

Використання функцій.

Визначення власних функцій.

Визначення функції

тип поверненого значення
ім'я функції (список параметрів)
{
об'яви
оператори
}

Приклад 1

Завдання. Створити функцію, яка допомагає вирішувати квадратне рівняння

Текст програми

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

float f1(float x,float y,float z) {
    float k;
    if (x==0 && y==0&& z==0) { printf ("Error!!!"); return -1;

} else k=y*y-4*x*z;

    if (k<0) {printf("ERROR\n");return k ;

}
    else {k=sqrt(k);printf("%f\n",k);return k;}
}
```

```
int main() {  
float a,b,c,d,q,x1,x2;  
int r;  
printf("a= ");  
scanf("%f",&a);  
printf("b= ");  
scanf("%f",&b);  
printf("c= ");  
scanf("%f",&c);  
q=f1(a,b,c);  
if(q>=0){  
x1=(-b+q)/(2*a);  
x2=(-b-q)/(2*a);  
printf("x1= %f , x2= %f",x1,x2); }  
return(0);  
}
```

Результат роботи

```
a= 1  
b= 5  
c= 2  
x1= -0.438447 , x2= -4.561553
```

Текст програми 2

```
#include <stdio.h>
#include <math.h>
void f2(float x,float y,float z);

int main() {
float a,b,c;
    printf("a= ");
    scanf("%f",&a);
    printf("b= ");
    scanf("%f",&b);
    printf("c= ");
    scanf("%f",&c);
    f2(a,b,c);
return(0);
}
```

```
void f2(float x,float y,float z) {  
    float k,x1,x2;  
    if (x==0 && y==0&& z==0) { printf ("Error!!!");  
} else k=y*y-4*x*z;  
if (k<0) printf("ERROR\n");  
else {k=sqrt(k);  
    x1=(-y+k)/(2*x);  
    x2=(-y-k)/(2*x);  
printf("x1= %f , x2= %f",x1,x2);  
    }  
}
```



```
a= 2  
b= 5  
c= 1  
x1= -0.219224 , x2= -2.280777_
```

```
a= 5  
b= 2  
c= 5  
ERROR  
_
```

```
a= 0  
b= 0  
c= 0  
Error!!!_
```

Приклад 2

Завдання. Написати функцію, яка знаходить більше з двох чисел.

Текст програми

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

```
float maxm (float x, float y); //прототип функції
```

```
void main () {
```

```
float a,b,m;
```

```
    printf("\nEnter a=");
```

```
    scanf("%f",&a);
```

```
    printf("Enter b=");
```

```
    scanf("%f",&b);
```

```
    printf("max=%f", maxm(a,b));    //виклик функції
```

```
}
```

```
float maxm(float x,float y){ //заголовок функції
```

```
if (x>y) return x;
```

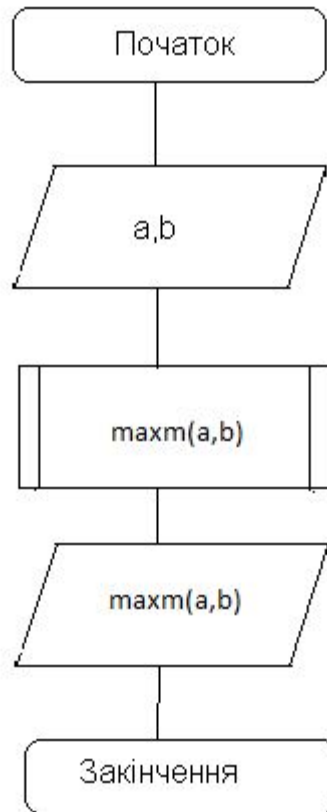
```
else return y;
```

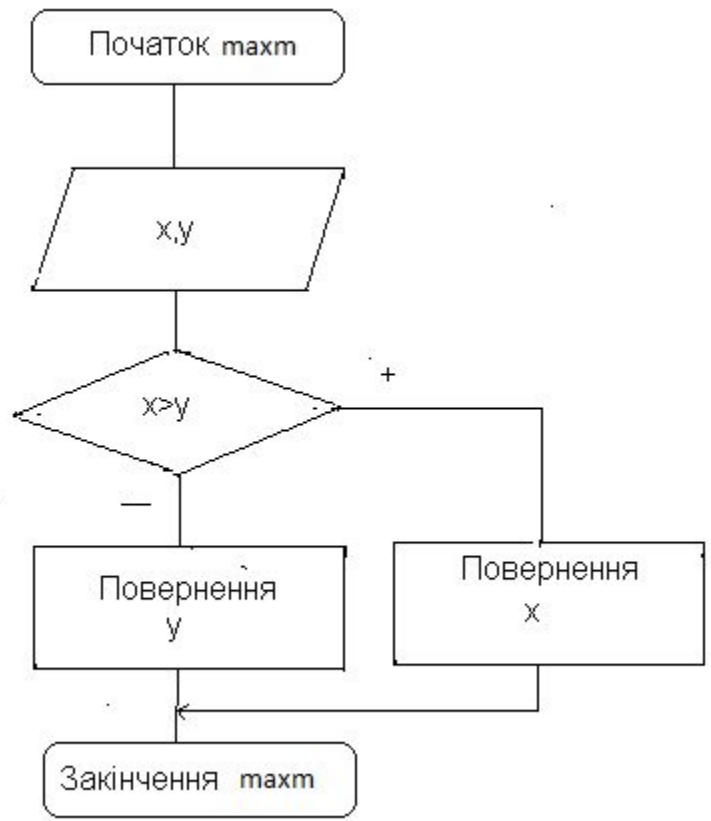
```
}
```

Результат роботи програми

