Chapter 3

Selections

Section 3.1-3.16

If Statements

- **الجمل الشرطية** هي من اهم الشغلات الي رح نحتاجها في البرمجة ، وعشان نفهم شو هي ال **if statement** خلينا نطرح **المثال التالي**:

- خلينا نتخيل انه عنا شخص اسمه محمد ، ومحمد بده يتغدا وقرر **التالي**:

--> **اذا** كان مطعم صاج فاتح رح اتغدى منه

--> **واذا** كان مسكر بدي اروح على الكافتيريا

--> **واذا** كانت مسكرة رح اروح على أي معطم ثاني

- **في المثال السابق ، لو** كان صاج فاتح ، فهو اكيد رح يروح على صاج ويكمل يومه طبيعي (بدون ما يروح على الكفتيريا او يشوف أي مطعم ثاني) ، لكن **لو** كان مسكر كان رح يشوف الشرط الي بعده وهكذا حتى ينفذ **شرط واحد على الأقل** بالنهاية

- نفس الاشي في ال **if statement** رح يكون في عنا شرط اذا تحقق رح ينفذ ما بداخله ويطلع يكمل الشغل طبيعي ، **واذا** الشرط ما تحقق رح يروح على الشرط الي بعده حتى ينفذ ما بداخلها ويكمل ما تبقى من البرنامج ، **واذا** انتهت جميع الشروط بضل عنا اخر اشرط وهي **else** وتعني اذا ولا شرط زبط فنفذ الي بداخلها

The bool Type and Operators

- زي ما فهمنا انه **الجمل الشرطية** بتحتاج شرط تتأكد منه واذا كان صحيح رح تنفذ ما بداخله ، لكن قبل ما نتعرف على الجمل الشرطية ، بدنا نتعرف على **data type** جديد الي رح يساعدنا بتكون الجمل الشرطية (**Boolean type**) والي بتميز انه بتكون من قيمتين فقط **true (1)** , **false (0)**

```
bool isTrue = true
```

```
bool isTrue = 1
```

```
bool isFalse = false;
```

```
bool isFalse = 0
```

```
cout << "isTrue = " << isTrue;
     << " isFalse = " << isFalse << endl;
```

isTrue = 1	isFalse = 0

- **ملاحظة :** اتفقنا ال **bool** ما عنده الا **true** او **false** لكن لو حطينا :

أي رقم رح يساوي = **true**, الا الصفر رح يساوي = **false** ، **مثال توضيحي:**

- bool b1 = -1.5; // true
- bool b2 = 0; // false
- bool b3 = 1.5; // true

---

the bool operators :

- حتى نكون جمل شرطية ، بنحتاج بعض العمليات الي رح تساعدني على تكوين هاي الجمل ، وبتتسمى ال **bool operators** :

Operator	Math symbol	Name	Example (i=5)	Result
<	<	اصغر من	i < 0	false
<=	≤	اصغر من او يساوي	i <= 0	false
>	>	اكبر من	i > 0	true
>=	≤	اكبر من او يساوي	i >= 0	true
==	=	يساوي	i == 0	false
!=	≠	لا يساوي	i != 0	true

Examples : find the value of op in the following code :

```
int x = 2 , y = 1;
bool op;

op = (y < x);
cout << op;        true

op = (y > x);
cout << op;        false

op = (y == y);
cout << op;        true
```

Boolean expression

```
op = (x <= y);
cout << op;        false

op = (x >= y);
cout << op;        true

op = (x != x);
cout << op;        false
```

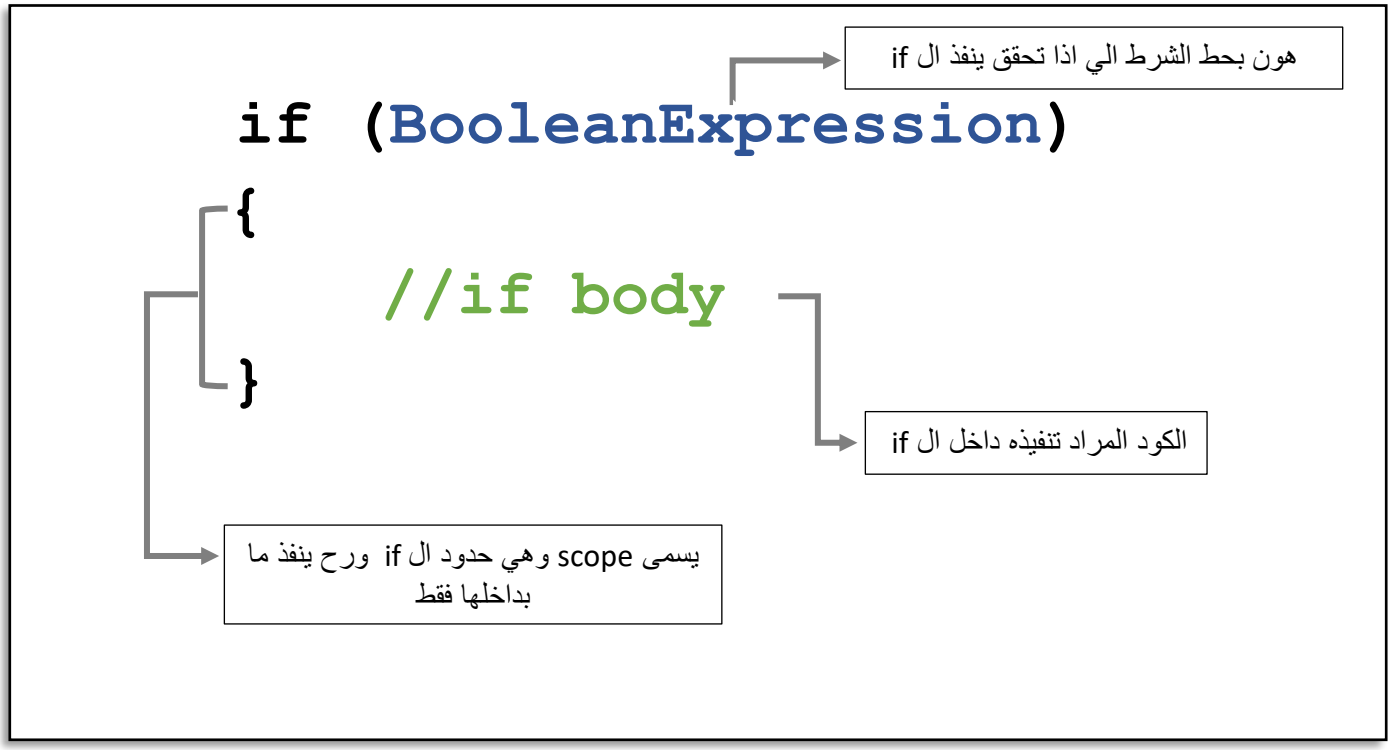
Boolean Expression: **bool operation** هي الجمل التي تحتوي على

