IEEE SCT SB

IEEE

Code in 10 days

Day 2

dotSlash
Workshop series

Topics for Today

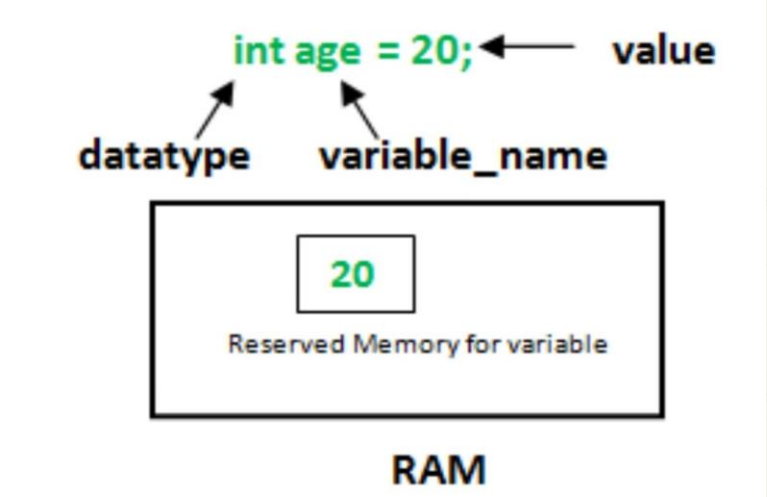
- Variables
- Operators
- Expressions
- Control statements

Variables

- Variables are containers for storing data values.

- Syntax

  Data type variable = value;