```
Enter a=2.234  
Enter b=2.236  
max=2.2360  
Process returned 10 (0xA)   execution time : 11.175 s  
Press any key to continue.  
-
```

```
Enter a=-23  
Enter b=8.32  
max=8.3200  
Process returned 10 (0xA)   execution time : 9.114 s  
Press any key to continue.  
-
```





Приклад 3

Завдання. Написати функцію, яка знаходить більше з трьох чисел.

Текст програми

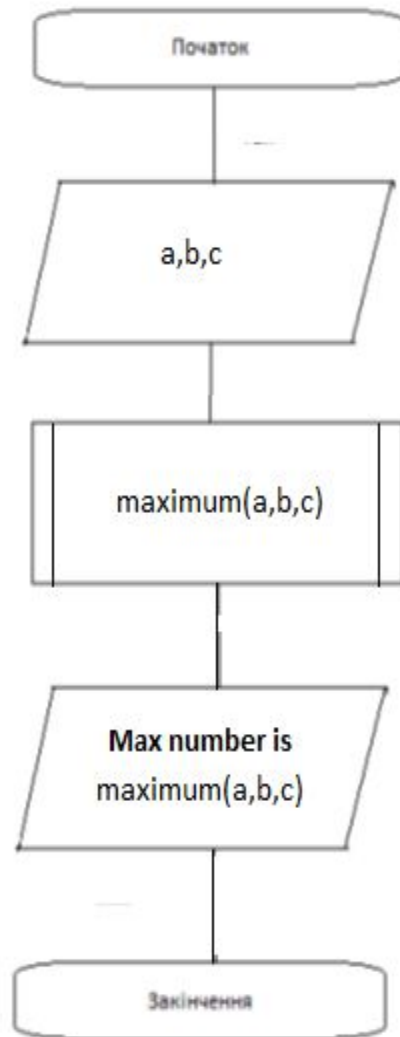
```
#include <stdio.h>
int maximum(int x, int y, int z);
main ()
{
int a, b, c;
printf ("Input 3 numbers:");
scanf ("%d %d %d", &a, &b, &c);
printf ("Max number is : %d\n", maximum(a, b,
c));
return 0;
}
```

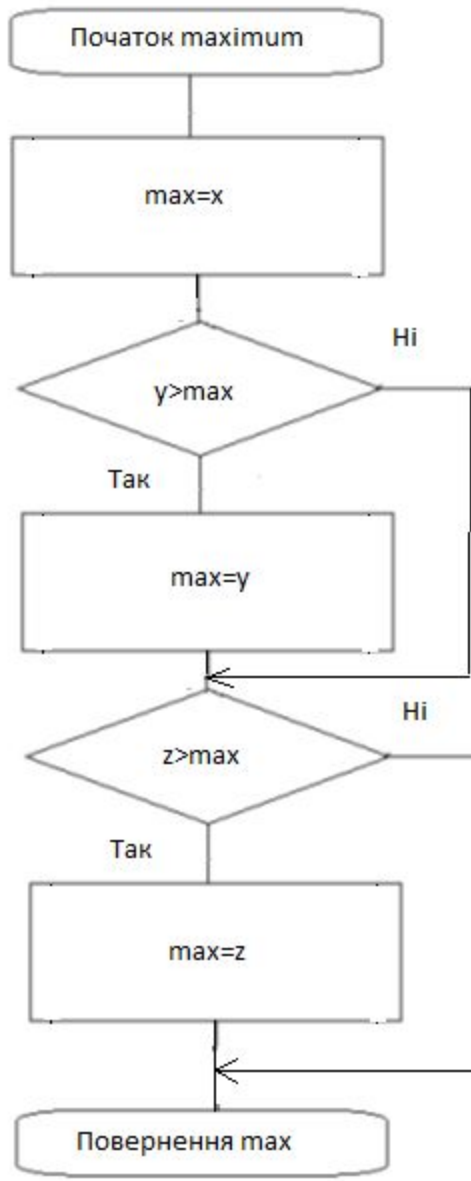


```
/*Визначення функції maximum*/  
int maximum(int x, int y, int z)  
{ int max=x;  
    if (y > max)  
        max=y;  
        if (z > max)  
            max=z;  
    return max;  
}
```

Результат роботи програми