كما في الأمثلة السابقة

- Type of if statement:

1) one way if-statement:

- وهي ابسط نوع من أنواع الجمل الشرطية وبتتكون من جملة شرطية واحدة



- **اذا** كان جواب ال **Boolean expression** :

true : سيتم الدخول الى if **وتنفيذ** ما بداخل ال **scope** ثم يكمل باقي الكود طبيعي

false : سيتم تجاهل ال if **ولن يتم** تنفيذ ما بداخل ال **scope** ثم يقفز لما بعد ال if ويكمل باقي الكود

true

```
if ( 1 < 2)
{
  cout << "true";
}
cout << " end" << endl;
```

Output : true end

نفذ ما بداخل ال **if** لأن الشرط = **true**

false

```
if (1 > 2)
{
    cout<< "false";
}
cout << " end" << endl;
```

Output : false end

تجاهل ما بداخل ال **if** لأن الشرط = **false**

Example : write a code that calculate the are of the circle , check the area is positive and nonzero

```
#include <iostream>
using namespace std;
int main()
{
    double radius,area;
    cin >> radius;

    if (radius > 0)
    {
        area = radius * radius * 3.14;
        cout << "The area for the circle of " <<
          " radius " << radius << " is " << area;
   }
}
```

الاعداد الموجبة هي الاعداد الأكبر من الصفر
ما رح يتم تنفيذ ما بداخل ال if الا اذا كانت الـradius اكبر من صفر ، يعني موجبة

Output:

```
radius = 1
```

The area for the circle of radius 1 is 3.14

```
radius = -1
```

ما طبع اشي لأنه الـradius سالبة ، فجواب ال Boolean expression رح يعطي false ورح يتخطى ال if

- بعض الحالات بكون في اكثر من if وحدة بالكود :

```
if (BooleanExpression1)
{
   // first if body
}

if (BooleanExpression2)
{
   //second if body
}
.
.
// باقي البرنامج
```

false

true

بعد ما ينتهي بروح على ال if الي بعدها

true

بعد ما ينتهي اذا كان في if بعدها بروح عليها، او بكمل باقي البرنامج

- **true : بنفذ** ال if بعدين بعمل check على ال if الي بعدها

- **False : ما بنفذ** ال if الي هون عندها وبنتقل على الي بعدها وبعمل check عليها

true

Check

false

```
if ( 1== 1)
    cout << "1" << endl;
if ( 1>= 2)
    cout << "2" << endl;
```

Output : 1

false

Check

true

```
if (1 < 2)
    cout << "1" << endl;
if (1 < 2)
    cout << "2" << endl;
```

Output : 2

Example: write a program that prompts the user to enter an integer. If the number is a multiple of 5, displays HiFive. If the number is even, displays HiEven.

```
#include <iostream>
using namespace std;
int main()
{
    int number;
    cout << "Enter a number: ";
    cin >> number;

    // if number multiple of 5
    if (number % 5 == 0)
        cout << "HiFive" << endl;

    // if number multiple of 2
    if (number % 2 == 0)
        cout << "HiEven" << endl;
 }
```

من مضاعفات ال 5 يعني انه باقي قسمته على 5 تساوي 0

من مضاعفات ال 2 يعني انه باقي قسمته على 2 تساوي 0

Examples:

number = 15

```
Enter a number: 15
HiFive
```

number = 1

```
Enter a number: 1
-----
```

لأن ال 1 لم يحقق أي من الشرطين

number = 4

```
Enter a number: 4
HiEven
```

number = 10

```
Enter a number: 10
HiFive
HiEven
```

هون تحقق **اول** شرط وهو ال 10 باقي قسمتها على 5 تساوي 0

وتحقق **ثاني** شرط وهو ال 10 باقي قسمتها على 2 تساوي 0

ملاحظات مهمة :

- ال **boolean-expression** لازم يكون محاط ب ()

```
if i > 0
{
  cout << "i is positive" << endl;
}
```

(a) Wrong

```
if (i > 0)
{
  cout << "i is positive" << endl;
}
```

(b) Correct

- **ممكن الاستغناء** عن ال { } في حال كان بعد ال if **فقط** جملة واحدة ، لكن اذا كان اكثر من جملة لازم نستخدم ال { }

```
if (i > 0)
{
  cout << "i is positive" << endl;
}
```

(a)

