

1. ערוך טבלת מעקב על שתי תוכניות אלה. במעקב התעלם משורות שגויות.

נתונה השורה הבאה של תכנית בשפת C:

```
int a, *b, c[4];
```

סמנו את כל ההוראות שאינן יכולות להופיע באופן חוקי בהמשך התכנית.
הוראה חוקית היא הוראה נכונה מבחינה תחבירית שעוברת קומפילציה ללא שגיאות ואזהרות.

- א) `*(c+3) = 8;`
- ב) `a = *(c + *b);`
- ג) `*(c++) = 12;`
- ד) `c = b;`
- ה) `b = c;`
- ו) `a = (*c)++;`
- ז) `*(b+1) = (*c)++;`
- ח) `a = *b - *c;`
- ט) `*c = *(b++);`
- י) `*(b++) = *(&a);`
- יא) `*b==2=a;`
- יב) `c[3] = *b == 2;`

נתונות השורות הראשונות של תכנית בשפת C:

```
void main() {  
    int arr[20], *ptr, num;  
    ptr = arr;
```

סמן את כל ההוראות שאינן יכולות להופיע באופן חוקי בהמשך התכנית.
הוראה חוקית היא הוראה נכונה מבחינה תחבירית שעוברת קומפילציה ללא שגיאות ואזהרות.

- א) `arr++;`
- ב) `arr = #`
- ג) `ptr++;`
- ד) `ptr = #`
- ה) `*ptr = 8;`
- ו) `*num = 8;`
- ז) `(arr+2) = 8;`
- ח) `&num = 8;`
- ט) `&num = &arr[5];`
- י) `num = *ptr;`

שאלה 2

בשפות התכנות Java ו-C# ניתן לגלות את אורכו של אובייקט מסוג מערך על-ידי שימוש בתכונה Length. שפת C אינה מונחית עצמים, ומערכים אינם אובייקטים (ולכן – גם אין להם תכונה Length). בשם Length.

א. כיצד ניתן לגלות את גודלו של מערך חד-מימדי (את כמות האיברים בו)? הריצו את התכנית הבאה:

```
#include <stdio.h>

int main()
{
    int arr[5] = {1,2,3,4,5};    /* מגדירים מערך של חמישה מספרים שלמים */
    printf("גודלו של משתנה מטיפוס שלם הוא %d בתים", sizeof(int));
    printf("גודלו של איבר במערך הוא %d בתים", sizeof(arr[0]));
    printf("גודלו של המערך כולו הוא %d בתים", sizeof(arr));
    printf("לכן, במערך יש %d איברים", sizeof(arr)/sizeof(arr[0]));
    return 0;
}
```

ב. מעוניינים לנהוג בצורה דומה גם כדי לערוך חישובים לגבי מערך דו-מימדי. השלימו את הביטויים החסרים בתכנית הבאה:

```
#include <stdio.h>

int main()
{
    int matrix[5][3];
    printf("גודלו של איבר במטריצה הוא %d בתים", (1));
    printf("גודלה של שורה במטריצה הוא %d בתים", (2));
    printf("גודלה של המטריצה כולה הוא %d בתים", (3));
    printf("מספר העמודות במטריצה הוא %d", (4));
    printf("מספר השורות במטריצה הוא %d", (5));
    printf("מספר האיברים במטריצה הוא %d", (6));
    return 0;
}
```

3. כתוב פונקציה המקבלת מספר מסוג long ומזירה את מספר ספרותיו ואת סכום ספרותיו.
4. כתוב פונקציה המקבלת שלושה מערכים בגודל זהה. הפונקציה תחבר את ערכי התאים של המערכים בהתאמה ותכניסם למערך שלישי. יש להשתמש במחזורים (pointers) ולא באינדקסים בכדי לעבור על התאים.

לדוגמא:

מערך א': 2,4,5,8,11,3

מערך ב': 9,6,7,2,1,1

מערך ג': 11,10,12,10,12,4

שאלה 5

בשאלה זאת נטפל בחשבונות בנק אשר נתונים כמערך ובו N חשבונות (N מוגדר ב-define). מספר החשבון מיוצג ע"י מספר התא (האינדקס) במערך, ויתרת החשבון מיוצגת ע"י תוכן התא

כתבו פונקציה `int transfer(int *from, int *to, int amount)`, המקבלת שני מצביעים לחשבונות שונים (`to, from`), וסכום כלשהו `amount`. הפונקציה פועלת באופן הבא: אם `amount` חיובי ובחשבון ה-`from` קיים לפחות הסכום `amount`, היא מעבירה את הסכום `amount` מחשבון `from` לחשבון `to`, ומחזירה 0. אחרת מעבירה את הכסף שנשאר בחשבון ומחזירה את יתרת התשלום שלא הועבר.

דוגמאות: עבור המערך המופיע מטה:

0	1	2	3	4	5	6	7
5	65	1	34	67	72	48	23

עבור `to=&arr[5], from=&arr[1]` וגם `amount=10`, הפונקציה תחזיר 0, והמערך ישתנה באופן הבא:

0	1	2	3	4	5	6	7
5	55	1	34	67	82	48	23

דוגמא נוספת:

0	1	2	3	4	5	6	7
5	65	1	34	67	72	48	23

עבור `to=&arr[2], from=&arr[3]` וגם `amount=40`, הפונקציה תחזיר 6 כי זו יתרת התשלום שלא שולמה. המערך ישתנה באופן הבא.

