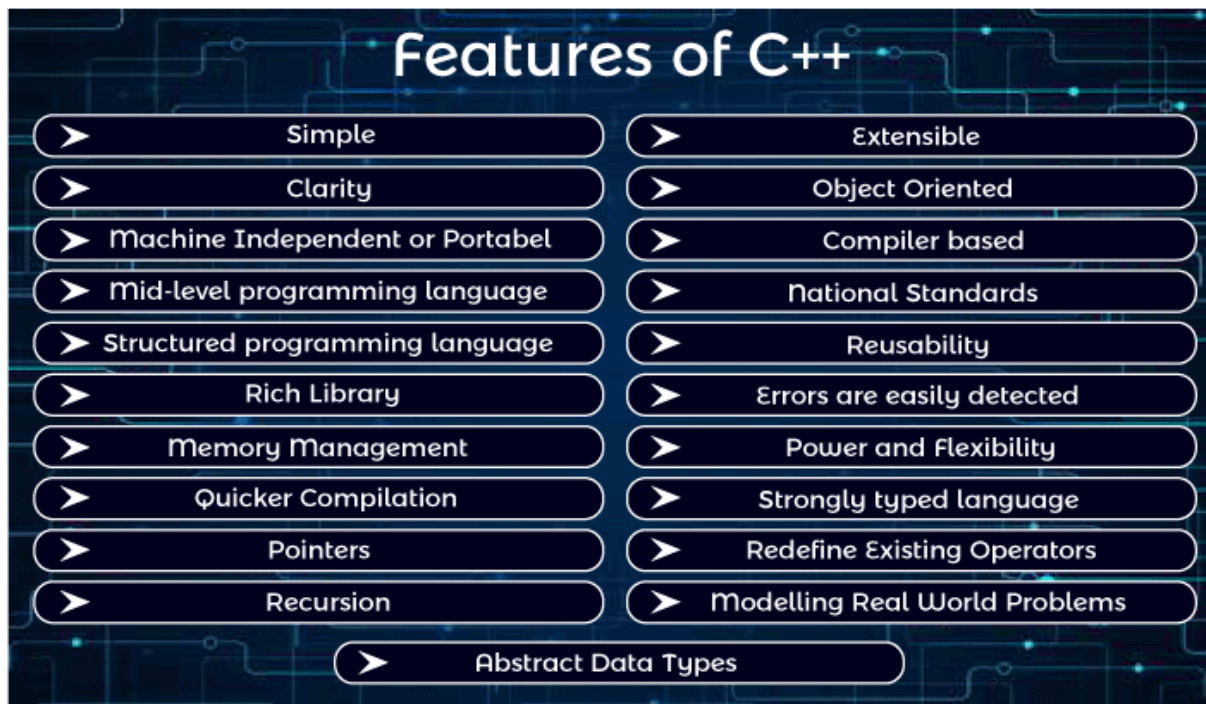# What is C++

C++ is a general purpose, case-sensitive, free-form programming language that supports object-oriented, procedural and generic programming.

C++ is a middle-level language, as it encapsulates both high and low level language features.

C++ Program

```
#include <iostream.h>
void main() {
  clrscr();
  cout << "Welcome to C++ Programming.";

}
```

#include<iostream.h> includes the **standard input output** library functions. It provides **cin** and **cout** methods for reading from input and writing to output respectively.

void main() The **main() function is the entry point of every program** in C++ language. The void keyword specifies that it returns no value.

cout << "Welcome to C++ Programming." is **used to print the data "Welcome to C++ Programming."** on the console.

C++ Variable

A variable is a name of memory location. It is used to store data. Its value can be changed and it can be reused many times.

It is a way to represent memory location through symbol so that it can be easily identified.

Let's see the syntax to declare a variable:

1. type variable_list;

Rules for defining variables

A variable can have alphabets, digits and underscore.

A variable name can start with alphabet and underscore only. It can't start with digit.

No white space is allowed within variable name.

A variable name must not be any reserved word or keyword e.g. char, float etc.

C++ Data Types

A data type specifies the type of data that a variable can store such as integer, floating, character etc.

Data Types in C++

Let's see the basic data types. It size is given according to 32 bit OS.

Data Types	Memory Size	Range
char	1 byte	-128 to 127
signed char	1 byte	-128 to 127
unsigned char	1 byte	0 to 127
short	2 byte	-32,768 to 32,767
signed short	2 byte	-32,768 to 32,767
unsigned short	2 byte	0 to 32,767
int	2 byte	-32,768 to 32,767
signed int	2 byte	-32,768 to 32,767
unsigned int	2 byte	0 to 32,767
short int	2 byte	-32,768 to 32,767

signed short int	2 byte	-32,768 to 32,767
unsigned short int	2 byte	0 to 32,767
long int	4 byte	
signed long int	4 byte	
unsigned long int	4 byte	
float	4 byte	
double	8 byte	
long double	10 byte	

C++ Keywords

A keyword is a reserved word. You cannot use it as a variable name, constant name etc.