Equivalent

```
if (i > 0)
  cout << "i is positive" << endl;
```

(b)

- تتم **مقارنة** الحروف حسب ترتيبها الابجدي :

* 'A' < 'B' < 'C' < 'D' …

* 'a' < 'b' < 'c' < 'd' …

* Small letter > Capital letter

ترتيب الحروف الهجائية متوافق مع ترتيب قيمتها

- يجب التفريق بين عملية التخزين (=) وعملية المقارنة (==) :

==
Comparison
int x = 3 ;
(x == 1)
true
(x == 3)
false

=
assignment
int x = 3;
(x = 3)
(x = 1)

Always true

تمت عملية تخزين قيمة بال x وبما انه القيمة مش 0 فالجواب true

2) Two way if-else statement:

- وهي ابسط نوع من أنواع الجمل الشريطية وبتتكون من جملة شرطية واحدة

هون بحط الشرط الي اذا تحقق ينفذ ال if

```
if (BooleanExpression)
{
  //if body
}
else
{
  //else body
}
```

true

false

لاحظ انه مافي شرط لل else لانها بتتنفذ اذا ما تم تنفيذ ال if

اذا كان جواب ال Boolean expression :

true : سيتم الدخول الى if وتنفيذ ما بداخل ال scope {} ، وما رح يفوت على الـ else ثم يكمل باقي الكود طبيعي

false : سيتم تجاهل ال if ولن يتم تنفيذ ما بداخل ال scope {} ثم يقفز الى الـ else وبنفذ الي داخلها ثم باقي الكود

---

true

```
if (1 < 2)
  cout << "true";

else
  cout << "else";
```

Output : true

اذا تم تنفيذ ال if فما بتم تنفيذ ال else

false

```
if (1 > 2)
 cout << "false";

else
  cout << "else";
```

Output : else

اذا لم يتم تنفيذ ال if بتم تنفيذ ال else

Example : write a code that calculate the are of the circle , if radius is negative print (negative radius).

```
#include <iostream>
using namespace std;
int main()
{
    int radius;
    cin >> radius;
    double area;

    if (radius >= 0)
    {
        area = radius * radius * 3.14;
        cout << "The area for the circle of " <<
            " radius " << radius << " is " << area;
    }
    else
        cout << "Negative radius";
}
```

اذا ما تم تنفيذ ال if معناته الرقم مش اكبر من 0 يعني الرقم سالب

Example: write a program that prompts the user to enter an integer and define if the number is even or odd :

```
#include <iostream>
using namespace std;
int main()
{
    int number;
    cout << "Enter number: ";
    cin >> number;

    if (number % 2 == 0)
        cout << "The number " << number << " is even";
    else
        cout << "The number " << number << " is odd";

}
```

أي رقم باقي قسمته على 2 يساوي 0 فهو زوجي

3) else if statemenet

- لو كان عندي شرط اول، اذا ما تحقق فعندي شرط اخر، هون الشرط الأول والثاني بعتمدوا على بعض (انه اذا الأول ما تحقق اعمل check على الثاني)

- بمثال الطالب محمد الي بده يتغدى :

--> الشرط الأول : اذا صاج فاتح رح يتغدى منه

--> الشرط الثاني : اذا صاج مش فاتح ، رح يشوف اذا الكافتيريا فاتحه ويتغدى منها

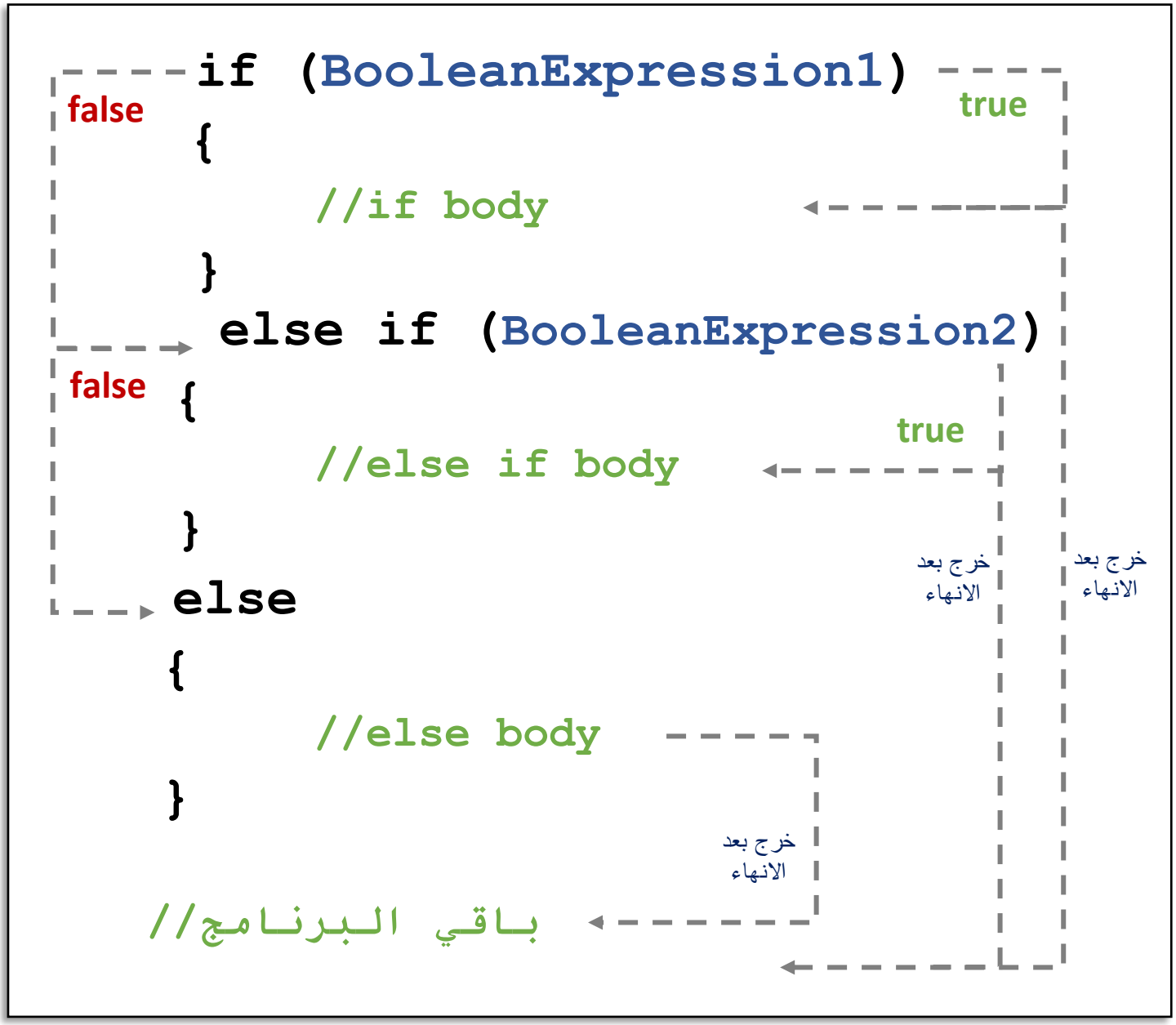
--> اذا ولا شرط تحقق : رح يروح على أي مطعم يتغدى منه

