

# Homework 3

+

## New topic C Programming break and continue Statement

---

### 1. break Statement

In C programming, break is used in terminating the loop immediately after it is encountered. The break statement is used with conditional if statement.

Syntax of break statement

**break;**

The break statement can be used in terminating all three loops for, while and do...while loops.

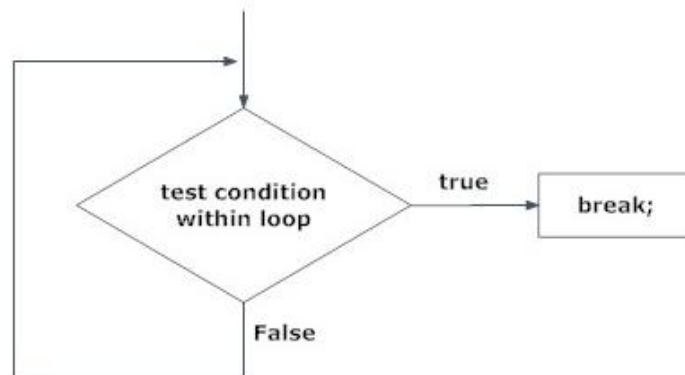
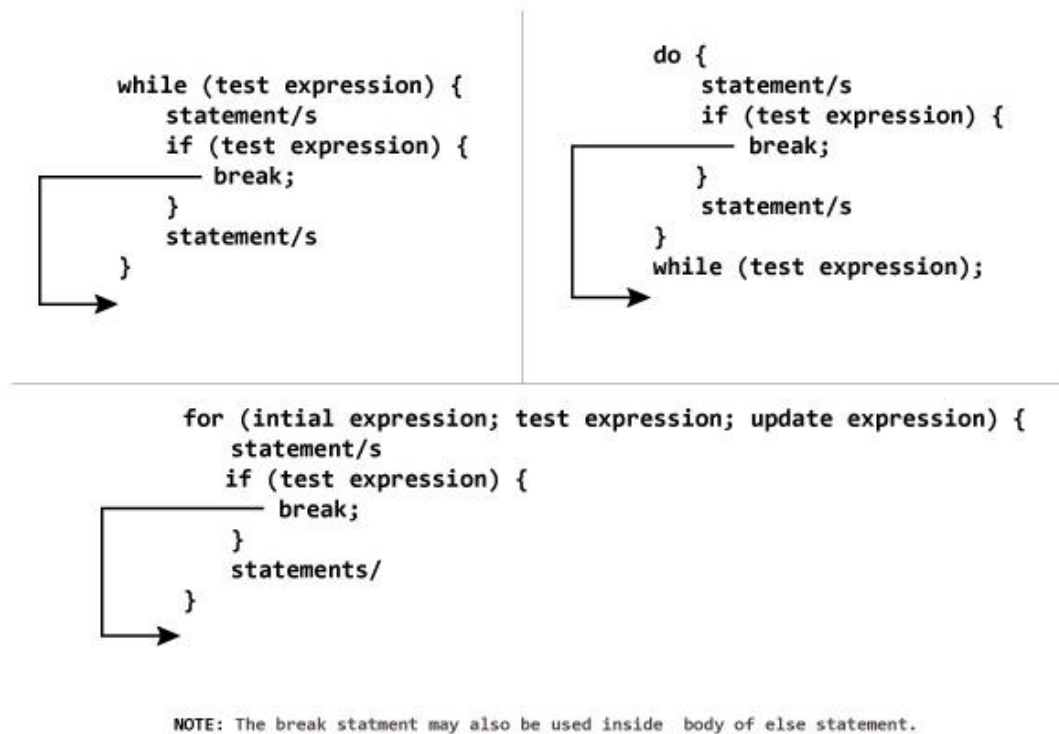


Figure: Flowchart of break statement

The figure below explains the working of break statement in all three type of loops.



## Example of break statement

Write a C program to find average of maximum of n positive numbers entered by user. But, if the input is negative, display the average(excluding the average of negative input) and end the program.

```

/* C program to demonstrate the working of break statement by terminating a
loop, if user inputs negative number*/
#include <stdio.h>
int main(){
    float num,average,sum;
    int i,n;
    printf("Maximum no. of inputs\n");
    scanf("%d",&n);
    for(i=1;i<=n;++i){
        printf("Enter n%d: ",i);
        scanf("%f",&num);
        if(num<0.0)
            break;
        //for loop breaks if num<0.0
    }
}

```

```
        sum=sum+num;
    }
    average=sum/(i-1);
    printf("Average=%.2f",average);
    return 0;
}
```

## Output

```
Maximum no. of inputs
4
Enter n1: 1.5
Enter n2: 12.5
Enter n3: 7.2
Enter n4: -1
Average=7.07
```

## 2. continue Statement

It is sometimes desirable to skip some statements inside the loop. In such cases, continue statements are used.

Syntax of continue Statement

```
continue;
```

Just like break, continue is also used with conditional if statement.