0	1	2	3	4	5	6	7
5	65	35	0	67	72	48	23

שאלה 6

ממשו את הפונקציה `int search_larger(int arr[], int n, int k)` המקבלת מערך של שלמים חיוביים `arr[]`, את גודלו `n` ומספר נוסף `k`. הפונקציה מחזירה את המספר הקטן ביותר במערך אשר גדול ממש `k`. אם אין מספר כזה הפונקציה מחזירה -1.

למשל, עבור המערך

arr[0]	arr[1]	arr[2]	arr[3]	arr[4]	arr[5]
6	5	1	2	7	9

והערך `k=8`, הפונקציה תחזיר 9. עבור `k=11` הפונקציה תחזיר -1.

מומלץ להשתמש בפונקציות עזר שמוצאת המינימום והמקסימום במערך.

שאלה 7

השלם את הפונקציה הבאה. הפונקציה מחשבת את אורך המחרוזת שהתקבלה

```
#include <stdio.h>
#define MAX_LENGTH 80
/*Calculates a string's length */
int my_strlen (char *s)
{
    char *p = _____1_____;
    while (_____2_____)
        _____3_____;
    return _____4_____;
}
void main()
{
    char str[MAX_LENGTH];
    int len;
    printf("Enter a string:");
    gets(str);
    len = my_strlen(str);
    printf("The lenght of the string %s is %d \n",str,len);
}
```

שאלה 8

כתוב פונקציה המקבלת לפחות את הפרמטרים הבאים:

מערך של מספרים וגודלו, ו-2 מספרים נוספים

הפונקציה תחזיר את כמות האיברים שערכם מתחלק ללא שארית במספר הראשון וכן את כמות האיברים שערכם מתחלק ללא שארית במספר השני

דוגמא: עבור המערך {1,2,3,4,5,6,7,8,9,10,11}, גודלו 11 והמספרים 2 ו-4, הפונקציה תחזיר:

- 5 (מאחר ו-5 ערכים מתחלקים ב-2 ללא שארית)
- 2 (מאחר ו-2 ערכים מתחלקים ב-4 ללא שארית)

לפניך פונקציה בשם `checkOneArray` המקבלת שלושה פרמטרים `A, n, place` כאשר `A` הוא מערך בגודל `n`. הפונקציה מוצאת ומחזירה את אורך הרצף הארוך ביותר של מספרים המסודרים בסדר עולה במערך `A`. אם הרצף הזה מכיל שני איברים לפחות, אז הפונקציה מציבה בפרמטר `place` את האינדקס (המיקום במערך `A`) שבו הרצף הזה מתחיל; אחרת היא מציבה בפרמטר `place` את הערך `(-1)`.

הערה: אם קיימים שני רצפים לפחות, שהם הארוכים ביותר, אז הפונקציה מציבה בפרמטר `place` את האינדקס (המיקום במערך `A`) של הרצף הראשון מביניהם.

בפונקציה חסרים **חמישה** ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת הבחינה את מספרי הביטויים החסרים (1) – (5) בלבד, בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

```
int checkOneArray( const int A[ ],int n,int *place)
{
    int i,j,max=1, count=1 ;
    *place= -1;
    for (i=0; i<n; i+=count)
    {
        count=_____ (1) _____ ;
        for (j=_____ (2) _____ ; j<n-1; j++)
        {
            if ( _____ (3) _____ ) count++; else break;
        }
        if (count>max)
        {
            max=count; _____ (4) _____ ;
        }
    }
    return _____ (5) _____ ;
}
```

ב. לפיך פונקציה בשם `checkmat` המקבלת מטריצה `mat` בגודל $n \times n$. הפונקציה מחזירה את אורך הרצף הארוך ביותר של מספרים המסודרים בסדר עולה, בשורה כלשהי במטריצה `mat`. אם הרצף הזה מכיל שני איברים לפחות בשורה כלשהי במטריצה `mat`, אז הפונקציה מחזירה באמצעות הפרמטרים `maxrowindex` ו-`maxcolindex` את האינדקס (מספר השורה ומספר עמודה במטריצה `mat`) שבו מתחיל הרצף הזה; אחרת הפונקציה קובעת בעבורם את הערך (-1).

הערות:

1. כל שורה במטריצה עומדת בפני עצמה ואינה נמשכת לשורה הבאה (העוקבת).
 2. אם קיימים שני רצפים לפחות, שהם הארוכים ביותר, אז הפונקציה תחזיר את האינדקס של הרצף הראשון מביניהם.
- בפונקציה חסרים שלושה ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת הבחינה את מספרי הביטויים החסרים (1) - (3) בלבד, בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

```
int checkmat(const int mat[][n],int n,int *maxrowindex,int *maxcolindex)
{
    int i,tmp,max=1, colindex;
    *maxcolindex=-1;
    *maxrowindex=-1;
    for (i=0; i<n; i++)
    { tmp=checkOneArray(_____(1)_____,n,_____(2)_____) ;
      if ( tmp>max)
      {   max=tmp;
          *maxcolindex=colindex;
          _____(3)_____=i;
        }
    }
    return max;
}
```

ג. לפיך קטע של קוד שבו מזמנים את הפונקציה שבסעיף הקודם (סעיף ב').

בקטע הקוד חסרים שלושה ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת הבחינה את מספרי הביטויים החסרים (1) – (3) בלבד, בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

אילוץ: בביטוי החסר (1) אסור להשתמש במשתנה n .

```
int ans= checkmat(A , ____ (1) ____ , ____ (2) ____ , ____ (3) ____ ) ;  
  
if (ans>1)  
    printf("Length :%d,found at row index :%d,column index :%d\n" ,  
           ans, rowind, colind) ;  
  
else puts("Such a sequence does not exist !") ;
```