Programming Fundamentals

By/ Merhan Atef

Section (4) Outline

1. **Practices**
2. **Relational and Logical Operators**
3. **Selection (Conditional) Statement**
   (If Statement)
4. **Conditional Operator (?:)**
5. **Go to**

Practice

Write a program that takes an integer input from the user and converts it to its binary representations.

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


int main() {
    int num;
    cout << "Enter an integer: ";
    cin >> number;
    cout << "Binary representation: " << bitset<32>(num) << endl;
    return 0;
}
```

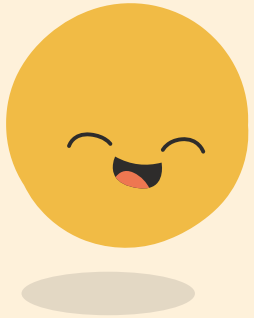
Practice

- Convert a given number of seconds to hours, minutes and seconds then print them in the format: 5:17:9.

```
#include<iostream>
using namespace std;
void main()
{
int seconds, minutes, hours;
cout << "Enter Seconds number ";
cin >> seconds;
hours = seconds / 3600;
minutes = seconds % 3600 / 60;
seconds = seconds % 3600 % 60;
cout << hours << ":" << minutes << ":" << seconds << endl;
system("Pause");
}
```

Operators

(Relational and Logical Operators)



Operators

Arithmetic

(- , + , * , / ,
% , -- , ++)

Relational

(> , < , >= ,
<= , == , !=)

Logical

(&& , || , !)

Relational and Logical Operators

Expressions that use relational or logical operators return 0 for false and 1 for true.

Relational Operators

Operator	Action
>	Greater than
>=	Greater than or equal
<	Less than
<=	Less than or equal
= =	Equal
!=	Not equal

Logical Operators

Operator	Action
&&	AND
| |	OR
!	NOT

Relational and Logical Operators

- Both the relational and logical operators are lower in precedence than the arithmetic operators.
- the relative precedence of the relational and logical operators:

Highest	!
	> >= < <=
	== !=
	&&
Lowest	||

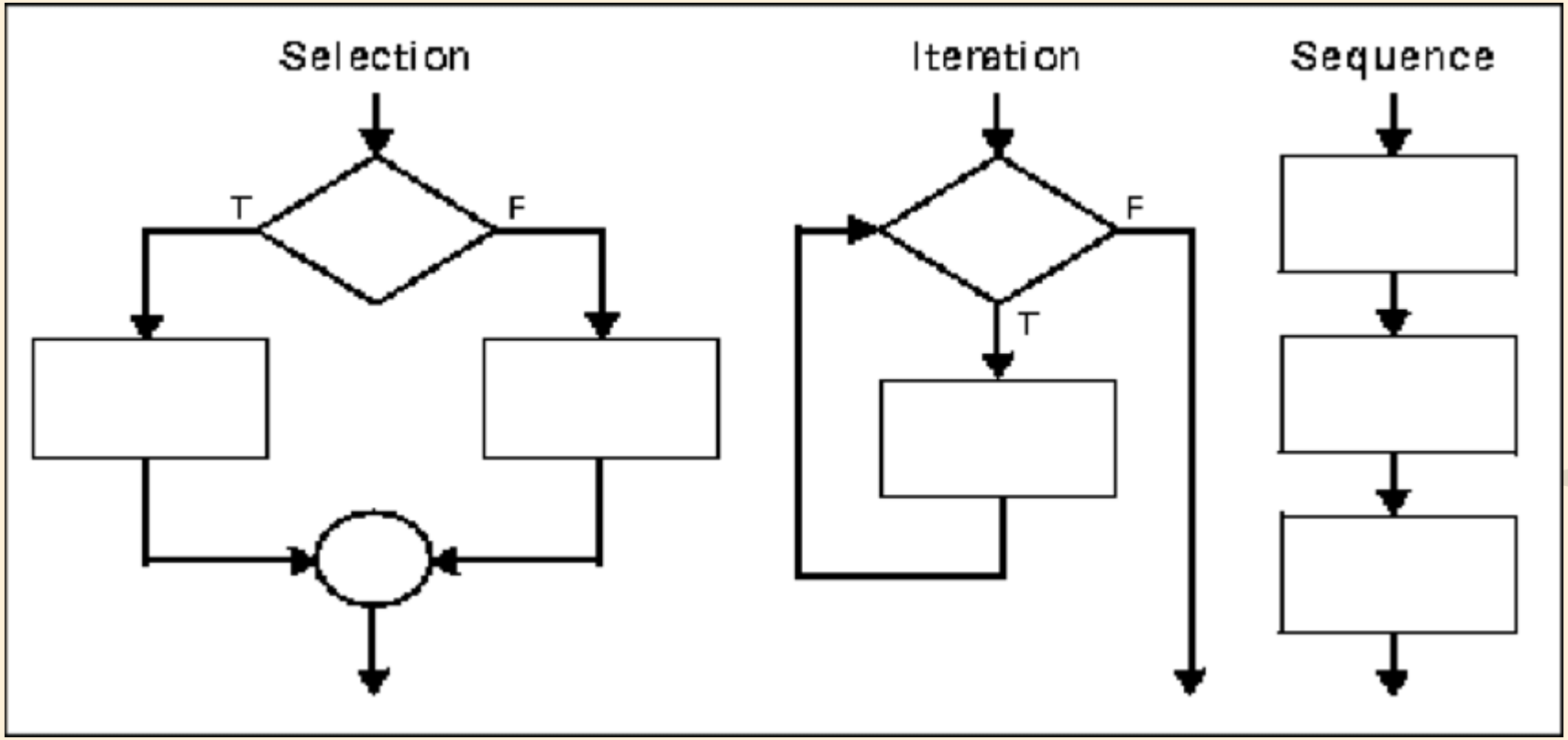
Example

10>5 && !(10<9) || 3<=4 ➡ True

Highest	( ) [ ] –> .
	! ~ ++ –– (type) * & sizeof
	* / %
	+ –
	<< >>
	< <= > >=
	== !=
	&
	^
	|
	&&
	| |
Highest	
	?:
	= += –= *= /= etc.
Lowest	,

Selection
(Conditional)
Statement

C++ Program



If Statement

- One way selection (If)
- Two-Way Selection (If …. else)
- Multiple Selections (Nested if)

One way selection

- General Form:

```
if (expression)
    statement
```

- The statement is executed if the value of the expression is true.

One way selection

```
#include<iostream>
using namespace std;
void main()
{
int number;
cout << "Enter number: ";
cin >> number;
if (number > 0)
cout << "number is Positive \n";
system("Pause");
}
```

Two-Way Selection

- General form:

```
if (expression)
     statement1
else
     statement2
```

- The statement1 is executed if the value of the expression is true. Otherwise, Statement2 is executed.

Two-Way Selection

```
#include<iostream>
using namespace std;
void main()
{
int number;
cout << "Enter number: ";
cin >> number;
if (number >= 0)
cout << "number is Positive \n";
else
cout << "number is negative \n";
system("Pause");
}
```

Multiple Selections

- General form:

```
If (expression1)
      Statement1
else if (expression2)
            Statement2
else if (expression3)
            Statement3
.
.
else
       Statement
```

Multiple Selections

```
void main()
{
int number;
cout << "Enter number: ";
cin >> number;
if (number > 0)
cout << "number is Positive \n";
else if (number==0)
cout << "number is zero \n";
else
cout << "number is negative \n";
system("Pause");
}
```

Block of Statement

```
void main()
{
int number;
cout << "Enter number: ";
cin >> number;
if (number > 0)
{
cout << "number is Positive \n";
number += 5;
cout << "number become : " << number << endl;
}
system("Pause");
}
```

Examples

- **Write a program that takes 3 integer from the user and prints the largest of these numbers.**

```
int x, y, z;
cout << "Enter 3 numbers \n";
cin >> x >> y >> z;
if (x >= y && x >= z)
cout <<"The largest number is:" <<x <<endl;
else if (y >= x && y >= z)
cout << "The largest number is:" << y << endl;
else
cout << "The largest number is:" << z << endl;
```

Examples

- Write a program that reads a number and prints if it is odd or even.

```
int x;
cout << "Enter your number";
cin >> x;
if (x % 2 == 0)
cout << "number is even \n";
else
cout << "number is odd \n";
system("Pause");
```

Conditional Operator (?:)

General form:

```
expression1 ? expression2 : expression3
```

If expression1 is true apply expression2 else apply expression3

```
int x;
cout << "Enter your number";
cin >> x;
if (x % 2 == 0)
cout << "number is even \n";
else
cout << "number is odd \n";
system("Pause");
```

```
int x;
cout << "Enter your number";
cin >> x;
(x % 2 == 0) ? cout << "even \n“ : cout <<
"odd \n";
system("Pause");
```

Go to

Go to

```
#include<iostream>
using namespace std;
void main()
{
int x, y;
Ahmed:
cout << "enter the first number: ";
cin >> x;
cout << "enter the second number: ";
cin >> y;

cout << y << " sum = " << x+y <<endl;
goto Ahmed;
system("Pause");
}
```

Go to

```
#include<iostream>
using namespace std;
void main()
{
int x, y;
a:
cout << "enter 2 numbers: ";
cin >> x >> y;
if (x > y)
cout << x << " is larger than " << y<<endl;
else
cout << y << " is larger than " << x<<endl;
goto a;
system("Pause");
}
```

Do you have any questions ?

Thank you

piece of advice

Aim for the impossible and don’t stop until you’ve made it possible