



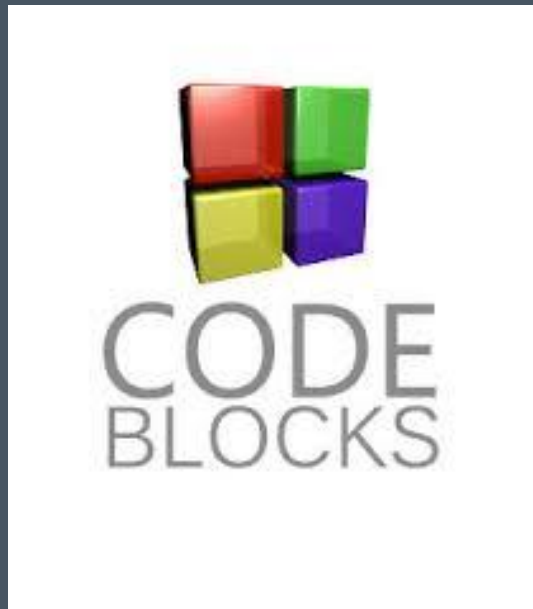
# Fundamentos de programación

*Departamento de  
tecnologías de la  
Información*



Práctica 0: Entorno de trabajo  
CodeBlocks v. 16.01

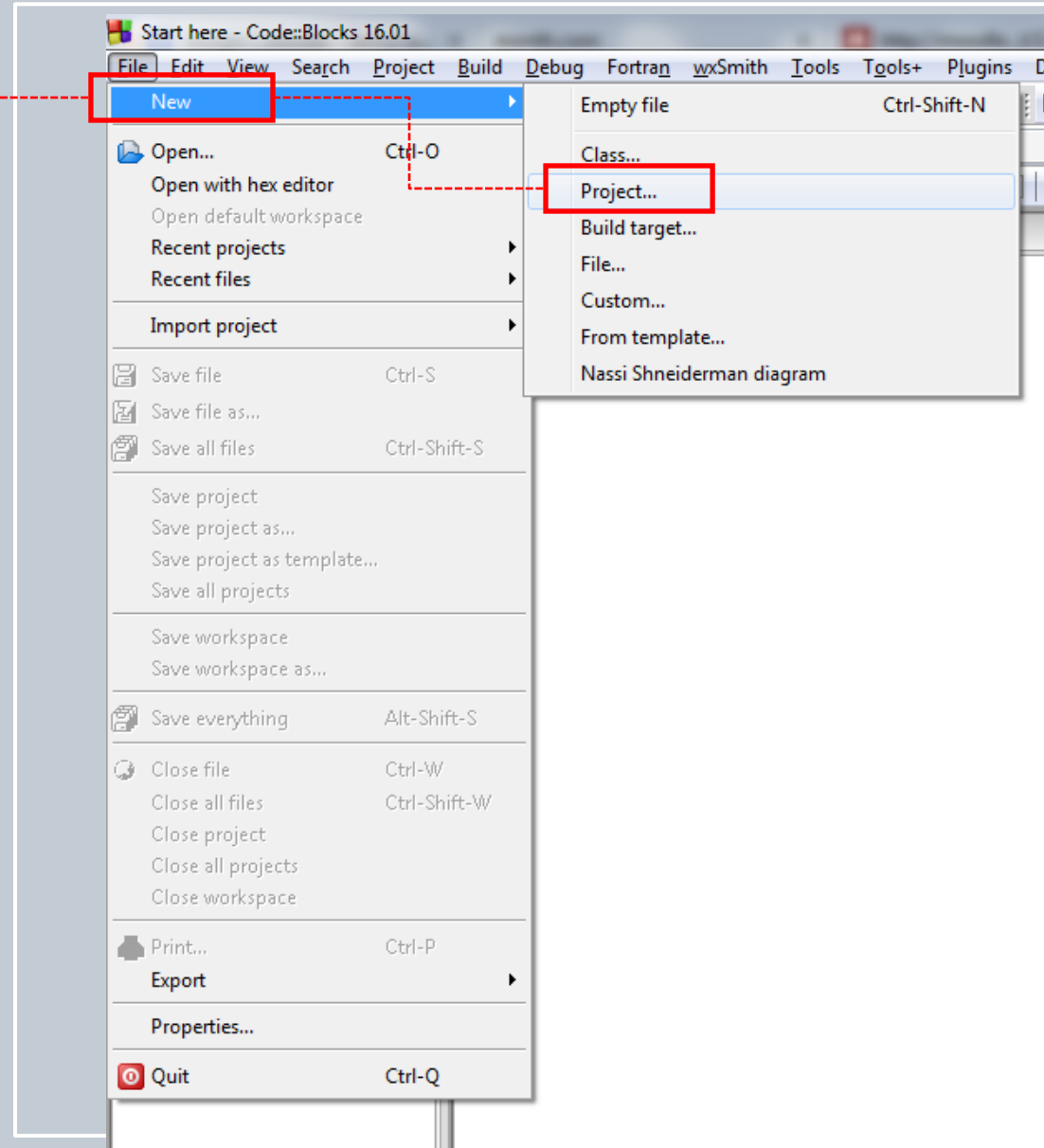
# ENTORNO DE TRABAJO CODEBLOCK



- › Crear un proyecto
- › Compilar y ejecutar
- › Depurar

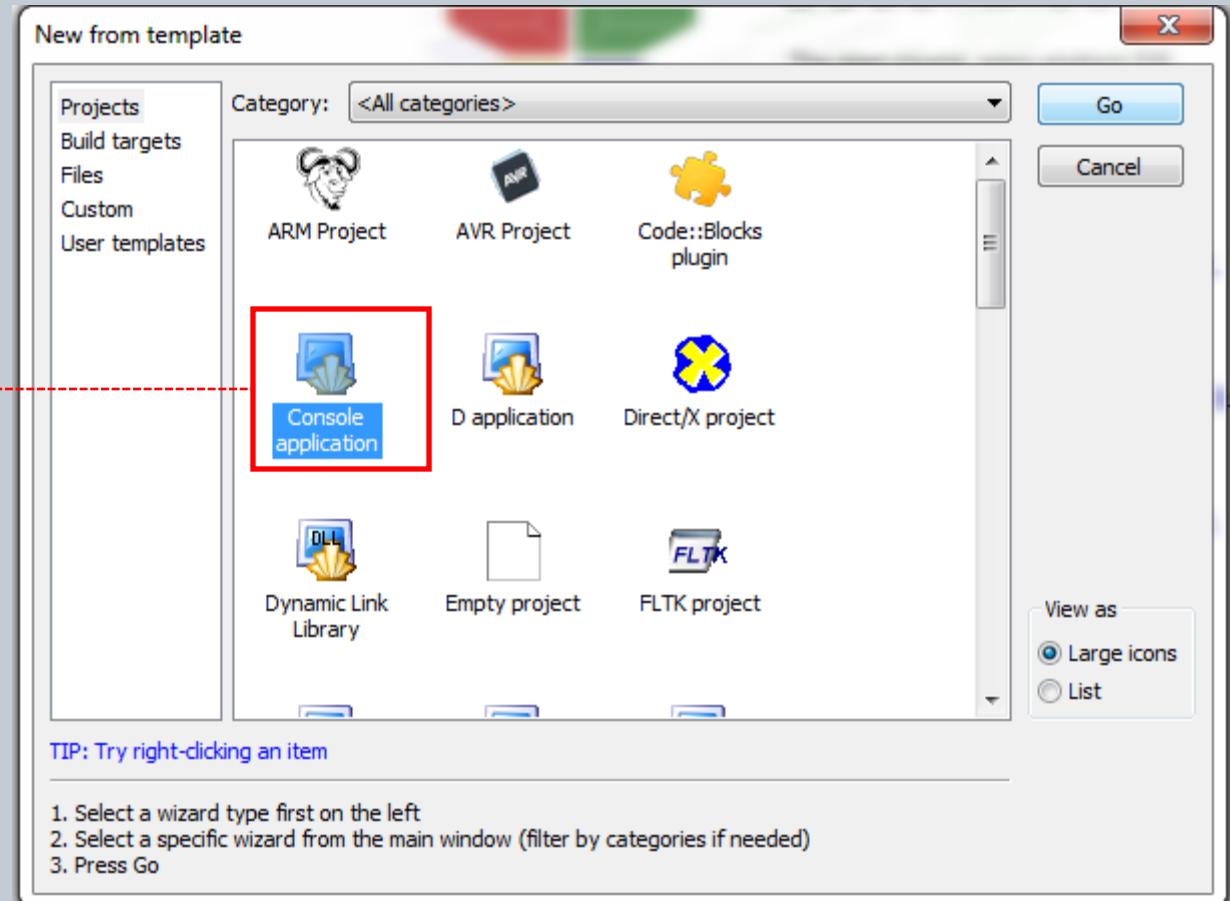
# CREACIÓN DE UN PROYECTO NUEVO

1. File → New → Project
