

2. C++ Basics: Essential Differences from C

2.1 Basic Syntax Differences

C vs C++ Comparison:

Feature	C	C++
File Extension	<code>.c</code>	<code>.cpp</code>
Input/Output	<code>scanf()</code> , <code>printf()</code>	<code>cin</code> , <code>cout</code>
Header Files	<code>#include <stdio.h></code>	<code>#include <iostream></code>
Namespace	Not used	<code>using namespace std;</code>
Comments	<code>/* */</code> and <code>//</code>	<code>/* */</code> and <code>//</code> (preferred)
Boolean Type	<code>int</code> (0/1)	<code>bool</code> (true/false)
String Type	<code>char[]</code>	<code>string</code> class

2.2 Hello World Comparison

C Program:

```
#include <stdio.h>

int main() {
    printf("Hello, World!\n");
    return 0;
}
```

C++ Program:

```
#include <iostream>
using namespace std;

int main() {
```

```
cout << "Hello, World!" << endl;
return 0;
}
```

2.3 Input/Output in C++

Basic I/O Operations:

```
#include <iostream>
using namespace std;

int main() {
    int age;
    string name;
    float height;

    // Output (cout with insertion operator <<)
    cout << "Enter your name: ";

    // Input (cin with extraction operator >>)
    cin >> name;

    cout << "Enter your age: ";
    cin >> age;

    cout << "Enter your height (m): ";
    cin >> height;

    // Multiple outputs
    cout << "Name: " << name << endl;
    cout << "Age: " << age << " years" << endl;
    cout << "Height: " << height << " meters" << endl;

    return 0;
}
```

C vs C++ I/O Comparison:

Operation	C	C++
Output integer	<code>printf("%d", x);</code>	<code>cout << x;</code>

Operation	C	C++
Output float	<code>printf("%.2f", x);</code>	<code>cout << fixed << setprecision(2) << x;</code>
Output string	<code>printf("%s", str);</code>	<code>cout << str;</code>
Input integer	<code>scanf("%d", &x);</code>	<code>cin >> x;</code>
Input string	<code>scanf("%s", str);</code>	<code>cin >> str;</code>
Multiple inputs	<code>scanf("%d %d", &a, &b);</code>	<code>cin >> a >> b;</code>

Reading a Full Line:

```
#include <iostream>
#include <string>
using namespace std;

int main() {
    string fullName;

    cout << "Enter your full name: ";
    getline(cin, fullName); // Reads entire line including spaces

    cout << "Hello, " << fullName << "!" << endl;
    return 0;
}
```

Important Note on cin Buffer:

```
int age;
string name;

cin >> age; // User enters "25" and presses Enter
// Buffer now contains the newline character

cin.ignore(); // Clear the newline from buffer
getline(cin, name); // Now can read full line properly
```

Revision #5

Created 2025-09-10 01:51:03 UTC by YP

Updated 2025-11-09 13:24:11 UTC by DS