Table of Contents

The commonly used control statements in C are:
If-else statement:
Switch statement:
Loops:
for loop:
while loop:
do-while loop:
Jump statements:
break statement:
continue statement:
goto statement:

In C programming, control statements are used to alter the flow of execution in a program based on certain conditions or to repeat a set of statements.

The commonly used control statements in C are:

If-Else Statement:

- The if-else statement is used to make a decision based on a condition.
- If the condition is true, the statements within the if block are executed; otherwise, the statements within the else block (optional) are executed.

C

```c
if (condition) {
    // code to execute if condition is true
} else {
    // code to execute if condition is false
}
```

Switch Statement:

- The switch statement is used to select one of many code blocks to be executed based on the value of an expression.
- It provides an alternative to a long sequence of if-else if-else statements.

C

```c
switch (expression) {
    case constant1:
        // code to execute if expression matches constant1
        break;
    case constant2:
        // code to execute if expression matches constant2
        break;
    // ...
    default:
        // code to execute if no case matches the expression
}
```

Loops:

Loops are used to repeat a set of statements until a specific condition is met.

The commonly used loops in C are:

for loop:

Executes a block of code repeatedly based on an initialization, condition, and increment/decrement.

C

```c
for (initialization; condition; increment/decrement) {
    // code to execute repeatedly
}
```

while loop:

Repeats a block of code as long as the condition is true.

C

```c
while (condition) {
    // code to execute repeatedly
}
```

do-while loop:

Executes a block of code first and then repeats it as long as the condition is true.

C

```
do {
    // code to execute repeatedly
} while (condition);
```

Jump Statements:

Jump statements are used to transfer control within a program to a different section of code.

The commonly used jump statements in C are:

break statement:

Terminates the current loop or switch statement and transfers control to the statement immediately following the loop or switch.

continue statement:

Skips the remaining code within the loop for the current iteration and proceeds to the next iteration.

goto statement:

Transfers control to a labeled statement within the same function. (Note: goto should be used with caution as it can make code harder to read and maintain.)

Related posts:

1. C prgoram to convert inch to feet

2. C program to convert KM to CM
3. C program to convert meter to centimeter
4. C program to calculate remainder, difference, division, product
5. C program to use printf() without semicolon ” ; “
6. C program to swap two numbers using 2 variables
7. C program to find nth term using Arithmetic progrssion
8. C program to find sum of first n even positive numbers
9. C program to calculate sum of first n even numbers
10. C program to find nth odd number
11. C program to find sum of first n odd positive numbers
12. C program to calculate perimeter and area of a rectangle
13. C program to calculate perimeter and area of a square
14. C program to calculate Perimeter and Area of Circle
15. Function in C Programming
16. C Programming Q & A
17. Main function in C Programming Q and A
18. Void main in C Programming
19. Variables Q and A in C Programming
20. Write a C Program to find the percentage of marks ?
21. Write a c program to find age of a person ?
22. Write a c program to get table of a number
23. What is Break statement in C Programming ?
24. Write a c program to generate all combinations of 1, 2 and 3 using for loop.
25. Write a C program to print all the prime numbers between 1 to 50.
26. Write a C program to get factorial of a number ?
27. What is user defined function in C programming ?
28. Difference between C and C++ Programming ?

29. Difference between C, C++ and Java Programming
30. C program addition of numbers using pointer
31. C Syntax
32. Comments in C
33. Variables in C
34. Data types in C
35. Format specifiers in C
36. Type Conversion in C
37. Constants in C
38. Operators in C
39. Pre and Post Increament Practice Problems
40. Pre and Post Increament
41. Array in C
42. C Introduction
43. C Get Started
44. C Pointers
45. C History
46. C Program Compiling and running
47. C While loop
48. C Do While Loop
49. C For loop
50. break and continue statement
51. C if-else ladder
52. C if statements
53. C 2-Dimensional array
54. C String library functions
55. C Functions

56. C Functions Categories
57. C Actual Arguments
58. Write a program that prints the message “Hello, World!”
59. Write a program that asks the user to enter two numbers, and then prints the sum of those two numbers.
60. Write a program that asks the user to enter a number and then determines whether the number is even or odd.
61. Write a program that swaps the values of two variables.
62. Write a program that asks the user to enter a number and then calculates and prints its factorial.
63. Write a program that asks the user to enter a number N and then prints the first N numbers in the Fibonacci sequence
64. Write a program that swaps the values of two variables without using a temporary variable
65. Converts a number into integer, float, and string
66. Program to find the length of the string
67. Program to convert string to uppercase or lowercase
68. Program to prints the numbers from 1 to 10.
69. What is identifier expected error
70. Difference between static and non static methods in Java
71. C String Input
72. C Character input
73. C Programming Variables MCQ
74. Object & Classes
75. C Programming find the output MCQs