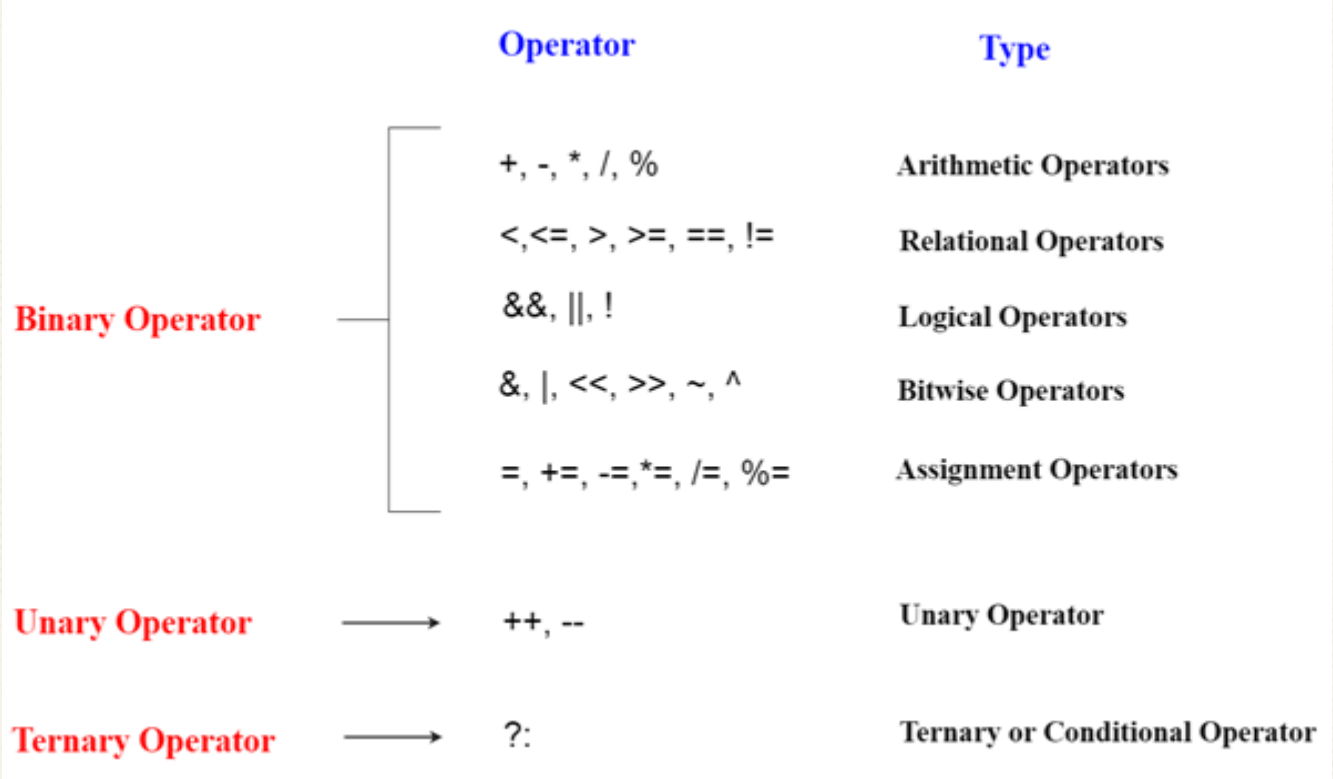


Variable Example

- int myNum = 5; // Integer (whole number without decimals)
- double myFloatNum = 5.99; // Floating point number (with decimals)
- char myLetter = 'D'; // Character
- string myText = "Hello"; // String (text)
- bool myBoolean = true; // Boolean (true or false)

Operators

- operator operates the operands.



Precedence of Operators

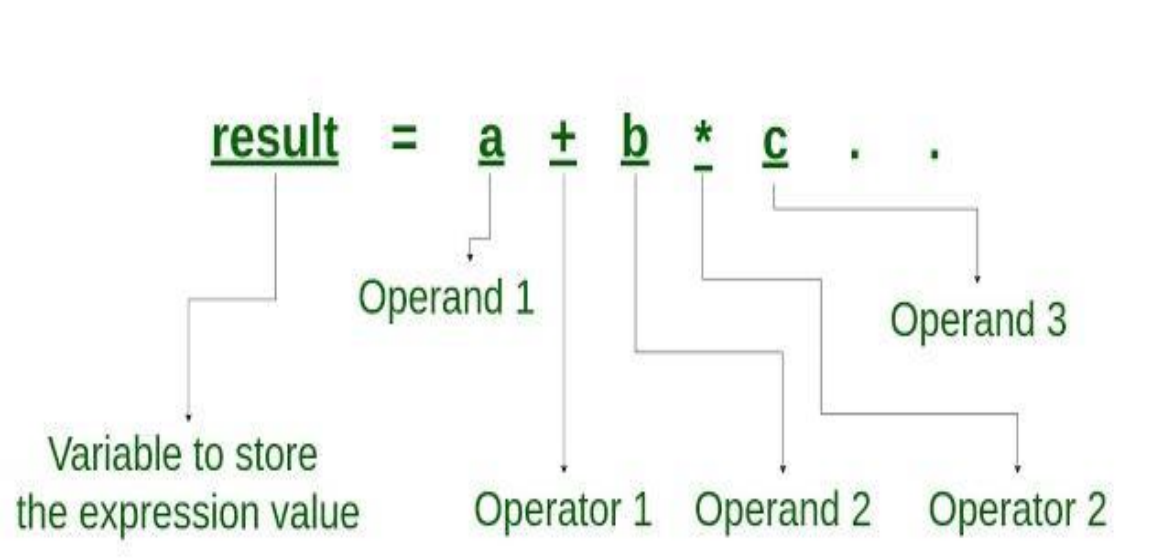
Category	Operator	Assosiativity
Postfix	++ -- () {} []	Left to right
Prefix and Unary	++ -- ! - +	Right to left
Multiplicative	* ÷ %	Left to right
Additive	+ -	Left to right
Relational	< <= > >=	Left to right
Equality	== !=	Left to right
Logical AND	&&	Left to right
Logical OR	||	Left to right
Assignment operator	=	Left to right
Comma operator	,	Left to right

Program 1

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

int main() {

  // evaluates 17 * 6 first
  int num1 = 5 - 17 * 6;

  // equivalent expression to num1
  int num2 = 5 - (17 * 6);