- لو نلاحظ، انه الشروط معتمدة على بعض ، عدم تحقق الأول بنقلني على الثاني else if ولو كان ولو ما تحقق بنقلني على الشرط الي بعده وهكذا

واذا ولا واحد منهم تحقق بنتقل على الحالة الأخيرة else

- وفي حالة احد الشروط تحققت بنفذها ومافي داعي يشوف الي بعدها

else if statemenet



اذا كان جواب الـ Boolean expression :

true : سيتم الدخول الى if وتنفيذ ما بداخل ال scope {} ، وما رح يفوت على أي اشي بعدها ثم يكمل باقي الكود طبيعي

false : سيتم تجاهل ال if ولن يتم تنفيذ ما بداخل ال scope {} ثم بنتقل الى else If الي بعدها واذا كانت false بروح على الي بعدها ، حتى وصوله لل else وبنفذ الي داخلها ثم باقي الكود

Example:what is the output with following code:

```
int x, y;
if (x > y)
    cout << "1" << endl;
else if (x == y)
    cout << "2" << endl;
else
    cout << "3" << endl;
```

input: x = 2 , y = 1	input: x = 1 , y = 1	input: x = 0 , y = 1
Output : 1	**Output :** 2	**Output :** 3
تم تنفيذ ال if الأولى	لم يتم تنفيذ الـ **if** ولكن تم تنفيذ الـ **else if**	لم يتم تنفيذ الـ **if** ولا الـ **else if** وتم تنفيذ الـ **else**

Example: Write a program to choose a student's grade based on his score

```
if (score >= 90)
     cout << "Grade is A";
else if (score >= 80)
     cout << "Grade is B";
else if (score >= 70)
     cout << "Grade is C";
else if (score >= 60)
     cout << "Grade is D";
else
     cout << "Grade is F";
```

input: 101	input: 88	input: 70
Output : A	**Output :** B	**Output :** C
تم تنفيذ ال if الأولى	تم تنفيذ ال else if الأولى	تم تنفيذ ال else if الثانية

input: 66	input: 0	input: -5
Output : D	**Output :** F	**Output :** F
تم تنفيذ ال else if الثالثة	تم تنفيذ ال else	تم تنفيذ ال else

4) Nested if statements :

لو رجعنا لمثال الطالب محمد ، طلع من المحاضرة بده يروح على صاج

لو كان صاج فاتح رح يروح يفوت عليه :

<-- لو كان عنده زنجر رح يتشري منه

<-- اذا لا رح يطلع من المحل ويشوف محل ثاني .

- ولاحظنا من المثال السابق انه عنا شرط جوا الشرط وهذا ما يسمى بالـ nested if

```
if (BooleanExpression1)
{
    if (BooleanExpression2)
    {
        //if body
    }
}
//باقي البرنامج
```

false / true / false / true

booleanExpresstion1 :

true : رح يدخل على الـ **if** الأولى ، ويعمل **check** على الـ **if** الثانية

false : ما رح يدخل على الـ **if** الأولى ، وما رح **يوصل** لل **if** الثانية ابدا لأنها **بداخل** الاولى

```
int i, j, k;

if (i > k) {
   if (j > k)
     cout << "i and j are greater than k" << endl;
}
else
   cout << "i less than or equal k" << endl;
```

input: i = 2, j = 3, k = 1	input: i = 2, j = 0, k = 1	input: i = 0, j = 2, k = 1
Output : i and j are greater than k	**Output** : ----------	**Output** : i less than or equal k
تم تنفيذ ال if الأولى	تم تنفيذ ال if الأولى ولم يتم تنفيذ الثانية	تم تنفيذ ال else فقط

Logical Operators:

- بعض الرموز الي بتعمل على تجميع ال boolean expression في جملة شرطية واحدة

Operator	Name	Description	example
!	not	بتعكس من true الى false والعكس	! (true) = false ! (false) = true
&&	and	لازم يتحقق الشرطين مع بعض حتى يكون الجواب true	true && true = true true && false = false false && true = false false && false = false
||	or	من الكافي ان يتحقق شرط واحد فقط حتى يكون الجواب true	true || true = true true || false = true false || true = true false || false = false

