TP2: Structure of a C++ Program

1. First Program

Problem: We want to write a program that adds two integer numbers.

Solution: To solve this problem, we follow the following steps (see the course):

1.1.Problem Analysis

In this step, we precisely identify the problem and define the data (inputs), results (outputs), and operations required to solve the problem.

For our problem:

- Input data: two integer numbers A and B.
- Output data: one integer number C.
- Operations: addition of the two integer numbers A and B (A+B).

1.2Writing the Algorithm

In this step, we define the order of operations using algorithmic language to obtain the results (outputs) from the data (inputs).

For our problem, the algorithm is as follows:

```
Algorithm Addition
A,B,C : integer;
BEGIN
A ← 15;
B ← 10;
C ← A + B;
Write(C);
END.
```

1.3. Writing the Program

In this step, we translate the algorithm into a programming language (C, Pascal, Fortran, etc.). In our case, we will use the C++ language. This process results in the program's source code, which is stored in one or more files on the computer.

The previous algorithm is translated into C++ as follows:

Algorithmic language	Language C++
Algorithm Addition A,B,C : integer; **begin** A ← 15; B ← 10; C ← A + B; Write (C); **End .**	#include <iostream> using namespace std; int A,B,C; **int main ()** **{**A=15; B=10; C=A+B; cout<< C; getchar(); return 0; **}**

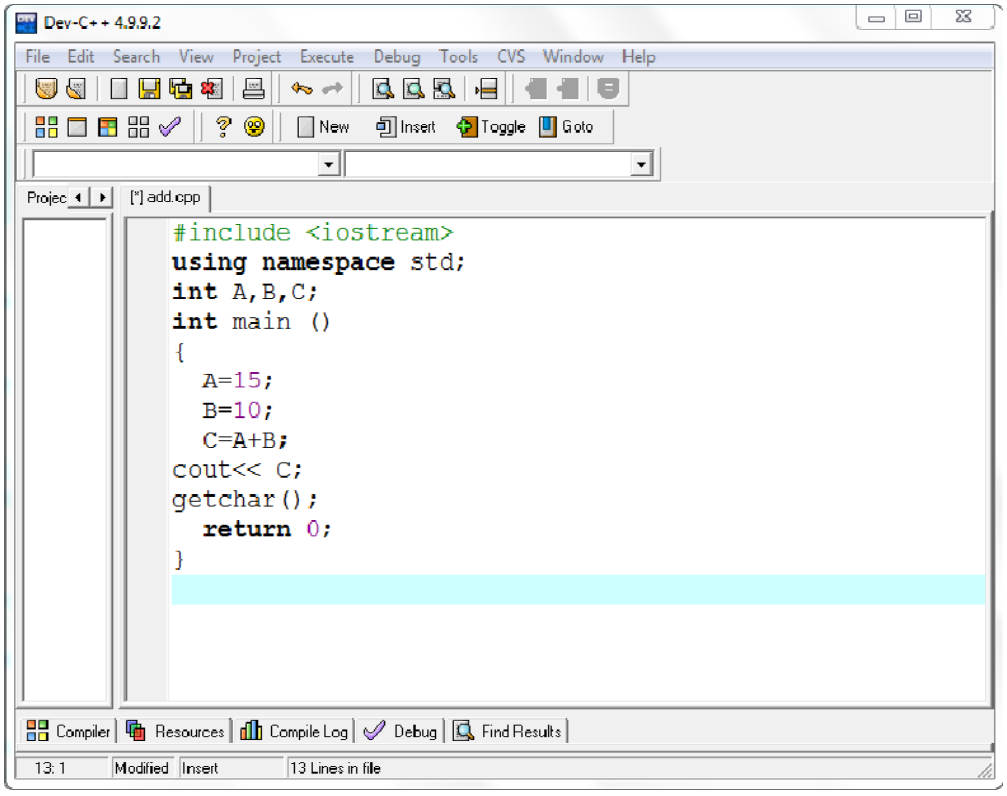
1.4. Compiling the Program

In this step, we translate the program's source code into machine language using special software called compilers. This process produces the machine code of the program, which is then executed directly by the computer.