  // forcing compiler to evaluate 5 - 17 first
  int num3 = (5 - 17) * 6;

  cout << "num1 = " << num1 ;
  cout << "num2 = " << num2 ;
  cout << "num3 = " << num3 ;

  return 0;
}
```

Expressions

An expression is a combination of operators, constants and variables.

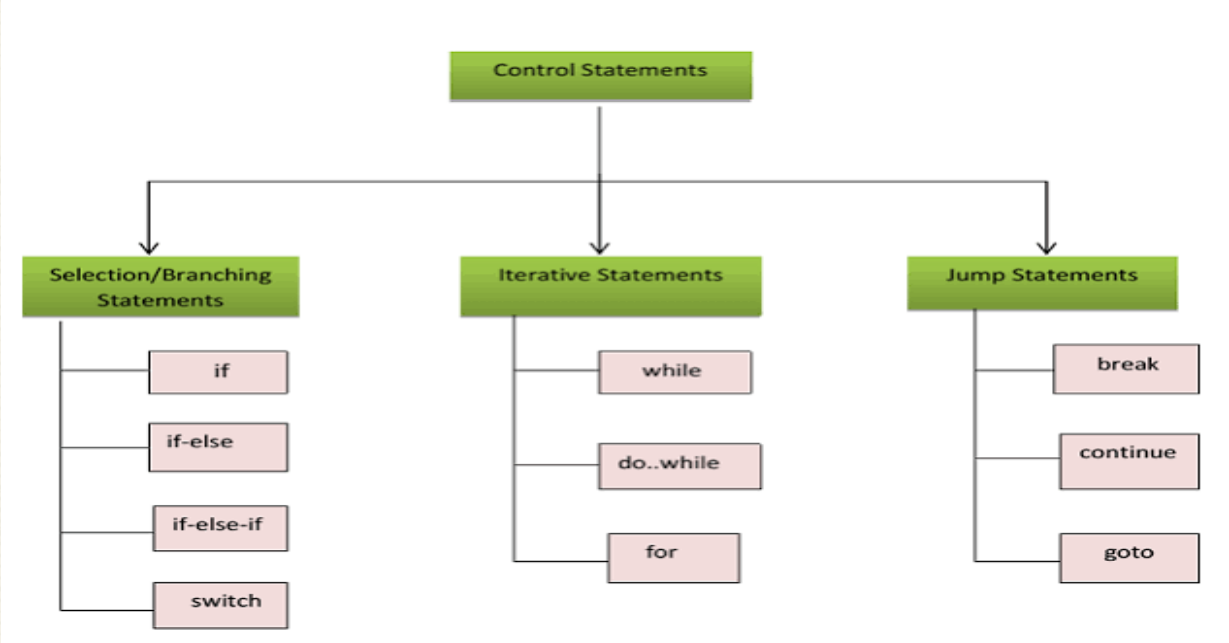


Types of Expressions

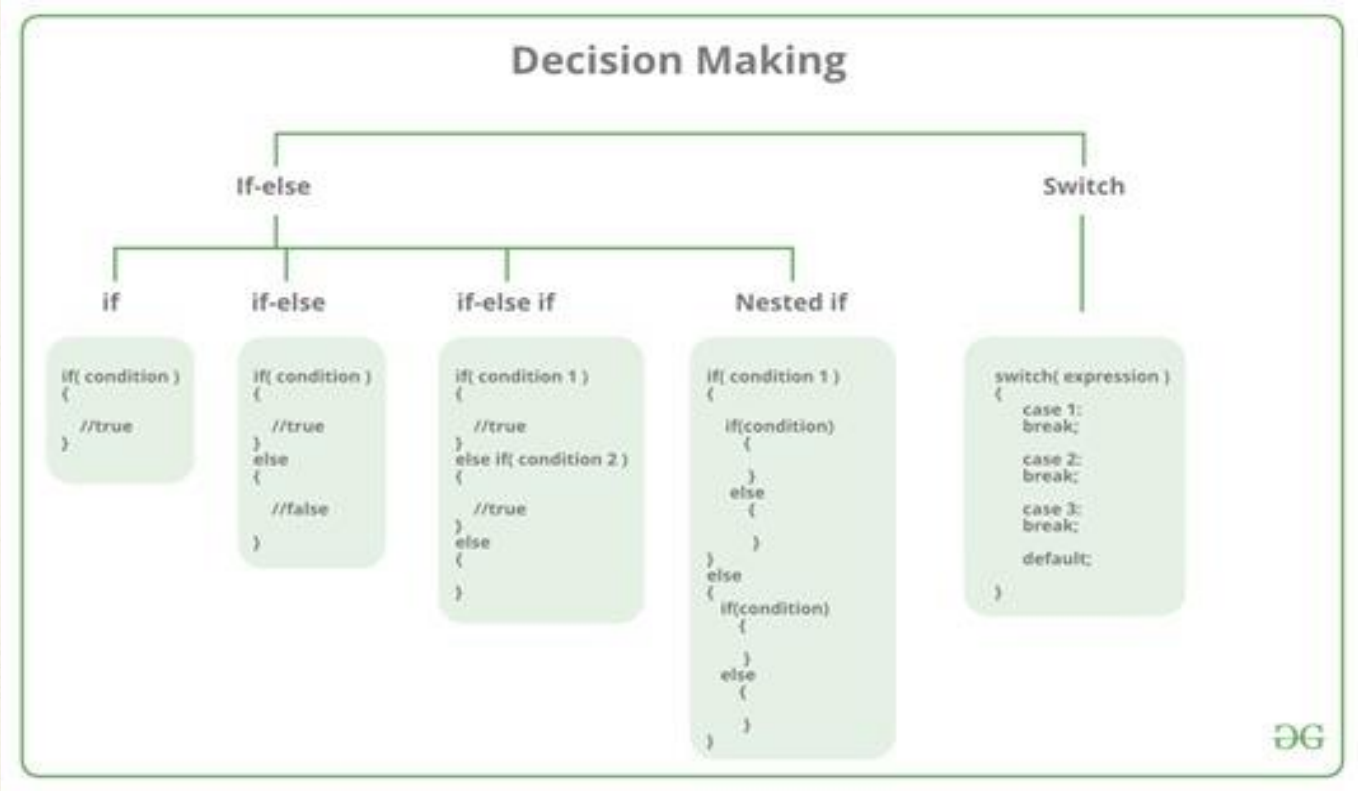
- Constant Expressions consists of only constant values.
  Eg- 10+5/6.0
- Integral Expressions are those which produce integer results.
  Eg- x+y, 6
