Ricky Downer

10/17/204

Assigemnt3

Explain the functionality of For and While loops with syntax.

\*Loops in Programming\*

Loops allow repeated execution of code blocks based on conditions.

\*For Loop\*

Used for iterating over arrays, vectors, or ranges.

\_Syntax\_

```

for (initialization; condition; increment/decrement) {

// code to execute

}

```

\_Components\_

1. Initialization: Executed once before the loop.

2. Condition: Evaluated before each iteration.

3. Increment/Decrement: Executed after each iteration.

\_Example\_

```

for (int i = 0; i < 5; i++) {

cout << i << endl;

}

```

\*While Loop\*

Used for conditional execution of code blocks.

\_Syntax\_

```

while (condition) {

// code to execute

}

```

\_Components\_

1. Condition: Evaluated before each iteration.

\_Example\_

```

int i = 0;

while (i < 5) {

cout << i << endl;

i++;

}

```

\*Do-While Loop\*

Similar to While Loop, but condition is evaluated after execution.

\_Syntax\_

```

do {

// code to execute

} while (condition);

```

\_Example\_

```

int i = 0;

do {

cout << i << endl;

i++;

} while (i < 5);

```

\*Key Differences\*

| Loop | Initialization | Condition | Increment |

| --- | --- | --- | --- |

| For | Yes | Yes | Yes |

| While | No | Yes | No |

| Do-While | No | Yes | No |

\*Best Practices\*

1. Use meaningful variable names.

2. Keep loop bodies concise.

3. Avoid infinite loops.

4. Use break and continue statements judiciously.

Would you like information on other control structures or loop types?

Write a C++ program to print numbers 1 to 10 using For loop.

Here's a simple C++ program using a for loop to print numbers from 1 to 10:

```

#include <iostream>

using namespace std;

int main() {

// For loop to print numbers from 1 to 10

for (int i = 1; i <= 10; i++) {

cout << i << endl;

}

return 0;

}

``)

\*\*Output:\*\*

```

1

2

3

4

5

6

7

8

9

10

```

\*\*Explanation:\*\*

1. Initialize the loop counter `i` to 1.

2. The condition `i <= 10` ensures the loop runs until `i` reaches 10.

3. `i++` increments `i` by 1 after each iteration.

4. `cout << i << endl;` prints the current value of `i` followed by a newline.

\*\*Compilation Steps:\*\*

1. Save the code in a file named `print\_numbers.cpp`.

2. Compile using `g++`: `g++ print\_numbers.cpp -o print\_numbers`.

3. Run the executable: `./print\_numbers`.

Would you like modifications or additional functionality?

```