C++ Operators

An operator is simply a symbol that is used to perform operations. There can be many types of operations like arithmetic, logical, bitwise etc.

There are following types of operators to perform different types of operations in C language.

- o Arithmetic Operators
- o Relational Operators
- o Logical Operators
- o Bitwise Operators
- o Assignment Operator
- o Unary operator
- o Ternary or Conditional Operator
- o Misc Operator

Operator	Type
+, -, *, /, %	Arithmetic Operators
<, <=, >, >=, ==, !=	Relational Operators
&&, ||, !	Logical Operators
&, |, <<, >>, ~, ^	Bitwise Operators
=, +=, -=,*=, /=, %=	Assignment Operators

Binary Operator

Unary Operator ⟶ ++, -- **Unary Operator**

Ternary Operator ⟶ ?: **Ternary or Conditional Operator**

C++ if-else

In C++ programming, if statement is used to test the condition. There are various types of if statements in C++.

- o if statement
- o if-else statement
- o nested if statement
- o if-else-if ladder

C++ switch

The C++ switch statement executes one statement from multiple conditions. It is like if-else-if ladder statement in C++.
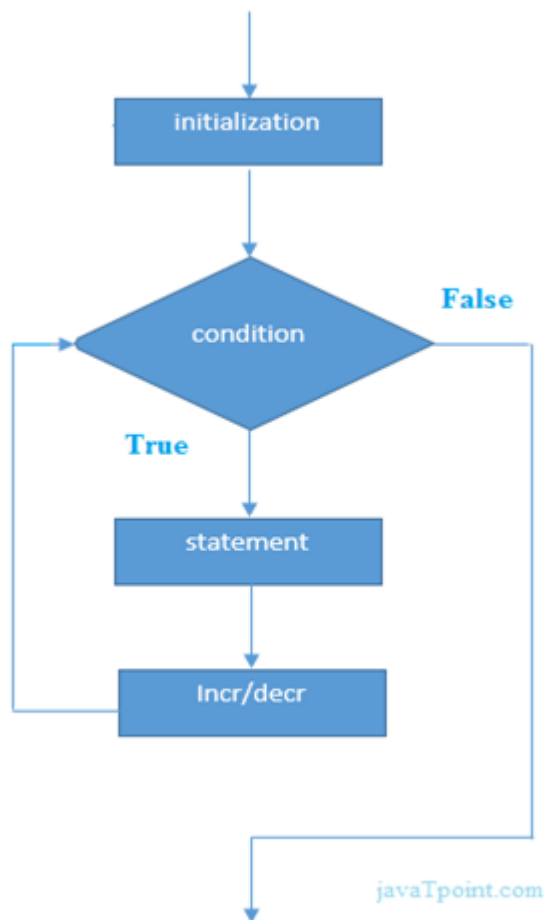
Fig: Switch Statement
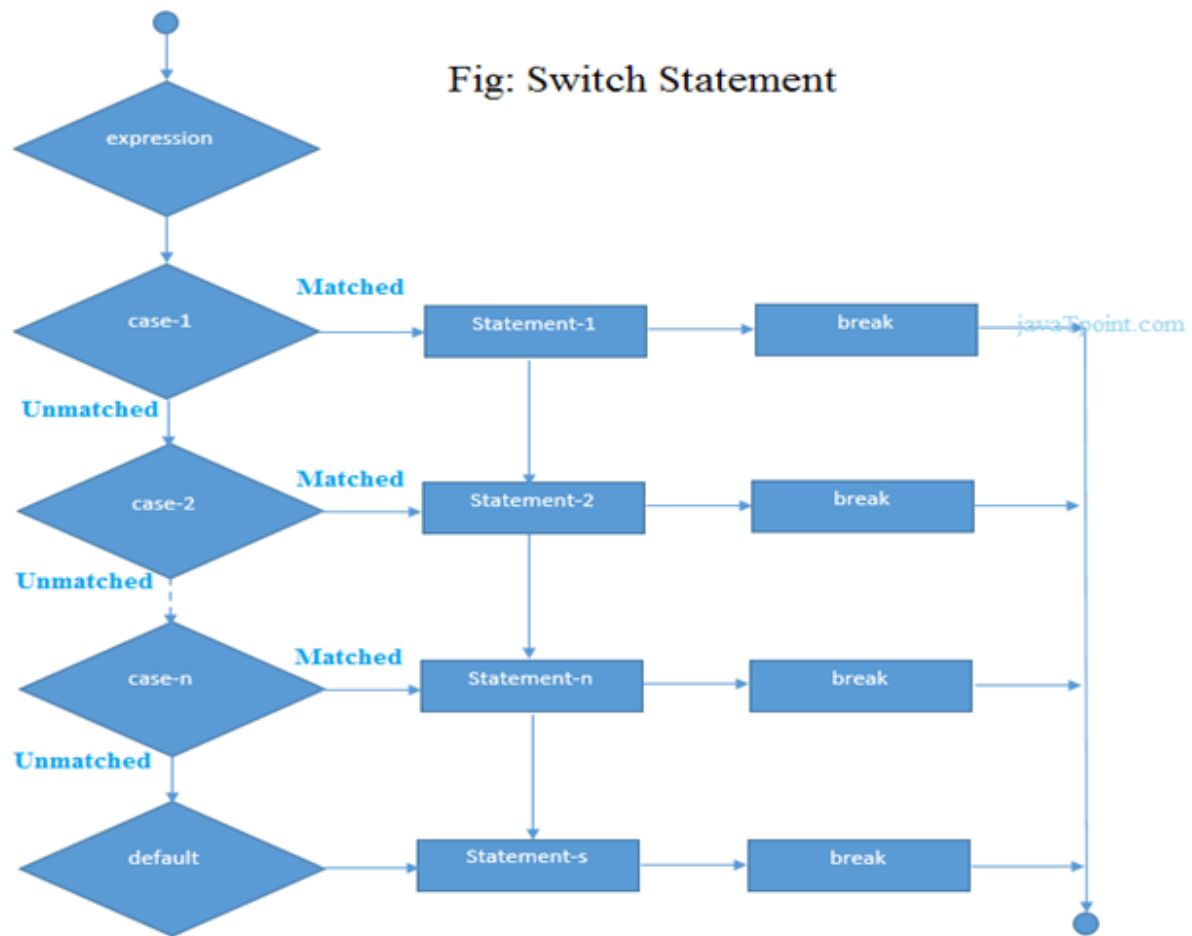
Loop