- and (&&) : لازم **يتحقق الشرطين** حتى يتم تنفيذ ال if statement
- or (||) : بكفي **تحقق شرط واحد** حتى يتم تنفيذ ال if statement
- not(!) : **بتعكس** جواب الشرط ، اذا كان true بصير false والعكس

Examples: find the value of the following boolean expression.

```
bool b1 = true,   b2 = false    int x = 1;

cout << !b1 << endl;                                    false

cout << !x << endl;                                     false

          false        true
cout << (!(b1) && (x >= 0)) << endl;                    false

            true          true
cout << ((x >= 0) && (x <= 100)) << endl;               true

           false          true
cout << ((x <= 0) || (x <= 100)) <<endl;                true

              true          true
cout << ((x + 2 <= 8) && !b2) << endl;                  true
```

Ex: A program that checks whether a number is divisible **by 2 and 3,** whether a number is divisible **by 2 or 3,** and whether a number is divisible by **2 or 3 but not both:**

```
int number;
cout << "Enter an integer: ";
cin >> number;

if (number % 2 == 0 && number % 3 == 0)
  cout << number << " is divisible by 2 and 3." << endl;
if (number % 2 == 0 || number % 3 == 0)
  cout << number << " is divisible by 2 or 3." << endl;
if ((number % 2 == 0 || number % 3 == 0) &&
     !(number % 2 == 0 && number % 3 == 0))
  cout << number << " divisible by 2 or 3, but not both.";
```

Ex: A program that lets the user enter a year and checks whether it is a leap year.

A year is a ***leap year*** if it **is divisible by 4** but **not by 100** or if it is **divisible by 400**. So you can use the following Boolean expression to check whether a year is a leap year:

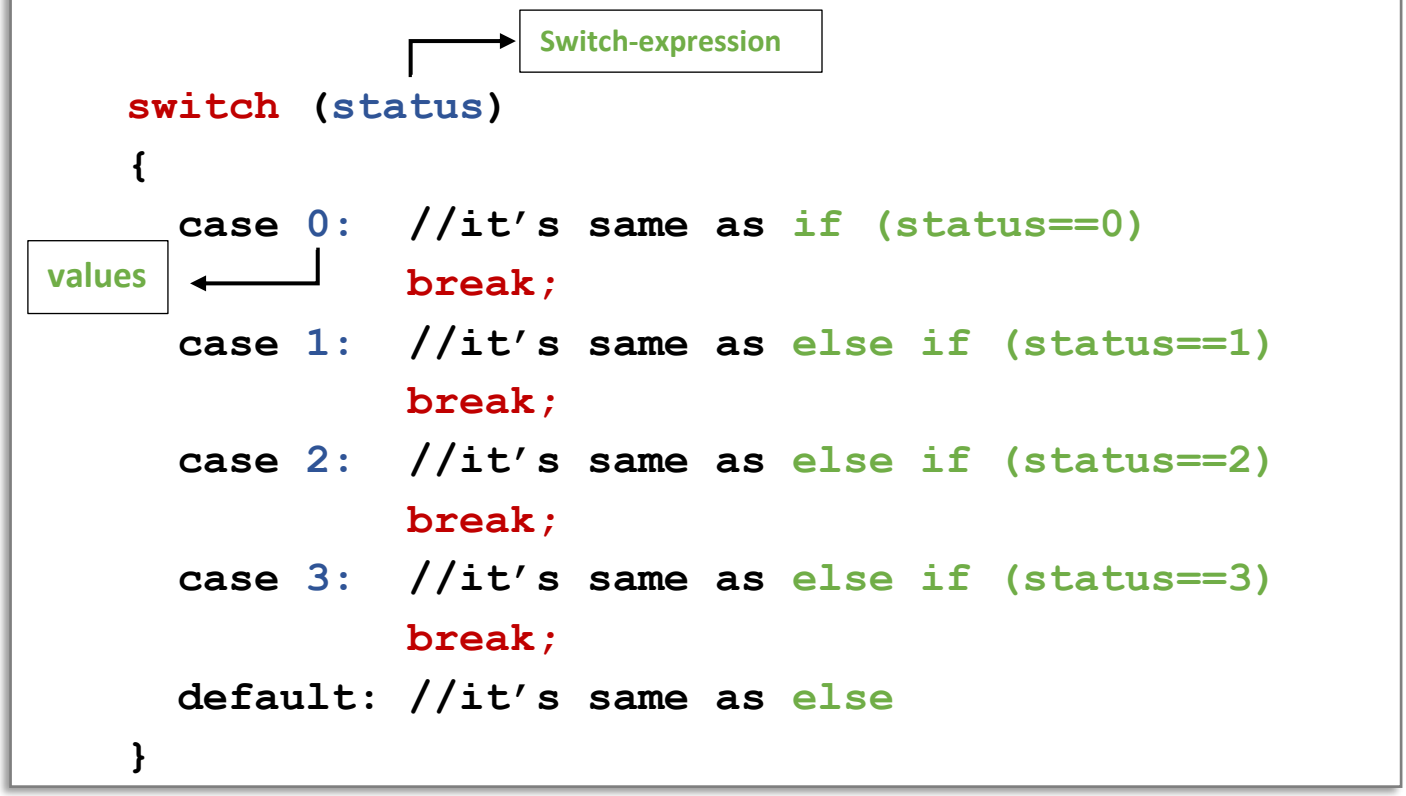
```
cout << "Enter a year: ";
int year;
cin >> year;

// Check if the year is a leap year
bool isLeapYear =
         (year % 4 == 0 && year % 100 != 0) ||
        (year % 400 == 0);

// Display the result in a message dialog box
if (isLeapYear)
  cout << year << " is a leap year" << endl;
else
  cout << year << " is a not leap year" << endl;
```

