**פונקציות בשפת C**

**תרגול מקוון מתאריך 2/6/2020**

קיימים 4 סוגי פונקציות :

1. פונקציה שמקבלת ערכים ולא מחזירה ערך.
2. פונקציה שלא מקבלת ולא מחזירה.
3. פונקציה שמקבלת ערכים ומחזירה ערך.
4. פונקציה שלא מקבלת אך מחזירה ערך.

**דוגמא א' - פונקציה שמקבלת ערכים ולא מחזירה ערך**

#include <stdio.h>

void printStars(int n); // הצהרה על הפונקציה

**int main()**

{

int n;

printf("Enter the first number : ");

scanf("%d",&n);

printStars(n) ; // קריאה לפונקציה

printf("\nEnter the second number : ");

scanf("%d",&n);

printStars(n) ; // קריאה לפונקציה

printf("\nEnter the third number : ");

scanf("%d",&n); // קריאה לפונקציה

}

**void printStars(int n) // פונקציה המקבלת ערך ומדפיסה שורת כוכביות**

{

int i;

for(i=1 ; i<=x ; i++)

printf("\*");

}

**דוגמא ב' - פונקציה שלא מקבלת ערכים ולא מחזירה ערך**

#include <stdio.h>

void printStars(); // decleration

**void main()**

{

printf("Enter the first number : ");

printStars() ; // קריאה לפונקציה

printf("\nEnter the second number : ");

printStars() ; // קריאה לפונקציה

printf("\nEnter the third number : ");

printStars() ; // קריאה לפונקציה

}

**void printStars() פונקציה הקולטת ערך ומדפיסה שורת כוכביות**

{

int n,i;

scanf("%d",&n);

for(i=1 ; i<=n ; i++)

printf("\*");

}

Output :

Enter the first number : 3

\*\*\*

Enter the second number : 18

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter the third number : 6

\*\*\*\*\*\*

**דוגמא ג' - פונקציה שמקבלת ערכים ולא מחזירה ערך**

#include <stdio.h>

void sumAB(int x, int y); // הצהרה על הפונקציה

void mulAB(int x, int y); // הצהרה על הפונקציה

**void main()**

{

int a,b ;

printf("Enter a : ");

scanf("%d",&a);

printf("Enter b : ");

scanf("%d",&b);

sumAB(a,b); // קריאה לפונקציה

mulAB(a,b); // קריאה לפונקציה

}

**void sumAB(int x, int y) // מקבלת שני ערכים ומדפיסה את סכומם**

{

int s ;

s=x+y;

printf("sum = %d\n",s);

}

**void mulAB(int x, int y) // מקבלת שני ערכים ומדפיסה את מכפלתם**

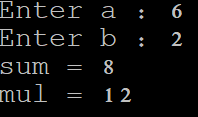
{

int m ;

m=x\*y;

printf("mul = %d\n",m);

}



**דוגמא ד' - פונקציה שמקבלת ערכים ומחזירה ערך**

#include <stdio.h>

int sumAB(int x, int y); // הצהרה על הפונקציה

int mulAB(int x, int y); // הצהרה על הפונקציה

**void main()**

{

int a,b,sum,mul ;

printf("Enter a : ");

scanf("%d",&a);

printf("Enter b : ");

scanf("%d",&b);

sum=sumAB(a,b); // קריאה לפונקציה

printf("sum = %d\n",sum);

mul=mulAB(a,b); // קריאה לפונקציה

printf("mul = %d\n",mul);

}

**int sumAB(int x, int y) // מקבלת שני ערכים ומחזירה את סכומם**

{

int s ;

s=x+y;

return s ; // ערך מוחזר

}

**int mulAB(int x, int y) // מקבלת שני ערכים מחזירה את מכפלתם**

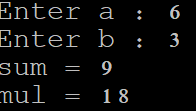
{

int m ;

m=x\*y;

return m; // ערך מוחזר

}



**דוגמא ה' - פונקציה שלא מקבלת ערכים ולא מחזירה ערך**

התכנית הבאה קולטת סדרת מספרים עד לקליטת 0 ומדפיסה את סכומם והממוצע שלהם.

#include <stdio.h>

void func(); // הצהרה על הפונקציה

**int main()**

{

func(); // קריאה לפונקציה

}

**void func()**

{

int num,sum=0, cnt=0 ;

float avg;

printf("Enter the first number (0 to quit) : ");

scanf("%d",&num);

while(num)

{

sum += num ;

cnt++;

printf("Enter the next number (0 to quit) : ");

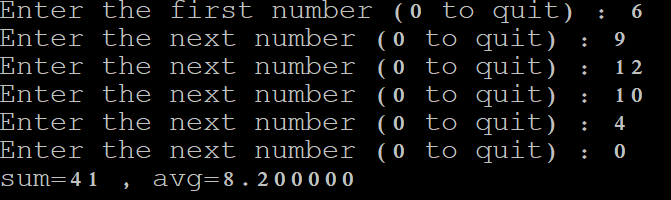
scanf("%d",&num);

}

avg=(float)sum/cnt ;

printf("sum=%d , avg=%f",sum,avg);

