

# פתרון מבחן בשפה עילית טכנאים אביב תשפ"ג סמל שאלון 711911

שאלה 5

נתונה התוכנית שלהלן:

```
#include <stdio.h>
void main()
{
    int num,i,div,mod,arr[100],j;
    printf("Enter an integer Number\n");
    scanf("%d",&num);
    div=num;
    i=0;

    while(div)
    {
        mod = div%2;
        div = div/2;
        arr[i] = mod;
        i++;

    }
    printf("The transformed Number is: ");
    for (j=i-1;j>=0;j--)
    {
        printf("%d",arr[j]);
    }
}
```

- א. (8 נק') הקלט המוזן על ידי המשתמש הוא המספר 23. בנו במחברת הבחינה טבלת מעקב עבור לולאת while שבתוכנית.
- ב. (5 נק') מה מבצעת לולאת ה־for שבתוכנית?
- ג. (6 נק') מה תפקידה של התוכנית כולה?
- ד. (6 נק') משנים את לולאת ה־while באופן הבא:

```
while(div)
{
    mod = div%8;
    div = div/8;
    arr[i] = mod;
    i++;

}
```

לאחר השינוי מריצים שוב את התוכנית עבור אותו הקלט 23. מה יהיה פלט התוכנית? נמקד את התשובה.

## פתרון שאלה 5

בהנחה שהמשתמש הקיש את המספר 23 :

num	23						
i		<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
mod=mod%2		1	1	1	0	1	
div =div/2	23	11	5	2	1	0	
arr[0]		1					
arr[1]			1				
arr[2]				1			
arr[3]					0		
arr[4]						1	
arr[5]							
-----							
arr[99]							

ב. לולאת ה for מדפיסה את המערך arr מהאיבר החמישי (אינדקס 4) עד האיבר הראשון ( אינדקס 0 ) אחד אחרי השני. במקרה הזה יודפס במסך :

The transformed Number is : 10111

ג. התוכנית מבצעת המרה של מספר עשרוני שהמשתמש הקיש למספר בינארי . את המספר הבינארי היא מדפיסה במסך . ההמרה מתבצעת על ידי לולאת while המחלקת ב 2 ואת השארית של החלוקה ( 0 או 1 ) היא מכניסה למערך arr אחד אחרי השני. הלולאה מסתיימת כאשר המספר הוא 0 .

ד. לולאת ה while החדשה מבצעת תהליך דומה ללולאת ה while הקודמת אבל כאן מחלקים את המספר ב 8 ובכל פעם רושמים את השארית במערך. במקרה זה המרנו את המספר לאוקטלי. הטבלה הבאה מתארת את החלוקה :

num	23			
i		<b>0</b>	<b>1</b>	<b>2</b>
mod=mod%8		7	2	
div =div/8	23	2	0	
arr[0]		7		
arr[1]			2	
arr[2]				
arr[3]				
arr[4]				
arr[5]				
-----				
arr[99]				

המספר שיודפס במקרה הזה הוא : The transformed Number is : 27

## שאלה 6

כתבו תוכנית שקולטת עשרה גבהים של משולש ועשרה בסיסים של משולש.

התוכנית תחשב ותדפיס עליגבי צג המחשב:

- את שטחו של כל אחד מהמשולשים

- את שטח המשולש הגדול ביותר

לוו את התוכנית בהערות הבהרה לכל שורה.

## פתרון תרגיל 6

התוכנית עם הנתונים שהכנסתי (שנראים בצבע אדום ב watches):

```

5 #include <stdio.h>
6 int main()
7 {
8     float height[10], base[10], area[10],maxArea;
9     int i,indexOfMaxTriangle;
10    //base[] ו height[] לתעריכים אותם למערכים
11    for(i=0;i<10;i++)
12    {
13        printf("Please enter the height of triangle # %d : ",i+1);
14        scanf("%f",&height[i]);
15        printf("Please enter the base of triangle # %d : ",i+1);
16        scanf("%f",&base[i]);
17        // area[] לחשב את שטח המשולשים ונכניס אותם למערך השטח
18        area[i]=height[i]*base[i];
19    }
20    // נבצע הדפסה של כל המשולשים ושטחם
21    printf("\n height\t\t base\t\t area\n");
22    for(i=0;i<10;i++)
23    {
24        printf(" %f\t%f\t%f\n",height[i],base[i],area[i]);
25    }
26    // נמצא מהו שטח המשולש הגדול ביותר בהנחה שיש אחד בלבד כזה
27    maxArea=area[0]; // הוא בעל השטח הגדול ביותר
28
29    for(i=1;i<10;i++)
30    {
31        if (maxArea<area[i])
32        {
33            maxArea=area[i];
34            indexOfMaxTriangle=i;
35        }
36    }
37    printf("\nThe triangle with the maximum area is : height = %f\t base=%f\t area = %f ",
38        height[indexOfMaxTriangle],base[indexOfMaxTriangle],area[indexOfMaxTriangle]);
39    return 0;
40 }
41