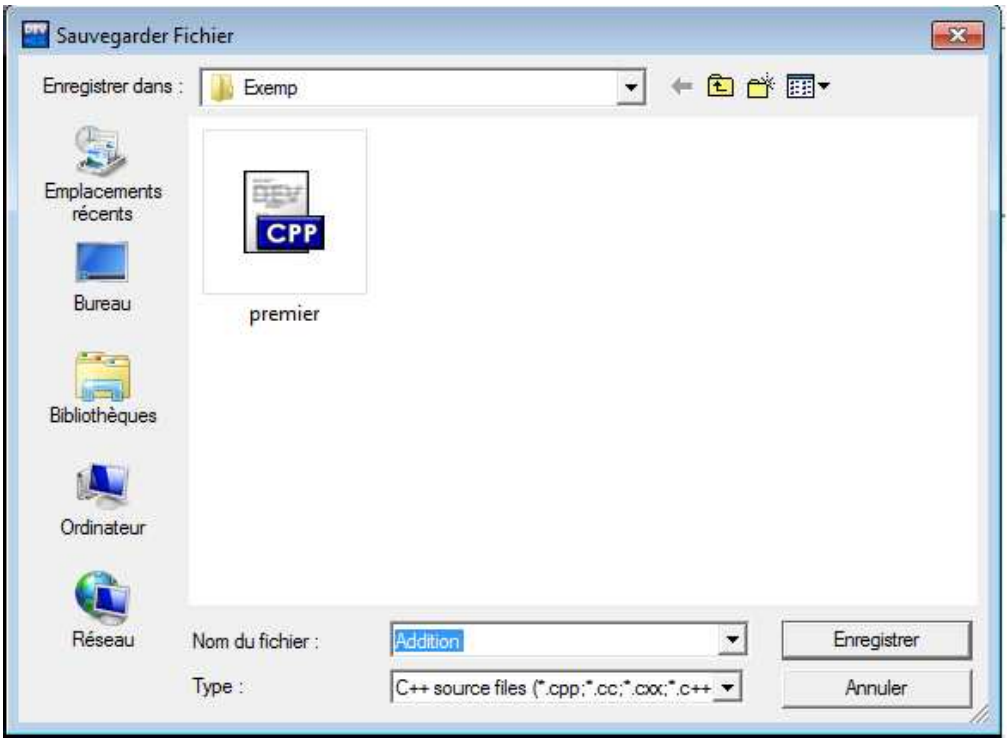
There are several software programs that integrate compilers for the C++ language, such as Dev C++, Turbo C++, Microsoft Visual C++ Express, and many others. These software programs also include text editors for writing code and various other utilities to assist in software development. In our case, we will use Dev C++.

To compile our program, we follow these steps:

1. Create a new source file by clicking on the "Source File" button (File → New → Source File).
2. Write the program using the text editor in Dev C++.



1. We save the program's source code to a file by clicking on the "Save" button (File → Save).

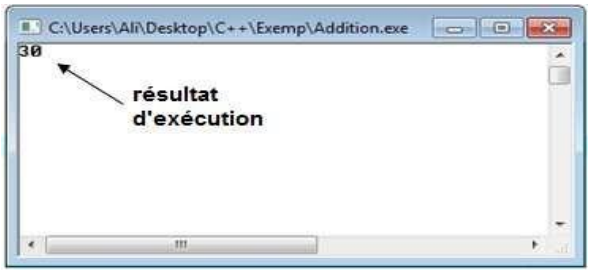


We compile the program's source code by clicking on the "Compile" button (Execute → Compile). The compiler will generate the program's machine code and save it in a file on the computer (the program's machine code is saved in the same location as the source code).

1.5. Execute the program

In this step, we run and test our program. To execute the program, we double-click the program's machine code file with the left mouse button.

- To run the program, we can also use the "Execute" button (Execute → Execute) in the Dev C++ menu.



Remarks:

- To create a new source file, click on the "Source File" button: (File → New → Source File), or use the keyboard shortcut Ctrl + N.
- To save a source file, click on the "Save" button (File → Save), choose the language used (in our case, C++), and the file location, then click the "Save" button. You can also use the keyboard shortcut Ctrl + S.
- To compile a source file, click on the "Compile" button (Execute → Compile), or use the keyboard shortcut Ctrl + F9.
- To execute a source file, click on the "Execute" button (Execute → Execute), or use the keyboard shortcut Ctrl + F10.
- To directly compile and execute a source file, click on the "Compile & Execute" button (Execute → Compile & Execute), or use the F9 keyboard shortcut.

TP: Write a program that calculates the sum of three integer numbers.

2. Modifying a Program:

We want to modify the previous program to allow keyboard input of data.