2. Console application
3. C++ Next
4. Damos nombre y espacio de trabajo
5. Next
6. Seleccionamos main.cpp y escribimos el programa



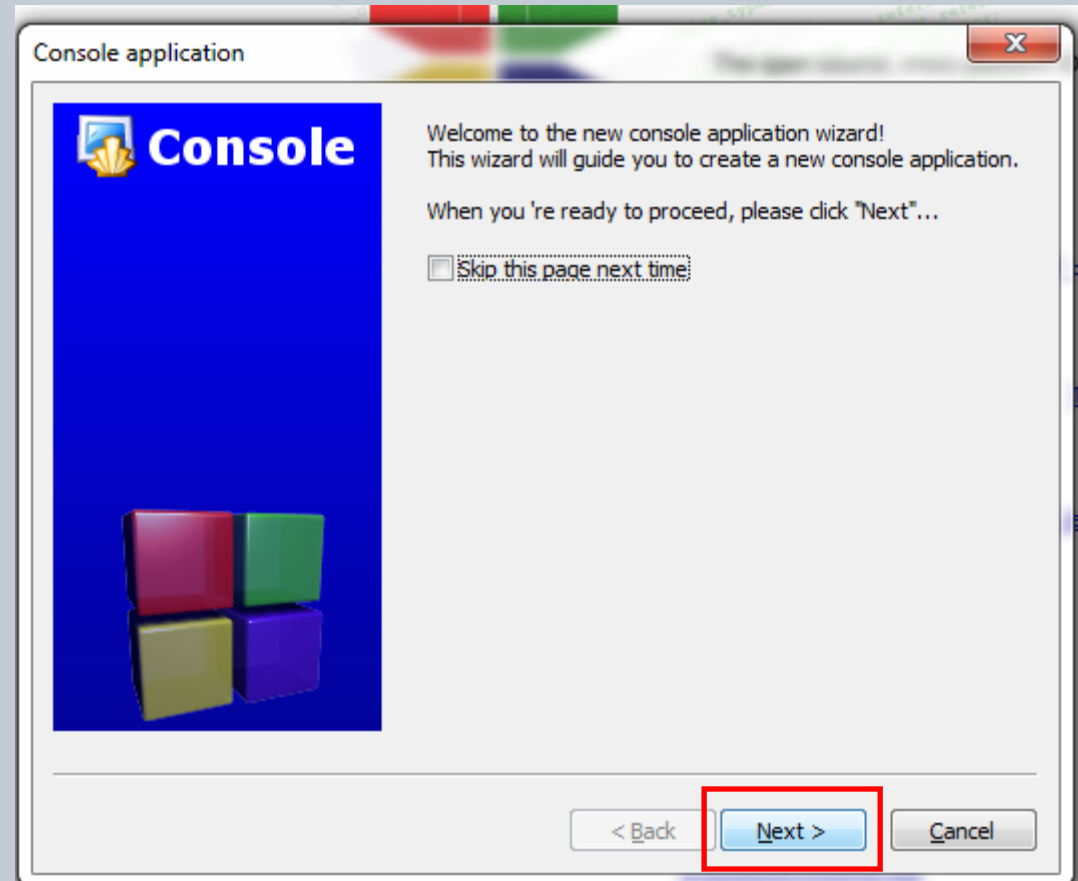
# CREACIÓN DE UN PROYECTO NUEVO

1. File→New→Project
2. Console application
3. C++ Next
4. Damos nombre y espacio de trabajo
5. Next
6. Seleccionamos main.cpp y escribimos el programa