- Float Expressions are which produce floating point results
  Eg- x+y,10.5
- Relational Expressions yield results of type bool which takes a value true or false.
  Eg- x <= y, x + y > 2
- Logical Expressions combine two or more relational expressions and produces bool type results.
  Eg- x > y && x == 10, x == 10 || y == 5

Control Statements

Decision making statements in programming languages decides the direction of flow of program execution.



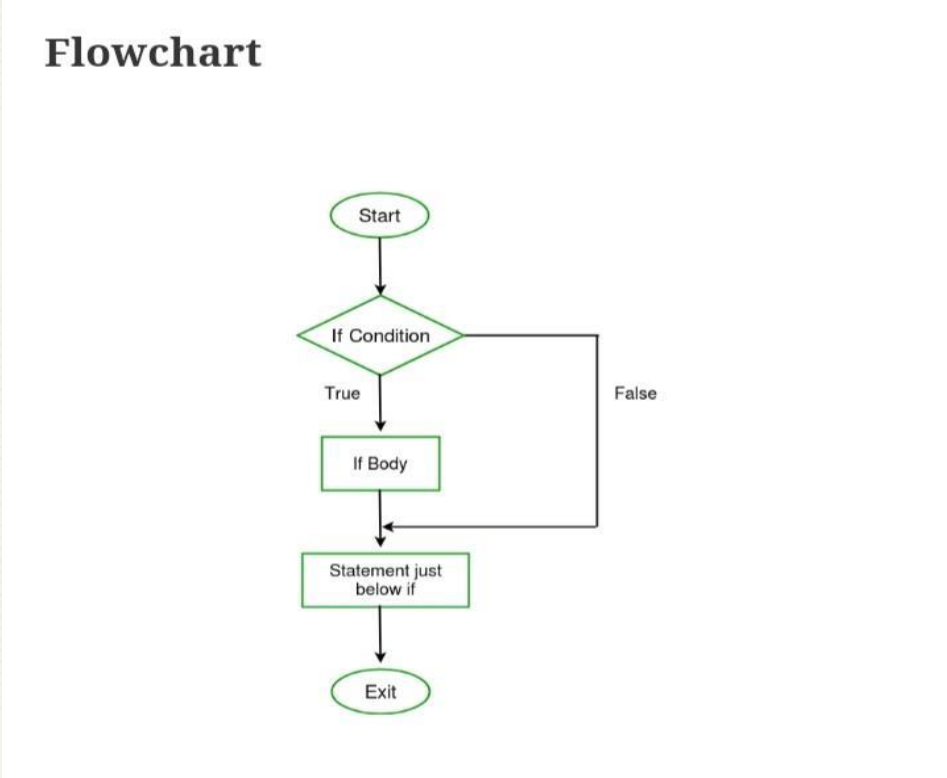
Conditional Statements



If statement

Syntax

