Introduction to C++

Language Features

- Object oriented
- Dynamic memory
- Structures
- Arrays
- Enumerations
- Inheritance
- Access Control (Scope)
- Free State Type (Abstract class)
- I/O
- Basic libraries (not used in C++)
- Standard functions
- Standard libraries
- Standard classes
- Standard containers
- Standard input
- Standard output
- Standard errors
- Standard namespace

Declarations / Member Functions

- Defines the way your program is written and how it is used by grouping commonly used functions into a single file.
- Frequently used includes files.

```c++
#include<iostream.h>
#include<conio.h>

void main()
{
    cout << "My very first program again!"
    getch();
}
```

The Pre-Processor and Directives

The Preprocessor is used to include the declarations specified by the compiler. The directives specify how files are interpreted. Typically, you include these files at the beginning of your code to ensure that the compiler reads them before the actual code.

```c++
#include "iostream"
```

Typically used to include functions which are extensively used in your code.

```c++
#include <iostream>
```
```c++
#include <iostream>

int main() {
    int num;
    cin >> num; // num % 2 computes the remainder when num is divided by 2
    if (num % 2 == 0) {
        cout << num << " is even ";
    }
    return 0;
}
```
void main() {
    int i, j, k;
    char c;
    float f;
    double d;

    printf("Enter the values for i, j, k, c, f, and d: ");
    scanf("%d %d %d %c %f %lf", &i, &j, &k, &c, &f, &d);

    printf("The values entered are: i = %d, j = %d, k = %d, c = '%c', f = %f, d = %lf\n", i, j, k, c, f, d);
}

// generate some random values for testing purposes
void generate_random() {
    int x, y, z;
    float x_f, y_f, z_f;

    x = rand() % 100;
    y = rand() % 100;
    z = rand() % 100;

    x_f = x / 10.0;
    y_f = y / 10.0;
    z_f = z / 10.0;

    printf("Random values generated: x = %d, y = %d, z = %d, x_f = %.2f, y_f = %.2f, z_f = %.2f\n", x, y, z, x_f, y_f, z_f);
}