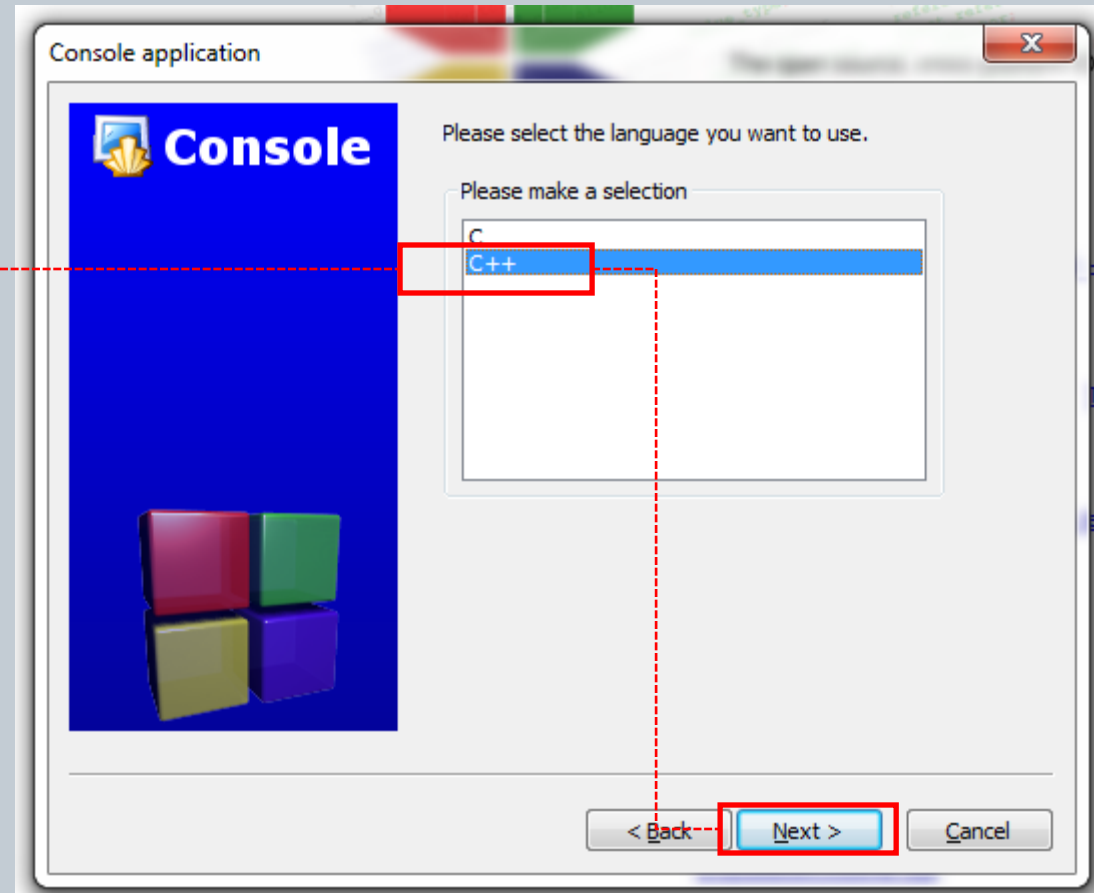
# CREACIÓN DE UN PROYECTO NUEVO

1. File→New→Project
2. Console application
3. C++ Next
4. Damos nombre y espacio de trabajo
5. Next
6. Seleccionamos main.cpp y escribimos el programa