```

Function arguments	
Locals	
height	
[0]	43.2099991
[1]	123.123001
[2]	456.119995
[3]	123.230003
[4]	234.455994
[5]	678.890015
[6]	34.5600014
[7]	234.559998
[8]	34.5600014
[9]	234.876007
base	
[0]	23.4200001
[1]	32.3199997
[2]	23.4500008
[3]	29.8899994
[4]	123.339996
[5]	13.4499998
[6]	123.120003
[7]	123.980003
[8]	56.7799988
[9]	120.120003
area	
[0]	1011.97821
[1]	3979.33545
[2]	10696.0146
[3]	3683.34473
[4]	28917.8008
[5]	9131.07031
[6]	4255.02734
[7]	29080.75
[8]	1962.31689
[9]	28213.3066
maxArea	29080.75
i	10
indexOfMaxTriangle	7

```

c:\targilim\c\testTechnaem2023\bin\Debug\testTechnaem2023.exe
Please enter the height of triangle # 3 : 456.12
Please enter the base of triangle # 3 : 23.45
Please enter the height of triangle # 4 : 123.23
Please enter the base of triangle # 4 : 29.89
Please enter the height of triangle # 5 : 234.456
Please enter the base of triangle # 5 : 123.34
Please enter the height of triangle # 6 : 678.89
Please enter the base of triangle # 6 : 13.45
Please enter the height of triangle # 7 : 34.56
Please enter the base of triangle # 7 : 123.12
Please enter the height of triangle # 8 : 234.56
Please enter the base of triangle # 8 : 123.98
Please enter the height of triangle # 9 : 34.56
Please enter the base of triangle # 9 : 56.78
Please enter the height of triangle # 10 : 234.876
Please enter the base of triangle # 10 : 120.12

height      base      area
43.209999   23.420000 1011.978210
123.123001  32.320000 3979.335449
456.119995  23.450001 10696.014648
123.230003  29.889999 3683.344727
234.455994  123.339996 28917.800781
678.890015  13.450000 9131.070313
34.560001   123.120003 4255.027344
234.559998  123.980003 29080.750000
34.560001   56.779999 1962.316895
234.876007  120.120003 28213.306641

The triangle with the maximum area is : height = 234.559998   base=123.980003   area = 29080.750000

```

## שאלה 7

- א. (15 נק') כתבו פונקצייה בשפת C, בשם `isPassword`, המקבלת מחרוזת. הפונקצייה תבדוק שהמחרוזת מהווה סיסמה חוקית. אם הסיסמה חוקית, היא תחזיר "TRUE". אחרת, היא תחזיר "FALSE".
- סיסמה חוקית היא מחרוזת באורך של בין 8 תווים ל-12 תווים (כולל), המכילה לפחות אות אחת קטנה ואות אחת גדולה, לפחות ספרה אחת ולפחות סימן אחד מיוחד מביין הסימנים הבאים: (!, @, #, \$).
- ב. (10 נק') כתבו תוכנית ראשית שתקלוט מחרוזת מהמקלדת. התוכנית תוודא שהמחרוזת שנקלטה חוקית בעזרת הפונקצייה `isPassword`. אם המחרוזת לא חוקית, יש להודיע על כך למשתמש ולאצו להזין סיסמה אחרת, שתהיה חוקית.

## פתרון שאלה 7

- יש לבצע את הבדיקות הבאות:
- שכמות התווים בין 8 ל 12

2. שמופיע לפחות תו קטן אחד

3. שמופיע לפחות תו גדול אחד

4. שמופיע לפחות ספרה אחת בין 0 ל 9

5. שמופיע לפחות אחד מ 4 תווי הבקרה הבאים : \$ # @ !

6. שלא מופיעים תווים אחרים כמו ~ או ^ וכו .