```
if(condition)
{
   // Statements to execute if
   // condition is true
}
```



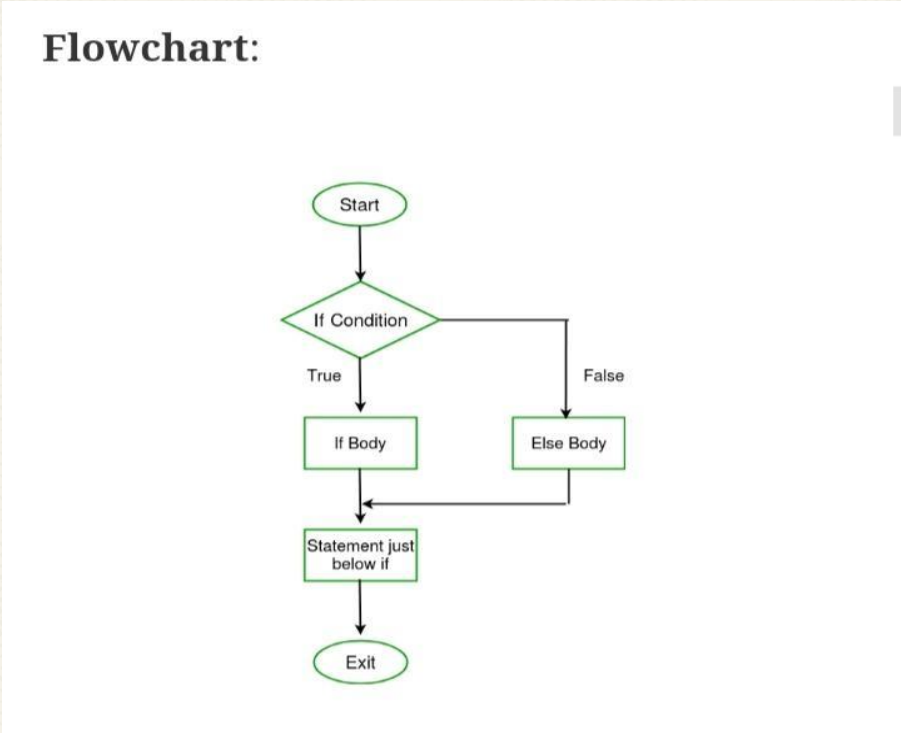
If example

```
//To display 'You have passed' if score is 18 or more

#include<iostream>
using namespace std;
int main()
{
int score ;
cout << "Enter your score: ";
cin >> score;
if (score >= 18)
cout << "You have passed";
return 0;
}
```

If-else statement

Syntax

