# C goto Statement

In this tutorial, you will learn to create goto statement in C programming. Also, you will learn when to use a goto statement and when not to use it.



The goto statement is used to alter the normal sequence of a C program.

### Syntax of goto statement

goto label;

... .. ...

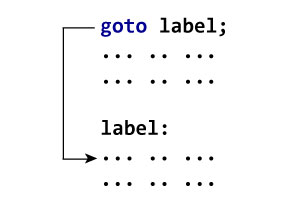
... .. ...

... .. ...

label:

statement;

The label is an identifier. When goto statement is encountered, control of the program jumps to label: and starts executing the code.



### Example: goto Statement

1. // Program to calculate the sum and average of maximum of 5 numbers
2. // If user enters negative number, the sum and average of previously entered positive numbers are displayed
3. # include <stdio.h>
4. int main()
5. {
6. const int maxInput = 5;
7. int i;
8. double number, average, sum=0.0;
10. for(i=1; i<=maxInput; ++i)
11. {
12. printf("%d. Enter a number: ", i);
13. scanf("%lf",&number);
14. // If user enters negative number, flow of program moves to label jump
15. if(number < 0.0)
16. goto jump;
17. sum += number; // sum = sum+number;
18. }
19. jump:
20. average=sum/(i-1);
21. printf("Sum = %.2f\n", sum);
22. printf("Average = %.2f", average);
23. return 0;
24. }

**Output**

1. Enter a number: 3

2. Enter a number: 4.3

3. Enter a number: 9.3

4. Enter a number: -2.9

Sum = 16.60

### Reasons to avoid goto statement

The use of goto statement may lead to code that is buggy and hard to follow. For example:

one:

for (i = 0; i < number; ++i)

{

test += i;

goto two;

}

two:

if (test > 5) {

goto three;

}

... .. ...

Also, goto statement allows you to do bad stuff such as jump out of scope.

That being said, goto statement can be useful sometimes. For example: to break from nested loops.

### Should you use goto statement?

If you think the use of goto statement simplifies your program. By all means use it. The goal here is to create code that your fellow programmers can understand easily.