```
Input 3 numbers:3 7 1  
Max number is : 7  
-
```





Текст програми 2

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

```
float maxm (float x, float y);
```

```
void main () {
```

```
float a,b,c,m1,m2;
```

```
printf("\nEnter a=");  
scanf("%f",&a);  
printf("Enter b=");  
scanf("%f",&b);  
printf("Enter c=");  
scanf("%f",&c);
```

```
m1=maxm(a,b);  
m2=maxm(c,m1);  
printf("max=%.3f", m2);  
}  
float maxm(float x,float y)  
{  
if (x>y) return x;  
else return y;  
}
```

```
Enter a=3
Enter b=7
Enter c=4,5
max=7.000
Process returned 9 (0x9)   execution time : 11.639 s
Press any key to continue.
```

```
Enter a=-9
Enter b=4
Enter c=6.65
max=6.650
Process returned 9 (0x9)   execution time : 11.906 s
Press any key to continue.
```

```
Enter a=-2
Enter b=-5
Enter c=-8
max=-2.000
Process returned 10 (0xA)  execution time : 9.989 s
Press any key to continue.
```



```
printf("\nEnter a=");  
scanf("%f",&a);  
printf("Enter b=");  
scanf("%f",&b);  
printf("Enter c=");  
scanf("%f",&c);  
printf("max=%.3f", maxm((maxm(a,b)),c));
```

Enter a=78

Enter b=32

Enter c=12

max=78.000

Process returned 10 (0xA) execution time : 10.567 s

Press any key to continue.

—

Приклад 4

Завдання. Необхідно розробити функцію, яка виводить певний символ, повторюючи його стільки разів, скільки необхідно користувачеві. Використати цю функцію у програмі, яка реалізує виведення різних символів різну кількість разів. Виведення кожного символу здійснюється в окремому рядку, починаючи з першого рядка екрану. Запрошення користувачеві ввести символ і кількість повторення вивести в рядках 22 і 23. Завершити виконання програми, якщо користувач натисне клавішу **ESC**

Текст програми

```
#include <conio.h>
```

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

```
void repChar (char ch, int n, int line); //прототип  
функції
```

```
void main () {
```

```
int count, symbol, line=0;
```

```
textmode(3);
```

```
do {  
    gotoxy(1,22);  
    printf ("Enter symbol->");  
    symbol=getche();  
    if (symbol!=27) {  
        printf("\nCount->");  
        scanf("%d",&count);  
        line++;  
    }  
}
```

```
repChar(symbol, count, line); //виклик функції
    }
else {
    printf("\nExit. Press any key.");
    }
} while (symbol!=27);
getch();
}
```

```
//заголовок функції
```

```
void repChar (char ch, int n, int line)
```

```
{int j;
```

```
gotoxy(1,line);
```

```
//тіло функції
```

```
j=n/80; line+=j;
```

```
for (j=0; j<n; j++) {
```

```
putchar(ch);
```

```
}
```

```
}
```


Функція `textmode (mode)` використовується для редагування відеорежима текстового екрана. Режим `mode` може використовувати одне з значень, показаних далі. Можливо використовувати або ціле значення, або відповідний макрос, який знаходиться у файлах `conio.h`:

BW40	0	40-стовпців чорно-білий
C40	1	40-стовпців кольоровий
BW80	2	80-стовпців чорно-білий
C80	3	80-стовпців кольоровий
MONO	1	80-стовпців монохромний
C4350	64	43-рядки EGA або 50- рядків VGA
LASTMODE	—1	Попередній режим

Функція `gotoxy ()` поміщає курсор текстового екрану в точку з координатами `x, y`. Якщо хоча б одна з координат недійсна, то ніяких дій не виконується.

Але ці функції можуть не працювати в деяких компіляторах. Тому пропоную наступний приклад з використанням власної функції

```
gotoxy1(x,y)
```

```
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
void gotoxy1(int x,int y)
{
    int i,j;
    for (i=1;i<=y;i++){
        for (j=1;j<=x;j++)
            {printf(" \n");}
    }
}
void repChar (char ch, int n, int line); //прототип функції
```

```
void main () {
int count, symbol, line=0;

do {
    gotoxy1(1,22);

    printf ("Enter symbol->");
    symbol=getche();
    if (symbol!=27) {
        printf("\nCount->");
        scanf("%d",&count);
        //line++;
        repChar(symbol, count, line); //виклик функції
    }
} else {
    printf("\nExit. Press any key.");
}
} while (symbol!=27);
getch();
}
```

```
void repChar (char ch, int n, int line)
{int j ;
system ("cls");
gotoxy1(1,line);           //тіло функції
for (j=0; j<n; j++) {
putchar(ch);
if ((j%19)==0&&j!=0) printf("\n");
}
}
```

```
Enter symbol->8  
Count->60
```