```
if (condition)
{
   // Executes this block if
   // condition is true
}
else
{
   // Executes this block if
   // condition is false
}
```



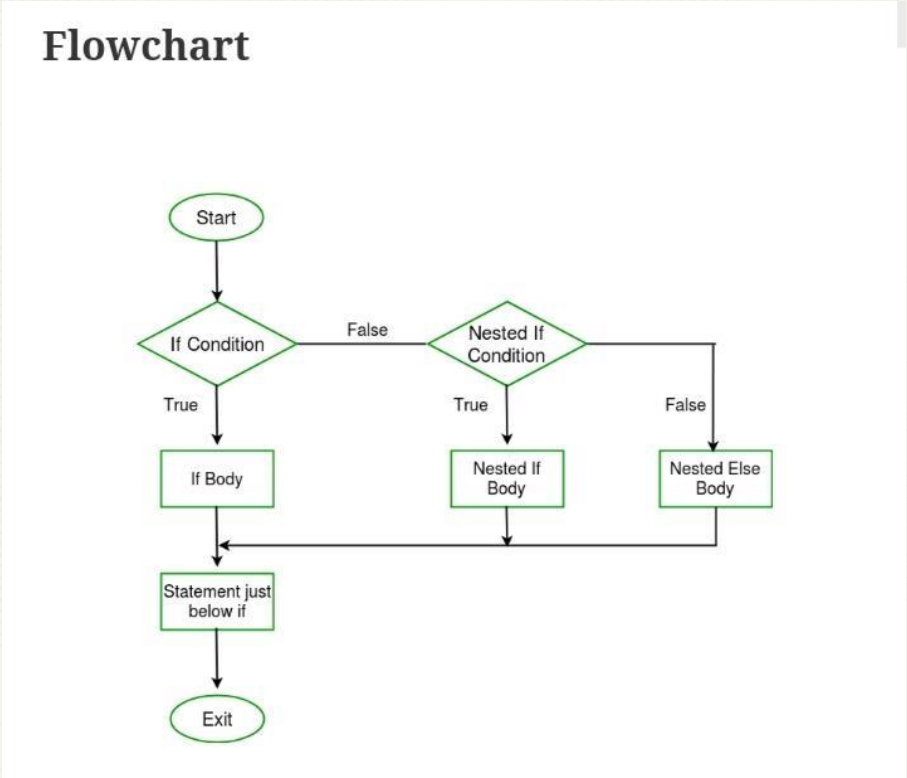
If-else example

```
//To check whether a given number is even or odd

#include <iostream>
using namespace std;
int main()
{
 int num;
cout << "Enter the number: ";
cin >> num;
if (num%2 == 0)
cout << "The given number is
Even";
else
cout << "The given number is
Odd";
return 0;
}
```

nested-if statement

Syntax

```
if (condition1)
{
  // Executes when condition1 is
true
  if (condition2)
  {
    // Executes when condition2 is
true
  }
}
```



nested-if example

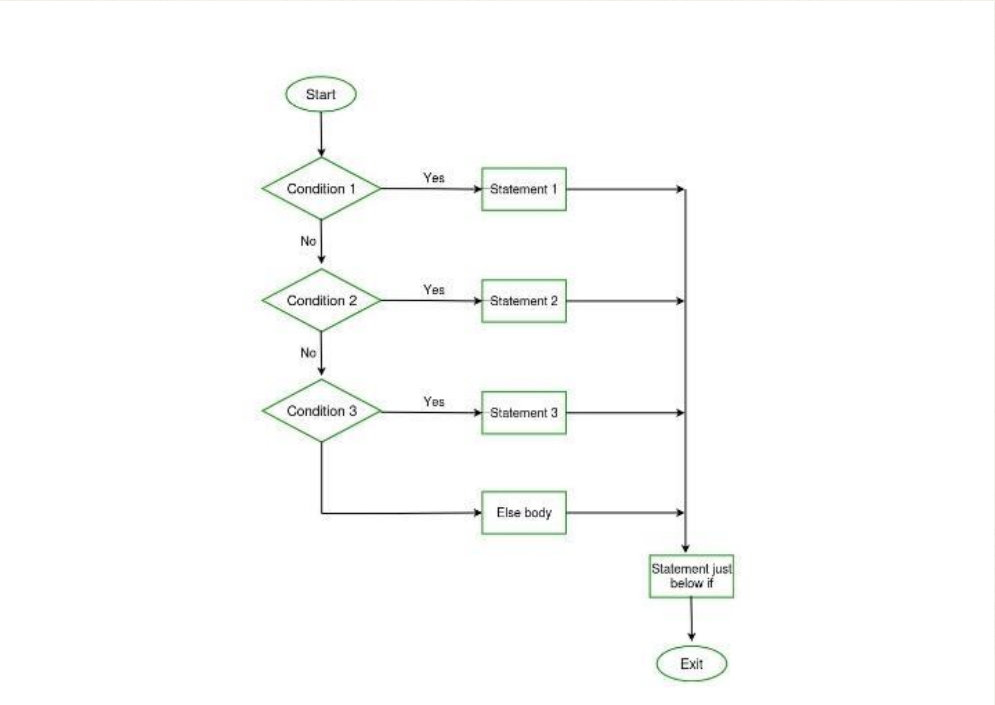
//To find the largest among three numbers