נשתמש במשתנה בשם check שערכו ההתחלתי 0 . כאשר עוברים אחת מ 6 הבדיקות שלמעלה נעלה את ערכו ב 1 . בסיום נבדוק האם ערכו הוא 6 ? אם כן הסיסמה חוקית ואם לא נבקש מהמשתמש להקיש שוב סיסמה.

```
/*
```

```
תוכנית הבודקת האם סיסמה חוקית
```

```
סיסמה חוקית היא סיסמה שבה יש בין 8 ל 12 תווים
```

```
מכילה לפחות אות אחת קטנה , לפחות אות אחת גדולה, לפחות ספרה אחת
```

```
ולפחות את אחד מהתווים $ # @ !
```

```
*/
```

```
#include <stdio.h>
```

```
int isPassword(char sisma[])
```

```
{
```

```
// המשתנה check בודק את התנאים אחד אחרי השני. כל תנאי המתקיים מגדילים אותו ב 1
```

```
int i,check=0;
```

```
for(i=0;sisma[i];i++); // נבדוק את אורך המחרוזת
```

```
if(i>=8 && i<=12) // האם אורך המחרוזת חוקי ?
```

```
check=1;
```

```
// נבדוק שאחד התווים הוא אות קטנה
```

```
for(i=0;check==1 && sisma[i];i++)
```

```
{
```

```
if(sisma[i]>='a' && sisma[i]<='z') //
```

```
check=2;
```

```
}
```

```
// נבדוק האם אחד התווים הוא אות גדולה
```

```
for(i=0;check==2 && sisma[i];i++)
```

```
{
```

```
if(sisma[i]>='A' && sisma[i]<='Z') //
```

```
check=3;
```

```
}  
// נבדוק שאחד התווים לפחות הוא ספרה בין 0 ל 9  
for(i=0;check==3 && sisma[i];i++)  
{  
    if( sisma[i]>='0' && sisma[i]<='9') // ? 9 ל 0 בין  
        check=4;  
}  
// נבדוק שאחד התווים לפחות הוא אחד מתווי הבקרה הרצויים  
for(i=0; check==4 && sisma[i];i++)  
{  
    if( sisma[i]=='!' || sisma[i]<='@' || sisma[1]=='#' || sisma[i]<='$') //  
        check=5;  
}  
// נבדוק שלא מופיע תו לא רצוי  
for(i=0;check == 5 && sisma[i] ;i++)  
{  
    if(sisma[i]!='!' && sisma[i]!='@' && sisma[1]!='#' &&  
        sisma[i]!='$' && !(sisma[i]>='a' && sisma[i]<='z') &&  
        !(sisma[i]>='A' && sisma[i]<='Z') &&  
        !(sisma[i]>='0' && sisma[i]<='9') )  
        check=6;  
}  
if ( check==6 ) // האם כל הבדיקות היו תקינות ?  
    return 1;  
else  
    return 0 ;  
}  
  
int main()  
{  
    char password[13];
```

```
printf("Please Enter Password : ");
gets(password);
while(!isPassword(password)) // כל עוד הערך החוזר מהפונקציה שבסעיף הקודם הוא 0 כלומר מחרוזת לא חוקית לא
{
    printf("Wrong password !! Try again\n");
    printf("Please Enter Password : ");
    gets(password);
}
printf("Good password .....");
return 0;
}
```

## שאלה 8

להלן קטע מתוכנית הכתובה בשפת C:

```
1. float avg (int * arr , int size )
2. {
3.     int j,sum=0;
4.     for (j=0;j<size;j++)
5.     {
6.         sum+=arr[j];
7.     }
8.     return( ( float ) sum/size);
9. }
```

- א. (5 נק') הסבירו את התהליך המתבצע עלידי הפונקצייה avg.
- ב. (5 נק') מה היה קורה אילו היו מסירים את הסימונים { ו-} משורות 5 ו-7 ?
- ג. (5 נק') הסבירו את משמעות הפרמטרים המועברים לפונקצייה avg וההכרזה אליה.
- ד. (5 נק') הסבירו את המושג CASTING וציינו היכן הוא בא לידי ביטוי בתוכנית.
- ה. (5 נק') הסירו את הביטוי (float) משורה 8. הסבירו את משמעות השינוי.