5) switch Statements

يوجد نوع اخر من الجمل الشرطية المفيدة ، الي بتساعد على كتابة كود اقصر
وهي **switch statements** :




```
switch (status)
{
   case 0:  //it's same as if (status==0)
            break;
   case 1:  //it's same as else if (status==1)
            break;
   case 2:  //it's same as else if (status==2)
            break;
   case 3:  //it's same as else if (status==3)
            break;
   default: //it's same as else
}
```

- شروط مهمة لل switch :

1) switch-expression لازم تكون رقم صحيح او حرف او جملة
ما بزبط تكون Boolean expression او عملية حسابية او كسر

2) الـ values لازم تكون رقم صحيح او حرف او جملة
وممنوع تحتوي على variable مثل : case (x + 1)

3) ال break بنحطها مع كل case حتى يخرج من الـ case بعد انتهاء تنفيذها
ولو ما كانت موجودة رح يفوت وينفذ الـ case الي بعدها حتى يوصل الـ default

Example: write a program that take a number from user between 1-7 and tell the user what the day of weekend is:

```
int day;
cout << "Enter number of day: ";
cin >> day;

switch (day)
{
     case 1: cout << "The day is Sunday" << endl;
          break;
     case 2: cout << "The day is Monday" << endl;
          break;
     case 3: cout << "The day is Tuesday" << endl;
          break;
     case 4: cout << "The day is Wednesday" << endl;
          break;
     case 5: cout << "The day is Thursday" << endl;
          break;
     case 6: cout << "The day is Friday" << endl;
          break;
     case 7: cout << "The day is Saturday" << endl;
          break;
     default :cout << "wrong number of day" << endl;
          break;
}
```

input: 3	input: -1	input: 8
Output: The day is Tuesday	**Output** : wrong number of day	**Output** : wrong number of day
تم تنفيذ ال case الثالثة	تم تنفيذ الـ default	تم تنفيذ الـ default

Conditional Expressions

يوجد اختصار للجمل الشرطية باستخدام ال conditional expressions :

(booleanExpression) ? statement1 : statement2

- اذا كان جواب ال **bool = true** رح يتم تنفيذ **statement1**
- اذا كان جواب ال **bool = false** رح يتم تنفيذ **statement2**

- Equivalent statements:

```
if (x > 0)
  y = 1;
else
  y = -1;
```

≡

```
y = x > 0 ? 1 : -1;
```

x = 1 --> y = 1

x = 0 --> y = -1

Ex1: Finding the max:

```
int max = (num1 > num2) ? num1 : num2;
```

num1 = 1, num2 = 2
max = num2 =2

num1 = 2, num2 = 1
max = num1 = 2

Ex2: Odd of even:

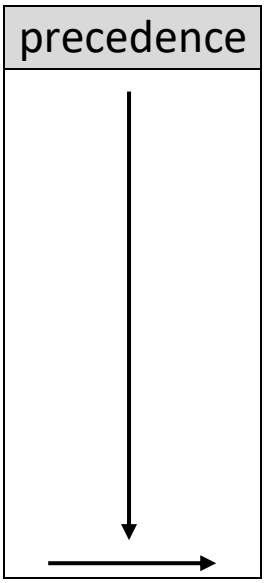
```
cout << (num % 2 == 0 ? "is even" : "is odd");
```

num = 2 or 4 or 6 ..
Output : is even

num = 1 or 3 or 5 ...
Output : is odd

Operator Precedence

- بعد ما تعرفنا تقريبا على جميع أنواع العمليات الحسابية ، بدنا نشوف ترتيبهم حسب الأولوية جميعهم مع بعض :

precedence	Operation	
	`var++, var--`	(postfix)
	`++var , --var`	(prefix)
	`static_cast<type>(v)`	(casting)
	`!`	(not)
	`* , / , %`	(mul, div, remain)
	`+ , -`	(add, sub)
	`< , <= , > , >=`	(relational)
	`== , !=`	(equality)
	`&&`	(and)
	`||`	(or)
	`=, +=, -=, *=, /=, %=`	(assignment operator)

دائما عند تساوي الاولويات ، الاولوية تكون الاولوية من اليسار لليمين

Ex: what is the output of the following expression :

bool x = (3 + 4 * 4 > 5 * (4 + 3) – 1 && (4 – 3 > 5)) ?

1) 3 + 4 * 4 > 5 * (4 + 3) – 1 && (4 – 3 > 5)
--> 3 + 4 * 4 > 5 * 7 – 1 && (1> 5)

2) 3 + 4 * 4 > 5 * 7 – 1 && (1> 5)
--> 3 + 4 * 4 > 5 * 7 – 1 && 0

3) 3 + 4 * 4 > 5 * 7 – 1 && 0
--> 3 + 16 > 35 – 1 && 0

4) 3 + 16 > 35 – 1 && 0
--> 19 > 34 && 0

5) 19 > 34 && 0
--> 0 && 0

6) 0 && 0
--> x is equal 0

Generating Random Numbers

- من الممكن الحصول على رقم عشوائي عن طريق استخدام rand() وهي function بتجيب رقم عشوائي من 0 ل 32,767.

- ومن الممكن تحديد **range** الارقام الي بحاجتها :

الحد الصغير ← a + rand() % b → الحد الكبير

تقوم بإرجاع رقم عشوائي بين الـ [a و a + b) .