```
#include <iostream>
using namespace std;
int main()
{
int x, y, z;
cout << "Enter three different numbers: ";
cin >> x >> y >> z ;
if (x > y)
{
if (x > z)
cout << "The largest number is: " << x;
else
cout << "The largest number is: " << z;
}
else
 {
if (y > z)
cout << "The largest number is: " << y;
else
cout << "The largest number is: " << z;
}
return 0;
}
```

If-else-if ladder statement

Syntax

```
if (condition)
    statement;
else if (condition)
    statement;
.
.
else
    statement;
```



If-else-if ladder statement

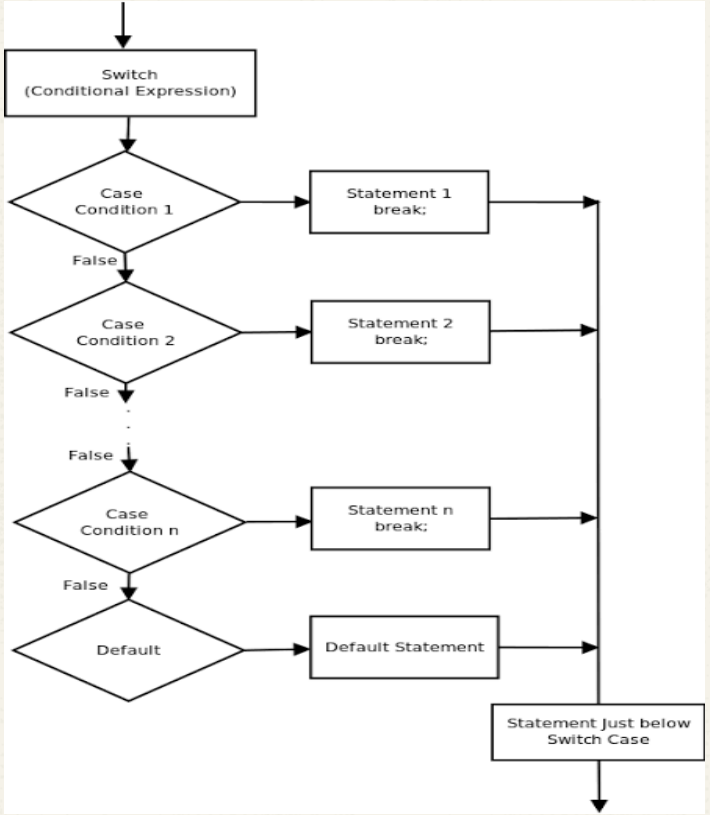
//To find the grade of a student for a given score

```
#include <iostream>
using namespace std;
int main()
{
int score;
cout << "Enter your score: ";
cin >> score;
if (score >= 80)
cout << "A Grade";
else if (score >= 60)
cout << "B Grade ";
else if (score >= 40)
cout << "C grade";
else if (score >= 30)
cout << "D grade";
else
cout << "E Grade";
return 0;
}
```

Switch statement

Syntax

```
switch(expression)
{
case constant_1 : statement block
1;
break;
case constant_2 : statement block
2;
break;
:
case constant_n-1 : statement
block n-1;
break;
default : statement block n;
}
```



Switch statement example

//To display the day of a week using switch statement