}



**דוגמא ו'**

התכנית הבאה קולטת שני מספרים : אם שניהם זוגיים, מדפיסה את סכומם. אחרת, מדפיסה את מכפלתם. לשם כך היא נעזרת בשתי פונקציות עזר, בשיטת bottom-up :

#include <stdio.h>

**int isEven(int num) // הצהרה + כותרת של הפונקציה**

{

if (num%2==0)

return 1;

else

return 0;

}

**int fun() // הצהרה + כותרת של הפונקציה**

{

int x,y,check1, check2;

printf("Enter 2 numbers : ");

scanf("%d %d",&x,&y);

check1=isEven(x); // קריאה לפונקציה

check2=isEven(y); // קריאה לפונקציה

if(check1==1 && check2==1)

return (x+y); // ערך מוחזר

else

return (x-y); // ערך מוחזר

}

**void main()**

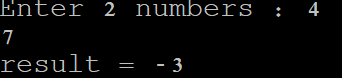
{

int result ;

result=fun(); // קריאה לפונקציה

printf("result = %d\n",result);

}



אותה התכנית בשיטת top-down :

#include <stdio.h>

int fun(); // הצהרה על הפונקציה

int isEven(int num); // הצהרה על הפונקציה

**void main()**

{

int result ;

result=fun(); // קריאה לפונקציה

printf("result = %d\n",result);

}

**int fun()**

{

int x,y,check1, check2;

printf("Enter 2 numbers : ");

scanf("%d %d",&x,&y);

check1=isEven(x); // קריאה לפונקציה

check2=isEven(y); // קריאה לפונקציה

if(check1==1 && check2==1)

return (x+y); // ערך מוחזר

else

return (x-y); // ערך מוחזר

}

**int isEven(int num)**

{

if (num%2==0)

return 1; // ערך מוחזר

else

return 0; // ערך מוחזר

}

**תרגיל**

כתוב תכנית הקולטת מספר שלם n ובודקת האם הוא שווה או גדול מ-0 :

אם לא – מדפיסה הודעת שגיאה.

אם כן – מדפיסה את העצרת שלו n!

לדוגמא :

עבור n=-3 יודפס : "ERROR!"

עבור n=5 יודפס : 120. הסבר : 5! = 1\*2\*3\*4\*5=120

עבור n=3 יודפס : 6

עבור n=0 יודפס : 1

**פתרון ללא פונקציות :**

#include <stdio.h>

void main()

{

int n,result=1,x ;

printf("Enter n : ");

scanf("%d",&n);

if (n>=0)

{

for (x=1 ; x<=n ; x++) //x=1,2,3,4,5,6

{

result = result \* x ; // result=1\*1\*2\*3\*4\*5=120

}

printf("result=%d",result);

}

else

printf("ERROR!");

}

**פתרון עם פונקציית עזר אחת :**

#include <stdio.h>

int fun(int num) ; // הצהרה על הפונקציה

**void main()**

{

int n,result=1 ;

printf("Enter n : ");

scanf("%d",&n);

if (n>=0)

{

result = fun(n); // קריאה לפונקציה

printf("result=%d",result);

}

else

printf("ERROR!");

}

**int fun(int num) // פונקציה המקבלת ערך ומחזירה את העצרת שלו**

{

int r=1, x ;

for (x=1 ; x<=num ; x++)

r\*=x ; //r=r\*x;

return r ; // ערך מוחזר

}

**פתרון עם מספר פונקציות עזר :**

#include <stdio.h>

void main()

{

int n,result=1 ;

n=kelet(); // קריאה לפונקציה

if(isLegal(n)==1) // קריאה לפונקציה

{

result = fun(n); // קריאה לפונקציה

printf("result=%d",result);

}

else

printf("ERROR!");

}

**int kelet() // פונקציה הקולטת ערך ומחזירה אותו**

{

int n;

printf("Enter n : ");

scanf("%d",&n);

return n; // ערך מוחזר

}

**int isLegal(int num) // פונקציה המקבלת ערך ומחזירה 1-זוגי, 0-אי זוגי**

{

if (num>=0)

return 1; // ערך מוחזר

else

return 0 ; // ערך מוחזר

}

**int fun(int num) // פונקציה המקבלת ערך ומחזירה את העצרת שלו**

{

int r=1, x ;

for (x=1 ; x<=num ; x++)

r\*=x ; //r=r\*x;

return r; // ערך מוחזר

}

**פתרון עם משתנה גלובלי :**

#include <stdio.h>

int n ; // global variable

~~0~~ , 5

**void main()**

{

int result=1 ;

kelet(); // קריאה לפונקציה

if(isLegal()==1) // קריאה לפונקציה

{

result = fun(); // קריאה לפונקציה

printf("result=%d",result);

}

else

printf("ERROR!");

}

**void kelet()**

{

printf("Enter n : ");

scanf("%d",&n);

}

**int isLegal()**

{

if (n >=0)

return 1; // ערך מוחזר

else

return 0 ; // ערך מוחזר

}

**int fun() // function header**

{

int r=1, x ;

for (x=1 ; x<= n ; x++)

r\*=x ; //r=r\*x;

return r; // ערך מוחזר

}

הערה : ניתן לקרוא על משתנה גלובלי במודל