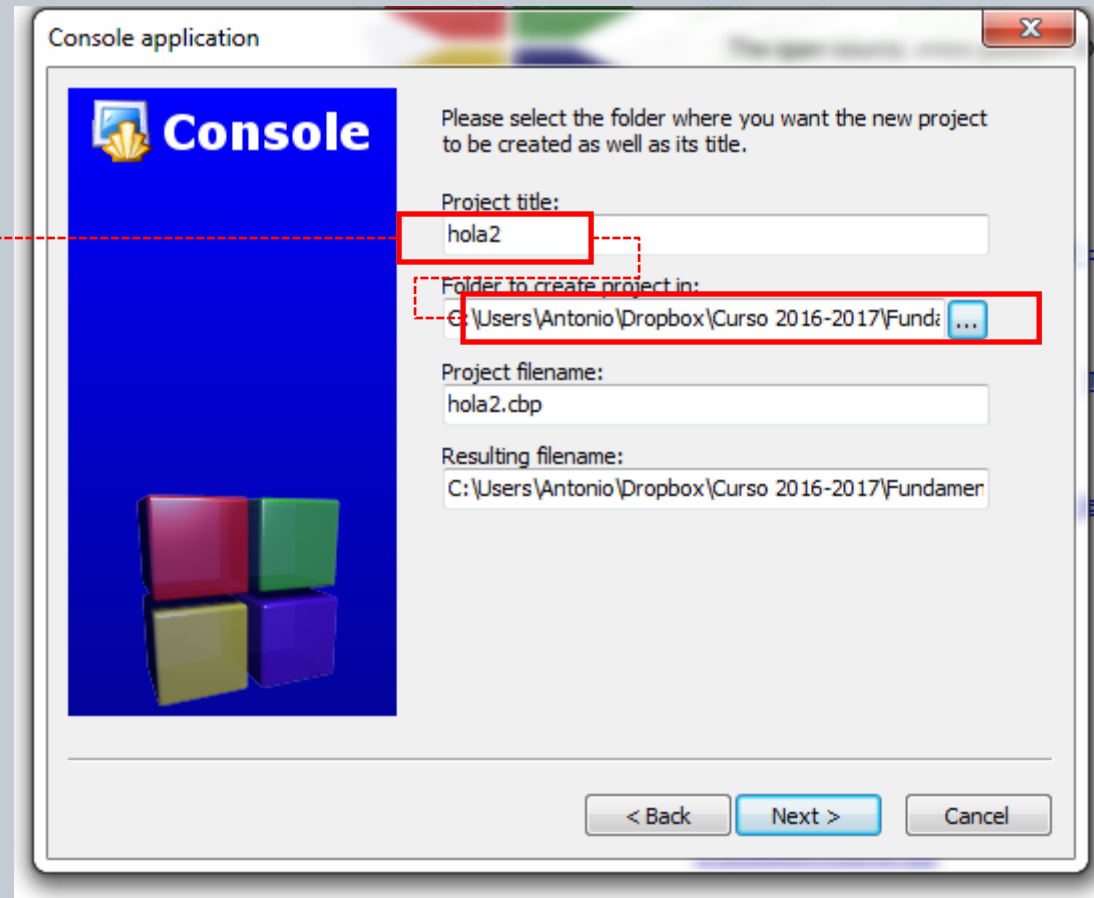
# CREACIÓN DE UN PROYECTO NUEVO

1. File→New→Project
2. Console application
3. C++ Next
4. Damos nombre y espacio de trabajo
5. Next
6. Seleccionamos main.cpp y escribimos el programa



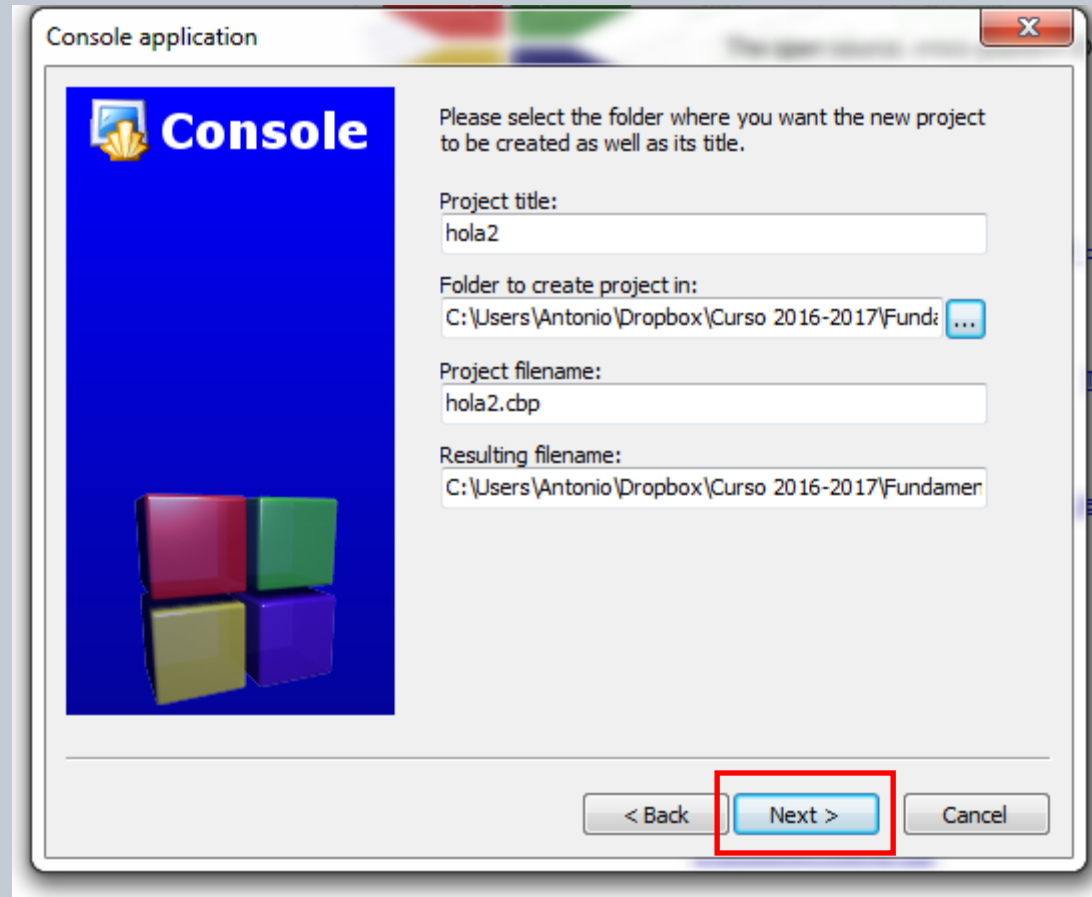
# CREACIÓN DE UN PROYECTO NUEVO

1. File→New→Project
2. Console application
3. C++ Next
4. Damos nombre y Espacio de trabajo
5. Next
6. Seleccionamos main.cpp y escribimos el programa