```
#include <iostream>
using namespace std;
int main()
{ int day ;
cout << "Enter a number between 1 and 7: ";
cin >> day ;
switch (day)
{
case 1: cout << "Sunday";
break;
case 2: cout << "Monday";
break;
case 3: cout << "Tuesday";
break;
case 4: cout << "Wednesday";
break;
case 5: cout << "Thursday";
break;
case 6: cout << "Friday";
break;
case 7: cout << "Saturday";
break;
default: cout << "Wrong input";
}
return 0;
}
```

Switch statement example

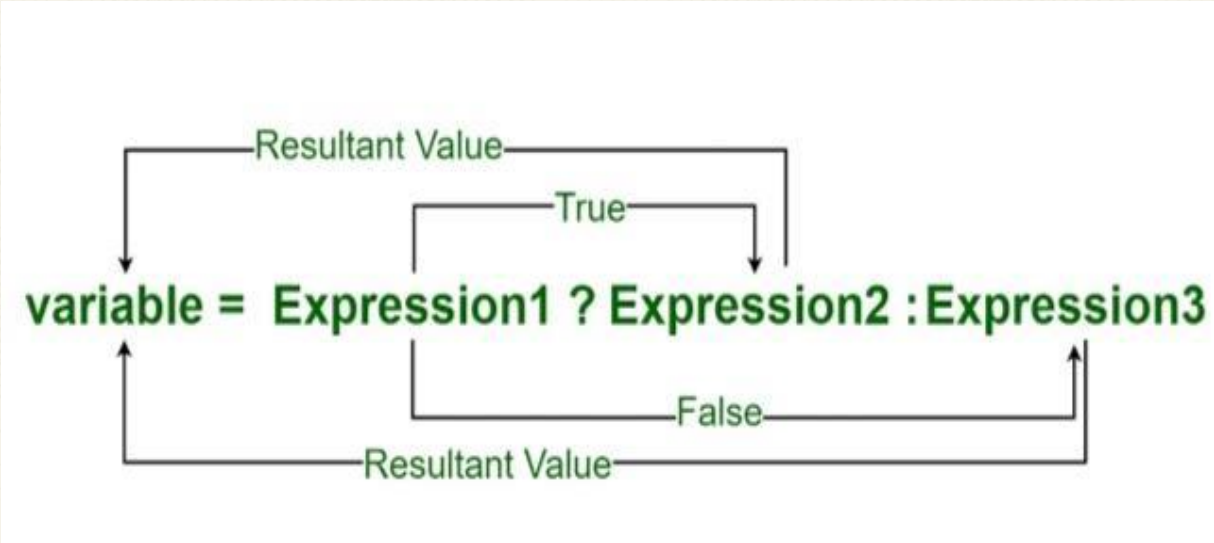
//To check whether the given character is a vowel or not

```
#include <iostream>
using namespace std;
int main()
{
char ch;
cout<<"Enter the character to check:
";
cin>>ch;
switch(ch)
{
case 'A' :
case 'a' :
case 'E' :
case 'e' :
case 'I' :
case 'i' :
case 'O' :
case 'o' :
case 'U' :
case 'u' : cout<<"The given character is
a vowel";
 break;
default : cout<<"The given character is
not a vowel";
}
return 0;
}
```

Conditional Operator

Syntax

variable = Expression1 ? Expression2 : Expression3



Conditional Operator example

```
//To find the larger number using the conditional operator

#include <iostream>
using namespace std;
int main()
{
int num1, num2;
cout << "Enter two numbers: ";
cin>> num1 >> num2 ;
(num1>num2)? cout<<num1<<"
is larger" : cout<<num2<<" is
larger";
return 0;
}
```

Program 2

//To check whether the given year is leap year or not

```
#include <iostream>
using namespace std;
void main()
{
int year ;
cout << "Enter the year (in 4-digits): ";
cin >> year;
if (year%100 == 0) // Checks for century year
{
if (year%400 == 0)
cout << "Leap year\n";
else
cout<< "Not a leap year\n";
}
else if (year%4 == 0)
cout << "Leap year\n";
else
cout<< "Not a leap year\n";
return 0;
}
```

Program 3

//To display the name of the day for a given day number

```
#include <iostream>
using namespace std;
int main()
{
int day;
cout << "Enter the day number (1-7): ";
cin >> day;
if (day == 1)
cout << "Sunday";
else if (day == 2)
cout << "Monday";
else if (day == 3)
cout << "Tuesday";
else if (day == 4)
cout << "Wednesday";
else if (day == 5)
cout << "Thursday";
else if (day == 6)
cout << "Friday";
else if (day == 7)
cout << "Saturday";
else
cout << "Wrong input";
return 0;
}
```

Thank You