```
#include <iostream.h>

template<class T>
int
f1(T x, const T *v)
{
    for (int i = 0; v[i]; ++i) {
        if (v[i] == x) return i;
    }
    return -1;
}

template<class T>
T *
f2(T target[], const T src[], const T dst[])
{
    for (int i = 0; target[i]; i++) {
        int ind = f1(target[i], src);
        if (ind >= 0) target[i] = dst[ind];
    }
    return target;
}
```

```
template<class T>
int
f3(const T s[])
{
    T p = 0;
    int mx = 0, cr = 0;

    for (int i = 0; s[i]; ++i) {
        mx = (cr > mx ? cr : mx);
        if (s[i] == p) {
            cr++;
        } else {
            cr = 1;
        }
        p = s[i];
    }
    return (cr > mx ? cr : mx);
}
```

```
int
main()
{
    int s = 0;
    for (char i = 'a'; i < 'e'; ++i) {
        static c = 1;
        s += i * (c *= -1);
    }
}
```

① cout << "s = " << s << endl;

① _____

② cout << "[1]" << f1('b', str1) << " [2]" << f1('e', str1) << endl;

② _____

③ cout << f2(str1, d1, d2) << endl;

③ _____

④ char str2[20] = "11aaabb2";
④ cout << f3(str2) << endl;
④ return 0;

④ _____

שאלה מס' 4 (25 נקודות)

בתוכנית מוגדרים `N` `const int` `M` `const int` `table [N][M]` וכן `[N][M] typedef int`.
 "טבלה" בגודל $N \times M$ נקראת "מלוכסנת" אם יש בה שני אלכסונים מצטלבים המכילים מספר אחד; לדוגמה, הטבלה הבאה מלוכסנת (עמ' ערך 6).

	1	2	3	4	2	
1	3	4	1,8	6	1	6
2	5	2,7	6	4	6	3
3	3,0	3,1	3	6	5	6

הערה: אלכסון צריך להיות באורך 2 לפחות.

כתבו פונקציה `()`

המקבלת "טבלה" כפרמטר ומחזירה `true` אם הטבלה מלוכנת, `false` אחרת.

28

שאלה מס' 5 נקודות ב- 30 זיהו).

עליכם לכתוב פונקציה `SumArray` ותואימה להצורה הבאה :

`int SumArray(int arr[])`

הfonקציה מקבלת כפרמטר מערך של `int`, באורך בלתי דינמי, ומחשבת את סכום האיברים במערך. בمعודן יכול להיות מספרים חיוביים ושליליים, וסכום מצוין עי' הערך 0 (אפס).

דוגמא :

5	-6	12	1	-9	0	...
---	----	----	---	----	---	-----

צון סוף
מערך

עליכם לפתרו את השאלה ללא שימוש בשוט לולאה!!!

`int SumArray(int arr[])`

3

`if arr[0] = 0 return 0;`

`return 0;`

`return arr[0] + sumArray(arr+1);`

3

30

 שאלה מס' 4 (15 נקודות כ- 22 דקות)

מחזרות תווים תיקרא מחרשת עליה אם התווים במחזרות מסודרים בסדר
עליה (עפ"י היצוג המספרי שלהם - ASCII).
המחזרות הריקת וכל המחרוזות באורך 1 נחשבות למחזרות עולות.

דוגמאות: "1336", "abbc", "a", "", "2463", "XX2" אין עלות.
תuren מחרשות עולות.

בתובנית מוגדר

```
#define TRUE 1
#define FALSE 0
```

א. (7 נקודות) כתבו פונקציה רקורסיבית המחזירה TRUE אםascend(char* s) הבודק אם s מחרצת עליה וFALSE אחרת.

```
int ascend(char* s)
```

```
{
    int static result = True;
    if (*s == '\0') return result;
    if (*s >= s[1] && *s <= s[1]) return result = false;
    if (result == True) ascend(s+1);
    return result;
}
```

ב. (8 נקודות) כתבו את הפונקציה הניל בצורה שאייה רקורסיבית.

```
int ascend(char* s)
```

```
{
    int index1, index2, result;
    for(index1=0, index2=1; *(s+index2) != '\0'; index1++, index2++)
        if (*(s+index1) > *(s+index2))
            result = True;
        else result = False;
    if (*(s+1) == '\0') result = True;
    return result;
}
```

```
15
```