# CREACIÓN DE UN PROYECTO NUEVO

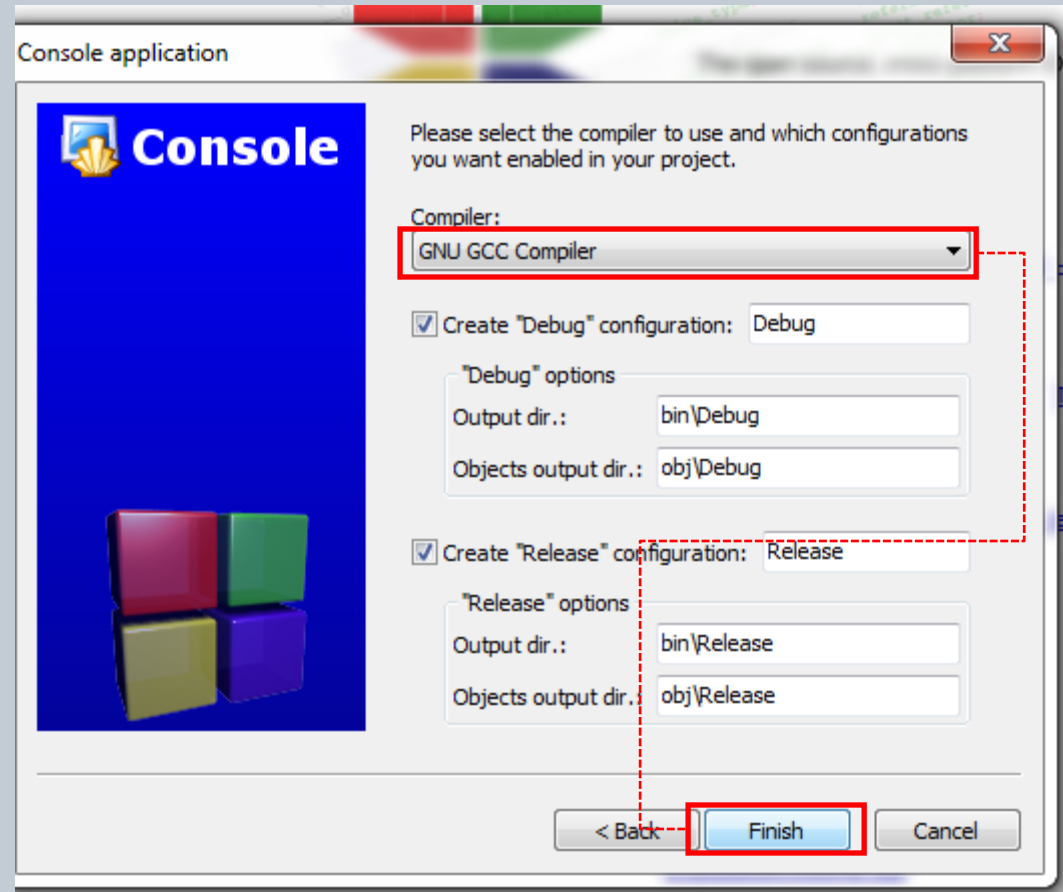
1. File→New→Project
2. Console application
3. C++ Next
4. Damos nombre y Espacio de trabajo
5. Next
6. Seleccionamos main.cpp y escribimos el programa





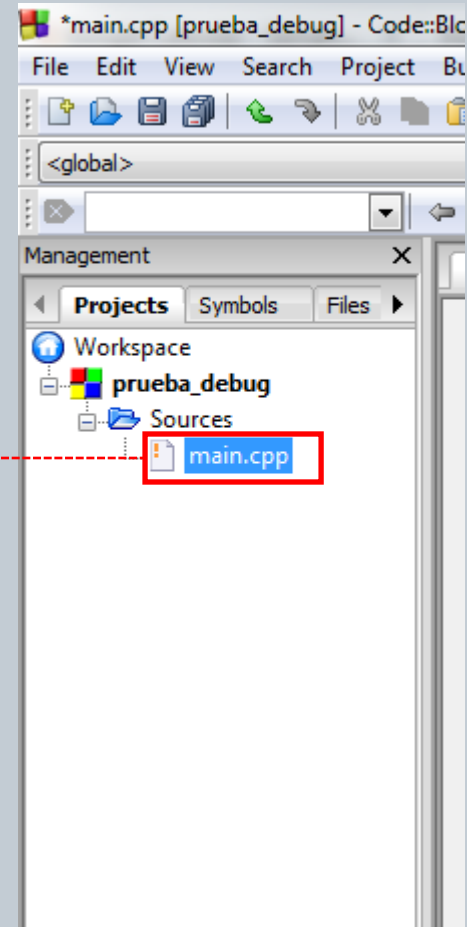
# CREACIÓN DE UN PROYECTO NUEVO

1. File→New→Project
2. Console application
3. C++ Next
4. Damos nombre y Espacio de trabajo
5. Next
6. Seleccionamos main.cpp y escribimos el programa