## פתרון שאלה 8

- א. הפונקציה מקבלת 2 נתונים : כתובת הנכנסת למצביע מטיפוס שלם ששמו arr וערך מטיפוס שלם למשתנה בשם size . הפונקציה מחזירה ערך מטיפוס ממשי ( float ) . בהמשך הפונקציה רואים כי הכתובת שנכנסה למצביע היא למעשה שם של מערך שבו size מציין את כמות האיברים במערך . **שם של מערך הוא למעשה הכתובת של האיבר הראשון של המערך !** בהמשך הפונקציה משתמשים במצביע ככתובת הראשונה במערך ועובדים עם j כאינדקס במערך . כלומר כשרושמים arr[j] לוקחים את הכתובת שבמצביע ומוסיפים לו j כתובות לוגיות ( כתובת לוגית של int היא 4 כתובות פיזיות) .
- הערה :** ניתן היה לעבוד עם arr כמצביע (פוינטר) ובמקום לקדם את j לבצע arr++ . בשורה 3 מגדירים 2 משתנים מטיפוס שלם בשם j ו sum ומאתחלים sum=0 . בשורה 4 מבצעים לולאת for של size פעמים ובה מסכמים אחד אחרי השני את כל הערכים שבמערך arr בעזרת השורה הבאה:
- שורה 5 : `sum+=arr[j];` שהיא כמו `sum = sum + arr[j];` . מוסיפים ל sum את הערך של האיבר במיקום j שבמערך .
- בשורה 8 מחזירים לשורה שקראה לפונקציה avg את היחס בין sum (סכום כל האיברים במערך) ובין size שהוא גודל המערך . כלומר מחזירים את הממוצע של האיברים במערך . לפני שמבצעים את החלוקה עושים לערך שבמשתנה sum ליהוק/הסבה או באנגלית casting שבו משנים את הערך שיש בו מטיפוס שלם לממשי . כך לדוגמה אם הערך שבו הוא 581 אז הערך החדש שעליו תבוצע החלוקה יהיה 581.000 . במקרה כזה החלוקה תיתן את הממוצע כערך מטיפוס ממשי .
- לדוגמה: עבור המערך : {90,67,78,87,89,91,77}; שבו 7 איברים ( size=7 ) נקבל בסיום הפונקציה את הערכים הבאים :
- ב. התו { פותח בלוק הוראות והתו } סוגר בלוק הוראות . במקרה כאן לא היה שינוי בתוכנית כי בבלוק ההוראות יש רק הוראה אחת (שורה 6) ולכן אין חובה לרשום תווים אלו. יש לשים תווים אלו אם יש יותר מהוראה אחת בבלוק ההוראות של הלולאה for .
- ג. הפרמטרים המועברים לפונקציה הם כתובת וערך מטיפוס שלם . בהמשך הפונקציה רואים שהכתובת המועברת היא של האיבר הראשון במערך כלשהו והערך השני הוא כמות האיברים באותו מערך .
- ההכרזה/הצהרה על הפונקציה בשורה 1: `float avg (int *arr, int size)` אומרת שמתקבלת כתובת למצביע ( פוינטר ) arr וערך למשתנה size . שניהם מטיפוס שלם int . הפונקציה מחזירה ערך מטיפוס ממשי float .
- ד. המושג casting באנגלית הוא ליהוק/הסבה . נותנים לערך מטיפוס כלשהו ( לדוגמה int ) לקבל ערך מטיפוס אחר (לדוגמה float) . בשורה 8 בפונקציה עושים casting מלהקים/עושים הסבה לערך שיש ב sum מערך מטיפוס int לערך מטיפוס float כדי שהחלוקה בין float/int תיתן float . אם לדוגמה sum=581 ו size=7 ולא היינו מבצעים casting אז היינו מקבלים  $84 = 579/7$  כי int/int נותן int ואז החלק של השבר אחרי הנקודה העשרונית לא היה מתקבל . כשעושים casting נקבל : 82.714285 .



האיור הבא מתאר תוכנית כולל הפונקציה main שבה מגדירים מערך בן 7 איברים וקוראים לפונקציה avg. מימין באיור רואים את טבלת ה watches בסיום הפונקציה avg.

```

/*
תוכנית הבדקה היא פשוטה ונוחה
סימטה חוקית היא סימטה שבה יש בין 8 ל 12 תווים
מכילה לפחות אות אחת סימטה, לפחות אות אחת גדולה, לפחות ספרה אחת
ולפחות את אחד מהתווים ! @ # $
*/
#include <stdio.h>
float avg (int *arr, int size)
{
    int j, sum=0;
    for (j=0;j<size;j++)
    {
        sum+=arr[j];
    }
    return( ( float ) sum/size);
}

int main()
{
    int array[7]={90,67,78,87,89,91,77};
    float average;
    average=avg(array,7);
    printf("The average of all elements in array is : %f",average);
    return 0;
}
    
```

הגדרת מערך בן 7 איברים

קריאה/זימון לפונקציה avg

Watches	
Function arguments	
arr	0x61fe00
size	7
Locals	
j	7
sum	579

בסיום התוכנית נקבל את הערכים הבאים בטבלת ה watches ואת ההדפסה למסך :

```

/*
תוכנית הבדקה היא פשוטה ונוחה
סימטה חוקית היא סימטה שבה יש בין 8 ל 12 תווים
מכילה לפחות אות אחת סימטה, לפחות אות אחת גדולה, לפחות ספרה אחת
ולפחות את אחד מהתווים ! @ # $
*/
#include <stdio.h>
float avg (int *arr, int size)
{
    int j, sum=0;
    for (j=0;j<size;j++)
    {
        sum+=arr[j];
    }
    return( ( float ) sum/size);
}

int main()
{
    int array[7]={90,67,78,87,89,91,77};
    float average;
    average=avg(array,7);
    printf("The average of all elements in array is : %f",average);
    return 0;
}
    
```

Watches	
Function arguments	
Locals	
array	
[0]	90
[1]	67
[2]	78
[3]	87
[4]	89
[5]	91
[6]	77
average	82.7142868

```

c:\targilim\c\testTechnaem2023\bin\Debug\testTec...
The average of all elements in array is : 82.714287
    
```