Example **: generate a random number between :**

1 to 8: 1 + rand() % 8

[1,9) : a = 1 , a + b = 9

b = 9 – 1 = 8

a + rand() % b

--> 1 + rand() % 8

-3 to -1: -3 + rand() % 3

[-3,0) : a = -3, a + b = 0

b = 0 + 3 = 3

a + rand() % b

--> -3 + rand() % 3

25 + rand() % 32 : 25 to 56

a = 25, b = 32

[a, a+b) = [25, 57)

--> [25, 56]

or

25 + rand() % 32 :

rand() % 32 : 0 to 31

25 + [0,31] = [25, 56]

- مقارنة الكلمات (strings) :

- من الممكن مقارنة الـ strings مع بعضهم البعض عن طريق استخدام الـ boolean expression ، وبتكون الالية على النحو التالي :

- بقارن اول حرف من كل string ، والحرف الأكبر هو الـ string الأكبر

- اذا كان اول حرف متساوين ، بقارن ثاني حرف من كل string، اذا متساوين بقارن الحرف الثالث من كل string ، وهكذا

- اذا خلصت وحدة من ال strings وكانو كل الحروف متساوية بتكون الكلمة الي الها اكثر عدد أحرف هي الأكبر ، اذا عدد الاحرف متساوين معناته الكلمتين متساويين

Examples :

```
string s1, s2;
```

```
s1 = "Hi";
s2 = "Hello";

  H = H -> i > e

     s1 > s2
```

```
s1 = "sunday";
s2 = "monday";

    s > m

   s1 > s2
```

```
s1 = "abcd";
s2 = "abc";

a = a -> b = b
-> c = c -> s2 long

    s1 > s2
```

```
s1 = "C++";
s2 = "C++";

 C = C -> + = +
-> + = +

     s1 = s2
```

طبعا تكون المقارنة حسب الـascii table لكل حرف

- بعض الـ boolean expression المشهورة :

. x is non zero

(x != 0)

. x is even

(x % 2 == 0)

. x is positive

(x >= 0)

. x is odd

(x % 2 != 0)

. x is negative

(x < 0)

. x between num1 and num2

(x > num1 && x < num2)

. x is a small letter

(x >= 'a') && (x <= 'z')

. x is a capital letter

(x >= 'A') && (x <= 'Z')

Problems with answers:

- write the code of the following problems:

1) Write a program to find maximum between two numbers using if else.

Sample test input :	ouput:
Enter two numbers : 10 20	the maximum is: 20

2) Write a program to find maximum between three numbers using if else.

Sample test input:	output:
Enter three numbers: 20 30 10	max : 30 , mid : 20 , min : 10

3) Write a program to check positive, negative or zero using simple if else

Sample test input1 :	output1 :
Enter number: 23	23 is positive
Sample test input2 :	output2 :
Enter number: -12	-12 is negative

4) write a program to check the input is character or number

If the input is character check if it's vowel or not and is upper or lower case :

Sample test input1:	output1:
Enter your input : 1	1 is number
Sample test input2 :	output2 :
Enter your input : O	O is vowel and upper
Sample test input3 :	output3 :
Enter your input : w	w is not vowel and is lower

5) write a program that take 3 angle and check whether a triangle is valid or not

Sample test input :

Enter 3 angles : 60 30 90

output:

The triangle is valid

6) Write a program that take two numbers and show a Menu-Driven Program to perform a simple calculation: (1.Add , 2. Subtract , 3. Multiple , 4. Division).

Sample test input:

Enter two numbers: 4 2

Enter your option: (1.Add , 2. Subtract , 3. Multiple , 4. Division): 1

Output:

4 + 2 = 6

7) Write a program which is a Menu-Driven Program to compute the area of the various geometrical shape.

input 1	Area of circle	input radius	3.14 * r^2
input 2	Area of rectangle	input length and width	L * w
input 3	Area of triangle	input base and height	0.5 * b * h

Sample test input :

Enter your option area : (1. circle 2. rectangle 3. triangle): 1

Enter the radius of circle : 2

Output :

The area of circle is : 12.56

8) write a program that read temperature and display a message according to temperature sate :

Temp < 0	Freezing weather
Temp 0 – 10	Very Cold weather
Temp 10 – 20	Cold weather
Temp 20 – 30	Normal in Temp
Temp >= 30	Very Hot

Sample test input : output:

Enter the temperature : 33 Very Hot

9) write a program that read 3 numbers and displays them in increasing order

Sample test input : output :

Enter 3 numbers : 7 4 1 increasing order : 1 4 7

10) write a program that generate a two random numbers between 1 and 20, and ask the user to answer the add of the numbers :

Sample test input :

What is the sum of (num1) + (num2) ? (user input)

Output :

Right, correct answer!

or

Wrong!, try again!