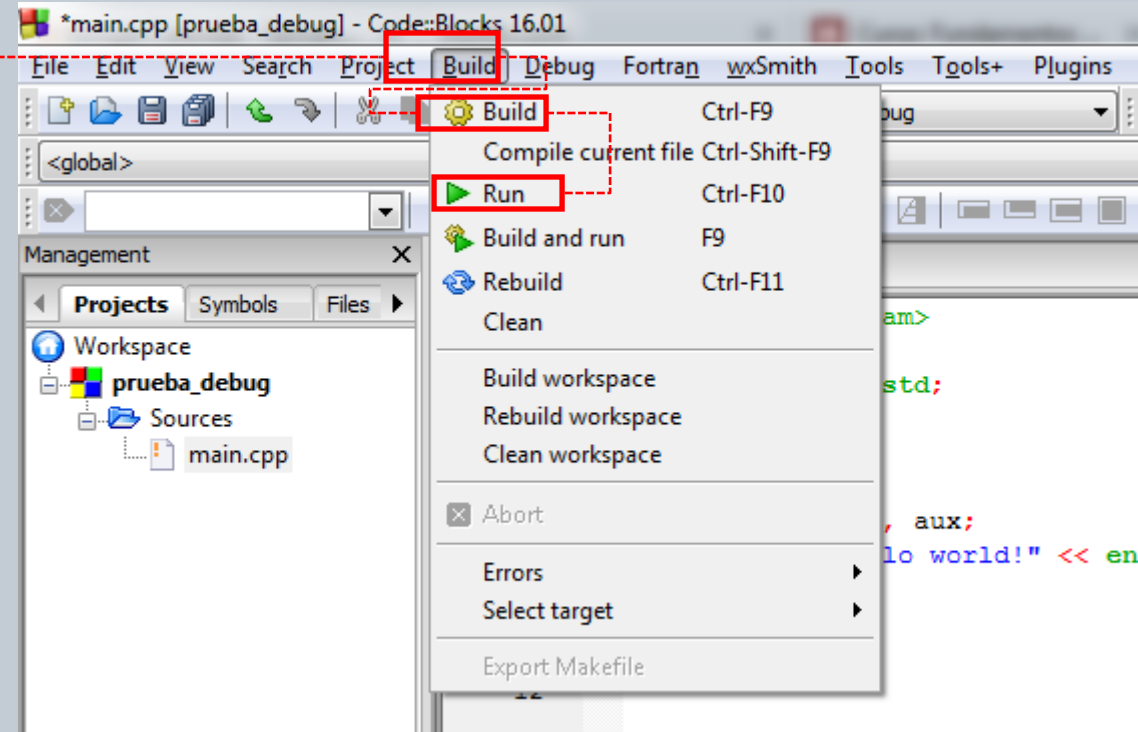
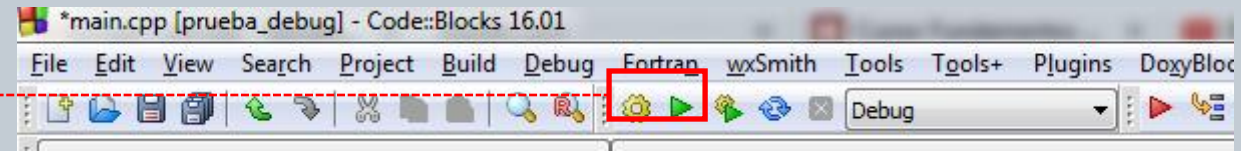
# EJECUTAR UN PROGRAMA

1. Escribir el código.  
Dándole doble click a main.cpp
2. Para ejecutar el programa primero botón Build y después botón Run o en el menú build→build→Run
3. Con la tecla *Enter* cerramos la ventana en la que se ejecuto.
4. Será necesario realizar estos pasos cada vez que modifiquemos el programa



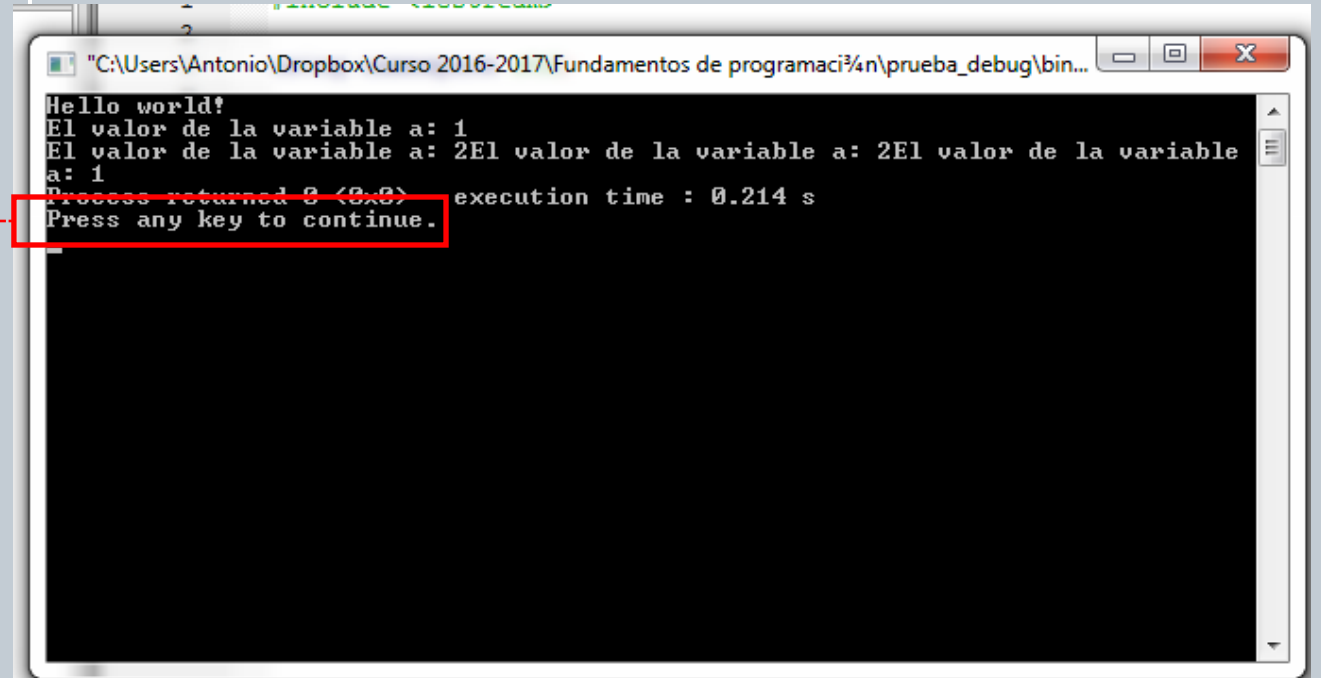
# EJECUTAR UN PROGRAMA

1. Escribir el código.  
Dándole doble click a main.cpp
2. Para ejecutar el programa primero botón Build y después botón Run o en el menú Build Build and Run
3. Con la tecla *Enter* cerramos la ventana en la que se ejecutó.
4. Será necesario realizar estos pasos cada vez que modifiquemos el programa