C++ For Loop

The C++ for loop is used to iterate a part of the program several times. If the number of iteration is fixed, it is recommended to use for loop than while or do-while loops.
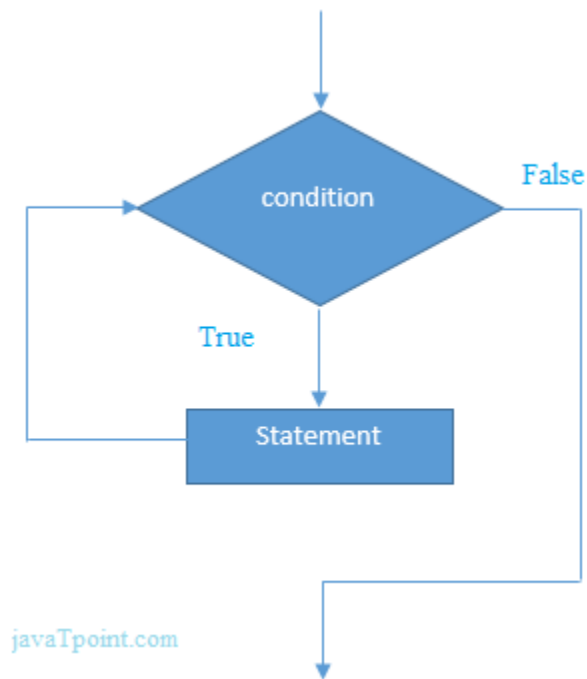
```
for(initialization; condition; incr/decr){
//code to be executed
}
```

C++ While loop

In C++, while loop is used to iterate a part of the program several times. If the number of iteration is not fixed, it is recommended to use while loop than for loop.

```
while(condition){
//code to be executed
}
```
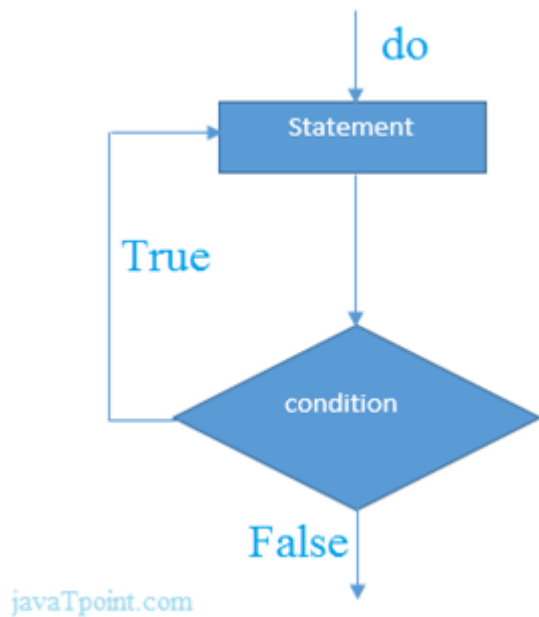
C++ Do-While Loop

The C++ do-while loop is used to iterate a part of the program several times. If the number of iteration is not fixed and you must have to execute the loop at least once, it is recommended to use do-while loop.

The C++ do-while loop is executed at least once because condition is checked after loop body.

```
do{
    //code to be executed
}while(condition);
```

C++ Break Statement

The C++ break is used to break loop or switch statement. It breaks the current flow of the program at the given condition. In case of inner loop, it breaks only inner loop.

1. jump-statement;
2. **break**;

C++ Continue Statement

The C++ continue statement is used to continue loop. It continues the current flow of the program and skips the remaining code at specified condition. In case of inner loop, it continues only inner loop.

1. jump-statement;
2. **continue**;