11) **Write a program to calculate the salary as per the following table**

Gender	Year of Service	Qualifications	Salary
Male	>= 10	Post - Graduate	15000
	>= 10	Graduate	10000
	< 10	Post - Graduate	10000
	< 10	Graduate	7000
Female	>= 10	Post - Graduate	12000
	>= 10	Graduate	9000
	< 10	Post - Graduate	10000
	< 10	Graduate	6000

Sample test input :

Enter your gender (m/f): f

Enter the years of service: 5

Enter the qualifications (G, P): G

output:

Your salary is : 6000

- show the output of the following question :

Note : a =1 , b =2 , c = 3

1)
```
if (c >= a + b)
cout << "Hello";
cout << " There";
```

2)
```
if (a > b)
if (a > c)
    cout << "111";
else
    cout << "222";
```

3)
```
switch (a) {
        case 1: cout << "Saturday" << endl;
        case 2: cout << "Sunday" << endl;
            break;
        case 3: cout << "Monday" << endl;
            break;
        case 4: cout << "Tuesday" << endl;
        case 5: cout << "Wednesday" << endl;
            break;
        default: cout << "Invalid day" << endl;
            break;
}
```

4) What is the expected range for i based on the following statements?

```
int i =  rand ( ) %  15  - 5 ;
i -=   rand() % 17;
```

answers:

```
1)int x, y;
cout << "Enter two numbers: ";
cin >> x >> y;
if (x > y)
     cout << "The maximum is: " << x << endl;
else
     cout << "The maximum is: " << y << endl;

2) int x, y, z;
cout << "Enter three numbers: ";
cin >> x >> y >> z;
if (x > y && x > z)
{
     if (y > z)
      cout << "max:" << x << " mid:" << y << " min:" << z;
     else
      cout << "max:" << x << " mid:" << z << " min:" << y;
}
else if(y>x&&y>z)
{
     if (x > z)
       cout << "max:" << y << " mid:" << x << " min:" << z;
     else
       cout << "max:" << y << " mid:" << z << " min:" << x;
}
else
{
     if (x > y)
       cout << "max:" << z << " mid:" << x << " min:" << y;
     else
       cout << "max:" << z << " mid:" << y << " min:" << x;
}

3)int x;
cout << "Enter number: ";
cin >> x;
if (x > 0)
     cout << x << " is positive" << endl;
else if (x < 0)
     cout << x << " is negative" << endl;
else
     cout << x << " is zero" << endl;
```

```
4)char c;
cout << "Enter your input:";
cin >> c;
if (c >= '0' && c <= '9')
     cout << c << " is number" << endl;
else if (c >= 'a' && c <= 'z')
{
   if (c == 'a'||c == 'o'|| c == 'u'|| c == 'i'|| c == 'e')
       cout << c << " is vowel and lower" << endl;
   else
       cout << c << " is not vowel and is lower" << endl;
}
else
{
  if (c == 'A'||c == 'O'||c == 'U'||c == 'I'||c == 'E')
     cout << c << " is vowel and upper" << endl;
  else
     cout << c << " is not vowel and is upper" << endl;
}
```

```
6)int x, y;
cout << "Enter two numbers: ";
cin >> x >> y;

int option;
cout << "Enter your option:
(1.add,2.subtact,3.multiple,4.division)";
cin >> option;

switch (option)
{
case 1: cout << x << "+" << y << "=" << x + y;
     break;
case 2: cout << x << "-" << y << "=" << x - y;
     break;
case 3: cout << x << "*" << y << "=" << x * y;
     break;
case 4: cout << x << "/" << y << "=" << x / y;
     break;
default: cout << "Invalid option";
     break;
}
```

```
5)int angle1, angle2, angle3;
cout << "Enter 3 angles: ";
cin >> angle1 >> angle2 >> angle3;
if (angle1 + angle2 + angle3 == 180)
     cout << "The triangle is valid" << endl;
else
     cout << "The triangle is invalid" << endl;



7)int option;
cout << "Enter your option area : (1.circle, 2.rectangle,
3.triangle)";
cin >> option;
if (option == 1)
{
     int r;
     cout << "Enter the radius of circle: ";
     cin >> r;
     cout << "The area of circle is : " << 3.14 * r * r;
}
else if (option == 2)
{
     int l, w;
     cout << "Enter the length and the width : ";
     cin >> l >> w;
     cout << "The area of rectangle is : " << l * w;
}
else if (option == 3)
{
     int b, h;
     cout << "Enter the base and the height: ";
     cin >> b >> h;
     cout << "The area of triangle is : " << 0.5 * b * h;
}
else
     cout << "Invalid option" << endl;
```

```
8)int temp;
cout << "Enter the temperature : ";
cin >> temp;
if (temp < 0)
     cout << "Freezing weather";
else if (temp >= 0 && temp < 10)
     cout << "Very Cold weather";
else if (temp >= 10 && temp < 20)
     cout << "Cold weather";
else if (temp >= 20 && temp < 30)
     cout << "Normal in Temp";
else
     cout << "Very Hot";

9)int x, y, z;
cout << "Enter 3 numbers:";
cin >> x >> y >> z;

if (x > y && x > z)
{
     if (y > z)
          cout << "";
     else
     cout << " ";
}

10)int num1, num2;
num1 =  1 + rand() % 10;
num2 = 1 + rand() % 10;

int answer;
cout << "what is the sum of " << num1 << "+" << num2 <<
"?";
cin >> answer;
if (answer == num1 + num2)
     cout << "Right,correct answer!" << endl;
else
     cout << "Wrong, try again!" << endl;
```

```
11)char gender, qualif;
int years;
cout << "Enter your gender (f/m): ";
cin >> gender;
cout << "Enter the qualifications (G,P): ";
cin >> qualif;
cout << "Enter the years of service: ";
cin >> years;

if (gender == 'm')
{
     if (qualif == 'G')
     {
          if (years >= 10)
               cout << "Your salary is: 10000";
          else
               cout << "Your salary is: 7000";
     }
     else
     {
          if (years >= 10)
               cout << "Your salary is: 15000";
          else
               cout << "Your salary is: 10000";
     }

}
else
{
     if (qualif == 'G')
     {
          if (years >= 10)
               cout << "Your salary is: 9000";
          else
               cout << "Your salary is: 6000";
     }
     else
     {
          if (years >= 10)
               cout << "Your salary is: 12000";
          else
               cout << "Your salary is: 10000";
     }
}
```