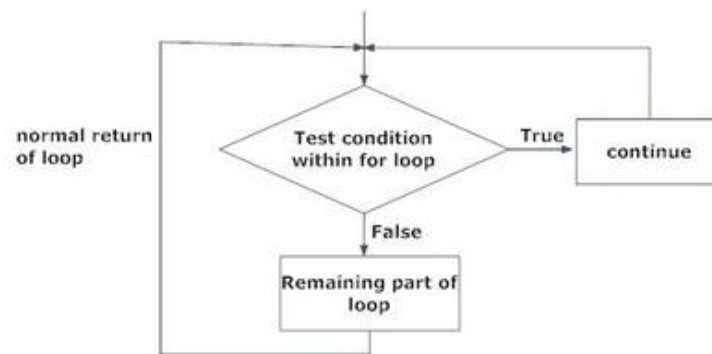
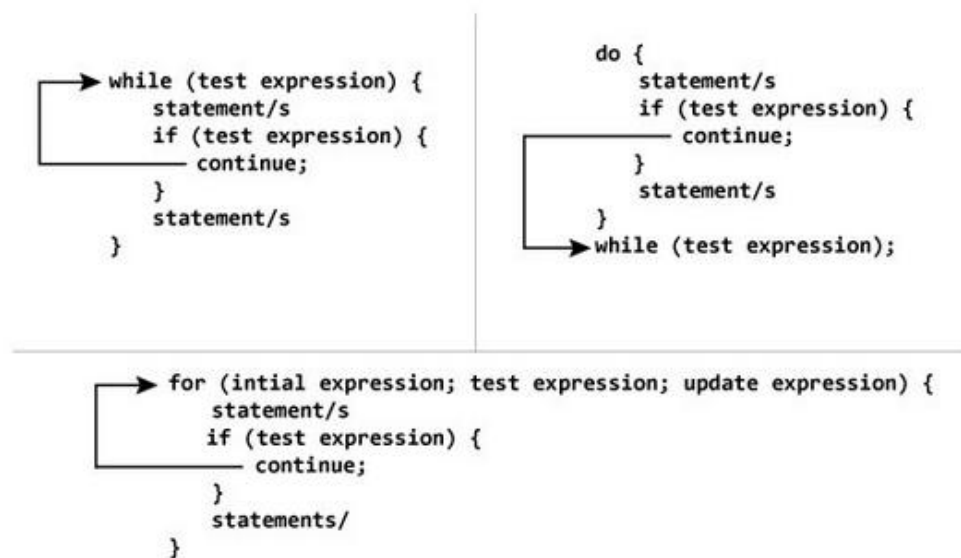


Fig: Flowchart of continue statement

For better understanding of how continue statements works in C programming. Analyze the figure below which bypasses some code/s inside loops using continue statement.



NOTE: The continue statment may also be used inside body of else statement.

Example of continue statement

Write a C program to find the product of 4 integers entered by a user. If user enters 0 skip it.

```
//program to demonstrate the working of continue statement in C programming
# include <stdio.h>
int main(){
    int i,num,product;
    for(i=1,product=1;i<=4;++i){
        printf("Enter num%d:",i);
        scanf("%d",&num);
        if(num==0)
            continue; / *In this program, when num equals to zero, it skips
the statement product*=num and continue the loop. */
        product*=num;
    }
    printf("product=%d",product);
    return 0;
}
```

Output

```
Enter num1:3
Enter num2:0
Enter num3:-5
Enter num4:2
product=-30
```

---

## HW Arrays

### EX1: Example of Multidimensional Array In C

Write a C program to find sum of two matrix of order 2\*2 using multidimensional arrays where, elements of matrix are entered by user.

Ouput

```
Enter the elements of 1st matrix
Enter a11: 2;
Enter a12: 0.5;
Enter a21: -1.1;
Enter a22: 2;
Enter the elements of 2nd matrix
```

```
Enter b11: 0.2;  
Enter b12: 0;  
Enter b21: 0.23;  
Enter b22: 23;
```

```
Sum Of Matrix:  
2.2    0.5  
-0.9   25.0
```

---

## EX2: C Program to Calculate Average Using Arrays

This program takes n number of element from user(where, n is specified by user), stores data in an array and calculates the average of those numbers.

Output

```
Enter the numbers of data: 6  
1. Enter number: 45.3  
2. Enter number: 67.5  
3. Enter number: -45.6  
4. Enter number: 20.34  
5. Enter number: 33  
6. Enter number: 45.6  
Average = 27.69
```

---

## EX3: C Program to Find Transpose of a Matrix

This program asks user to enter a matrix (size of matrix is specified by user) and this program finds the transpose of that matrix and displays it.

Output

Enter rows and column of matrix: 2  
3

Enter elements of matrix:

Enter elements a11: 1

Enter elements a12: 2

Enter elements a13: 9

Enter elements a21: 0

Enter elements a22: 4

Enter elements a23: 7

Entered Matrix:

1 2 9

0 4 7

Transpose of Matrix:

1 0

2 4

9 7

## EX4: C Program to Insert an element in an Array

Output of the Program :

Enter no of elements : 5

1 2 3 4 5

Enter the element to be inserted : 6

Enter the location : 2

1 6 2 3 4 5

---

## **EX5: C Program to Search an element in Array**

### **Output :**

Enter no of elements : 5

11 22 33 44 55

Enter the elements to be searched : 44

Number found at the location = 4

---

## **HW: Strings**

### **Ex1: C Program to Find the Frequency of Characters in a String**

This program asks user to enter a string and a character and this program checks how many times that character is repeated in the string entered by user.

Output



Enter a string: This website is awesome.

Enter a character to find frequency: e

Frequency of e = 4

---

## EX2: C Program to Find the Length of a String

You can use standard library function `strlen()` to find the length of a string but, this program computes the length of a string manually without using `strlen()` function.

Output

Enter a string: Programiz

Length of string: 9

---

## EX3: C Program to Reverse String Without Using Library Function

You can only use library function `strlen()`, To find the length of the string

**Output :**

Enter the string : Pritesh

Reverse string is : hsetirP