- The for loop is a widely used loop structure in C that allows you to execute a block of code repeatedly for a specified number of times.
- It consists of three parts: initialization, condition, and increment/decrement.

Syntax of a for loop in C:

C

```c
for (initialization; condition; increment/decrement) {
    // code to be executed
}
```

Here’s a breakdown of each part:

- Initialization: It is an expression that initializes the loop control variable. It is executed only once at the beginning of the loop.
- Condition: It is a Boolean expression that is checked before each iteration. If the condition is true, the loop continues; otherwise, the loop is terminated.
- Increment/Decrement: It is an expression that updates the loop control variable after each iteration. It is executed at the end of each iteration.

Example:

C

```c
#include <stdio.h>

int main() {
    for (int i = 1; i <= 5; i++) {
        printf("Iteration %d\n", i);
```

```
    }

    return 0;
}
```

Explanation:

- In this example, the loop control variable i is initialized to 1.
- The loop continues as long as i is less than or equal to 5.
- After each iteration, the value of i is incremented by 1.
- The loop executes the code block inside it (printing the iteration number) for each iteration.

Output:

Output

```
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
```

Practice Problems In For Loop:

Problem 1: Print Numbers in a Range

Write a program that prints all the numbers from a given start value to an end value.

C

```c
#include <stdio.h>

int main() {
    int start, end;

    printf("Enter the start value: ");
    scanf("%d", &start);

    printf("Enter the end value: ");
    scanf("%d", &end);

    for (int i = start; i <= end; i++) {
        printf("%d ", i);
    }

    return 0;
}
```

Explanation:

- The program takes input from the user for the start and end values.
- The for loop is used to iterate from the start value to the end value.
- The loop control variable i is initialized to the start value, and the loop continues as

long as i is less than or equal to the end value.

- In each iteration, the value of i is printed.
- The loop iterates until i reaches the end value.

Output

```
Enter the start value: 5
Enter the end value: 9
5 6 7 8 9
```

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. break and continue statement
50. Control Statements in C
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