The algorithm :

```
Algorithm Addition2
A,B,C : integer;
begin
read(A);
read(B);
C ← A + B;
write(C);
end.
```

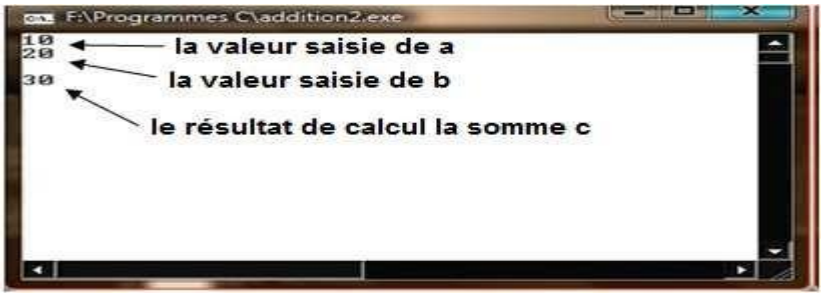
The program **C++ :**

```
#include <iostream>
using namespace std;
int main()
{
int a, b, c;
cin>> a;
cin>> b;
    c = a + b;
cout<< c;

getchar();
return 0;
}
```

Compiling the program :

executing the program



3. Adding Messages to the Program

We want to add some messages to the program to make it more user-friendly**.**

The Algorithm

```
Algorithm Addition3
A,B,C : integer;
Begin
write(' Please enter an integer:');
Read(A);
write (' Please enter a second integer:');
Lire(B);
C ← A + B;
Write('The sum of the two numbers is: ');
Write(C);
End.
```

The C++ program :

```
#include <iostream>
using namespace std;
int main()
{
int a, b, c;
cout<< " Please enter an integer:";
cin>> a;
cout<< " Please enter a second integer:"; cin>>
b;
c = a + b;
cout<< "the sum of the two numbers is :";
cout<< c;

getchar();
```

Compiling the program :

➢ **Correction of programming errors :**

Dev C++ displays the error messages of the program in a separate window below the editor.

The error message contains the line of the error and a description that helps to correct the error. Messages preceded by [Warning] are warnings, and the program's source code can be compiled even if they are not corrected.

Examples:

Program 1:

```
#Include <iostream>
using namespace std;
int main()
{
int a, b, c;
cout<< "enter an integer please :";
cin>> a;
cout<< "enter a second integer please :";
cin>> b;
c = a + b;
cout<< "the sum of the two numbers is  :";
cout>> c; // avec cout on utilise « << » au lieu de « >> »
getchar(); return 0;
}
```

C:\...\add.cpp no match for 'operator>>' in 'std::cout >> c'

Program 2 :

```
#Include <iosream> // iostream and not iosream
using namespace std;
int main()
{
int a, b, c;
cout<< "enter an integer please :";
cin>> a;
cout<< "enter a second integer please :"; cin>> b;
c = a + b;
cout<< "the sum of the two numbersis: ";
cout<< c;
getchar();
return 0;
}
```

C:\...\Ciosream: No such file or directory.

Program 3 :

```
include<iostream>
using namespace std;
intmain()
{
int a, b, c
cout<< "enter an integr please :";
cin>> a;
cout<< "enter a second integer please :"; cin>> b;
c = a + B; // B is different of b (uppercase and lowercase)
cout<< "the sum of the 2 numbers is :";
```

10 C:\...\addition1.cpp `B' undeclared (first use this function)

- **Program4 :**

```
#include <iostream> using
namespace std; int main()
{
int a, b, c
coutt<< "enter an integer please :"; // cout et pas coutt
cin>> a;
cout<< "enter a second integer please :"; cin>> b;
c = a + B;
cout<< "the sum of the two numbers :";
cout<< c;
cin.get();
return 0;
}
```

6 C:\...\addition1.cpp `coutt' undeclared (first use this function)

```
#include <iostream>
using namespace std;
int main()
{
int a, b, c
cout<< "enter an integer please :";
cin>> a;
cout<< "enter a second integer please :";
cin>> b;
c = a + B;
```

15 C:\...\addition1.cpp expected `}' at end of input

Note:

The number of error messages displayed does not necessarily mean there are that many errors in the program. For example, an error in the line #include <iostream> implies errors in the lines of code that contain the keywords "cin" or "cout." Once the error in the first line is corrected, the other errors will disappear.

TP :

1) Write a C++ program that calculates the sum of 3 integer numbers. The program should use messages to facilitate its usage.

2) Modify the previous program to display the following message:

"The sum of 4 and -12 and +7 is: -1"

Where: 4, -12, and +7 are the values entered by the user for the respective variables a, b, c, and -1 is the result.