# EJECUTAR UN PROGRAMA

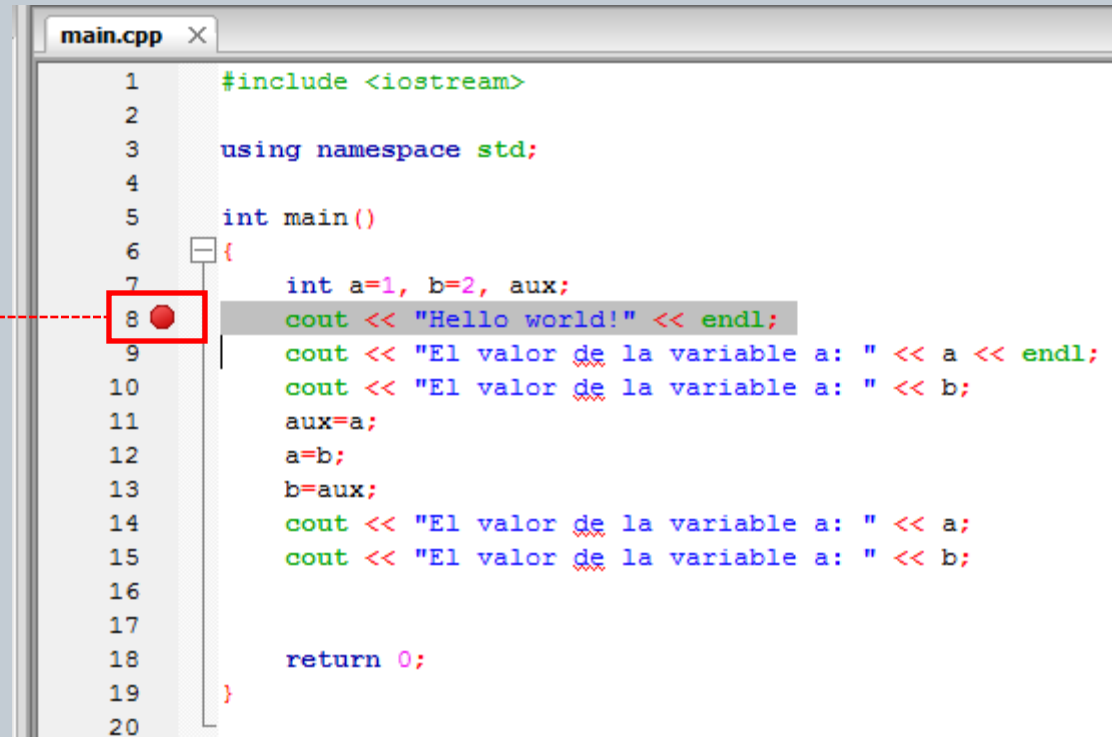
1. Escribir el código.  
Dándole doble click a main.cpp
2. Para ejecutar el programa primero botón Build y después botón Run o en el menú Build Build and Run
3. Con cualquier tecla cerramos la ventana en la que se ejecuto.
4. Será necesario realizar estos pasos cada vez que modifiquemos el programa



```
"C:\Users\Antonio\Dropbox\Curso 2016-2017\Fundamentos de programaci3n\prueba_debug\bin...
Hello world!
El valor de la variable a: 1
El valor de la variable a: 2El valor de la variable a: 2El valor de la variable
a: 1
Process returned 0 (0x0) execution time : 0.214 s
Press any key to continue.
```

# DEPURAR UN PROGRAMA

1. Ponemos un punto de ruptura donde queramos que se pare el programa.
2. Para ello hacemos click al lado del número de instrucción.



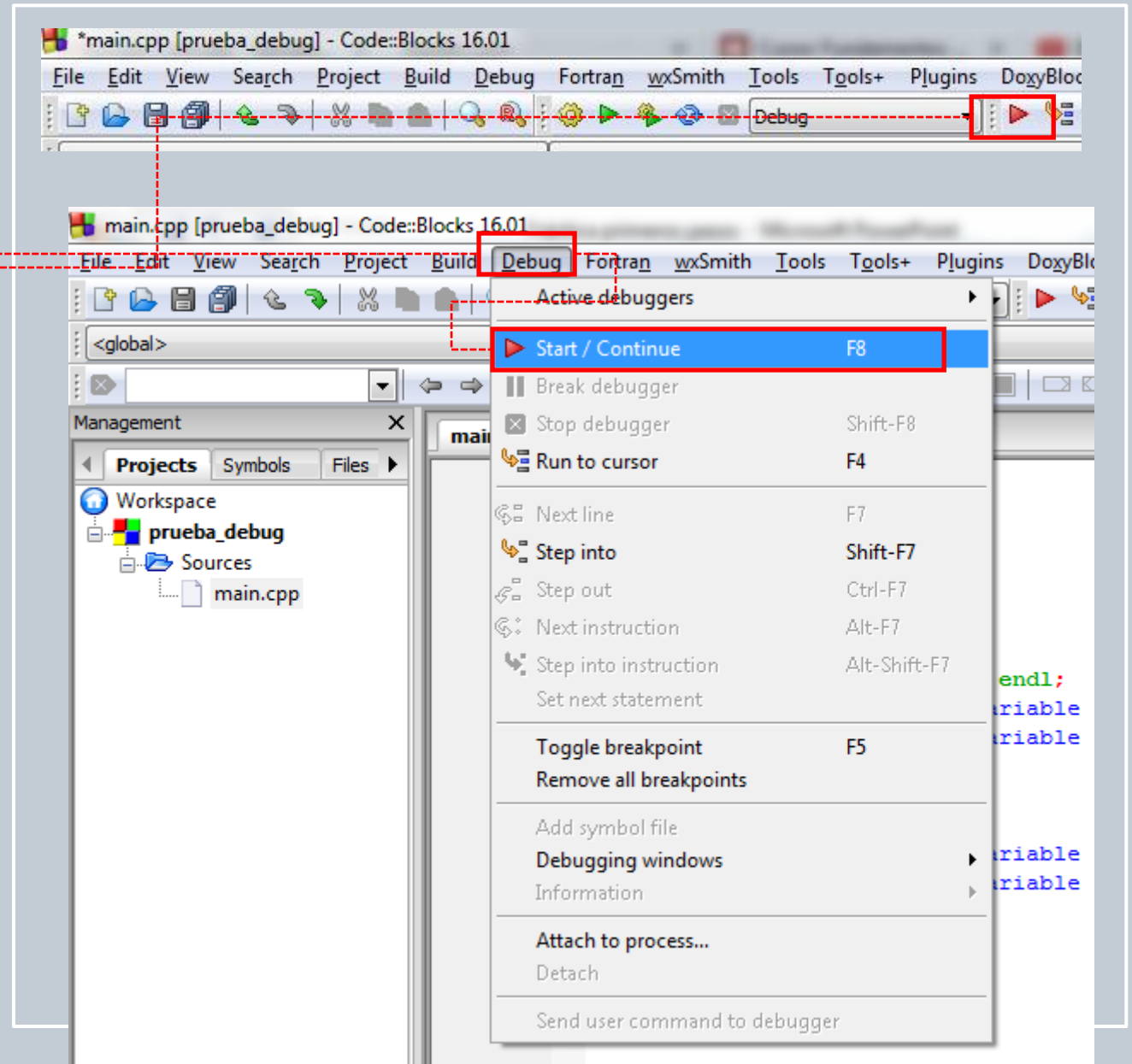
The screenshot shows a code editor window titled 'main.cpp'. The code is as follows:

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int a=1, b=2, aux;
8      cout << "Hello world!" << endl;
9      cout << "El valor de la variable a: " << a << endl;
10     cout << "El valor de la variable a: " << b;
11     aux=a;
12     a=b;
13     b=aux;
14     cout << "El valor de la variable a: " << a;
15     cout << "El valor de la variable a: " << b;
16
17
18     return 0;
19 }
20
```

A red square highlights the number '8' in the left margin, and a red dot is placed next to it, indicating a breakpoint. A dashed red line connects this point to the second step of the list on the left.

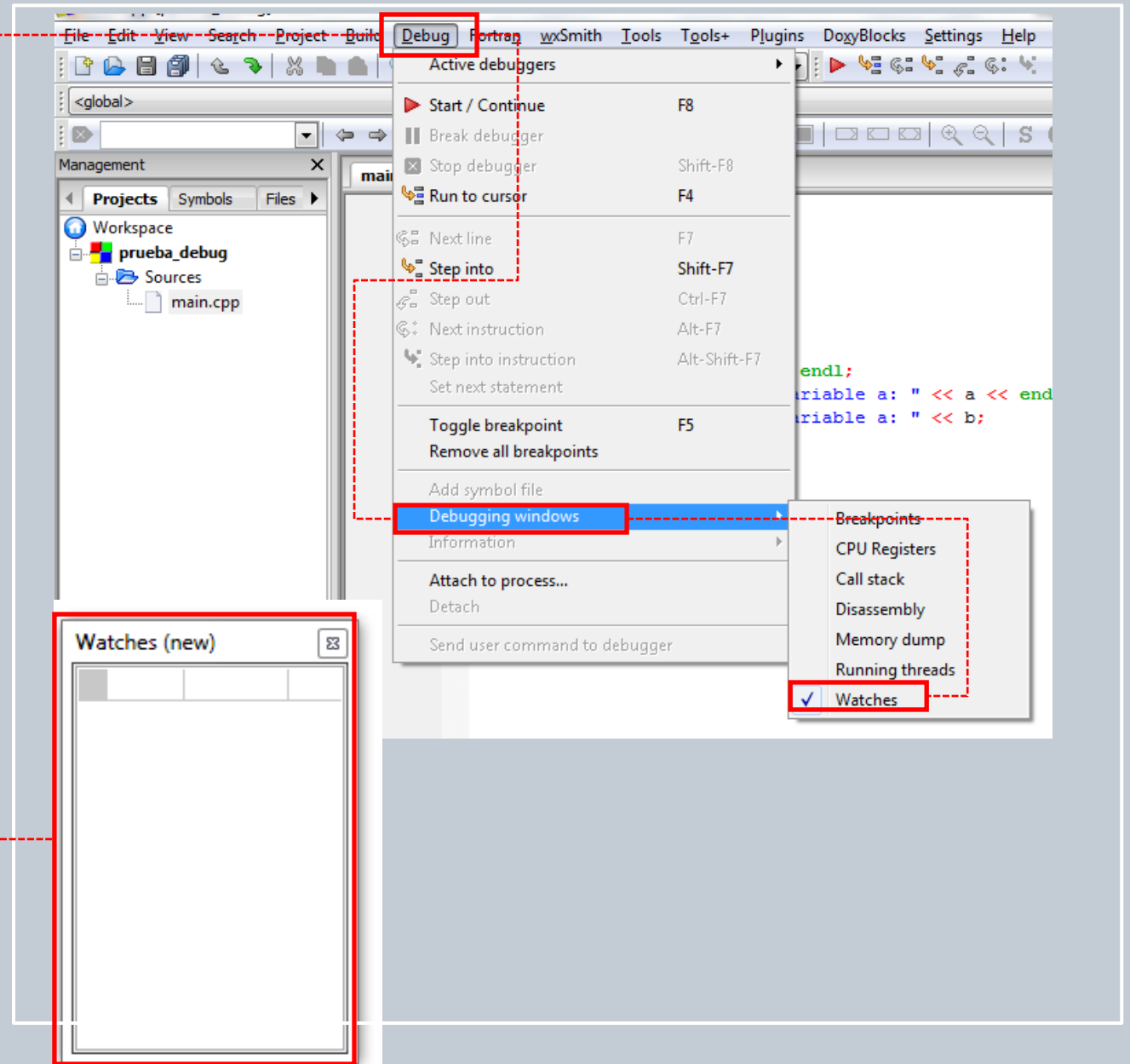
# DEPURAR UN PROGRAMA

1. Para el debug pulsamos botón o menú Debug→Start.
2. Para ver los valores de las variables ponemos en el menú de Debug y en la opción Debugging Windows marcar Watches. Esto nos abre una ventana donde podemos ver las variables y sus valores
3. Descripción de los comandos de debug



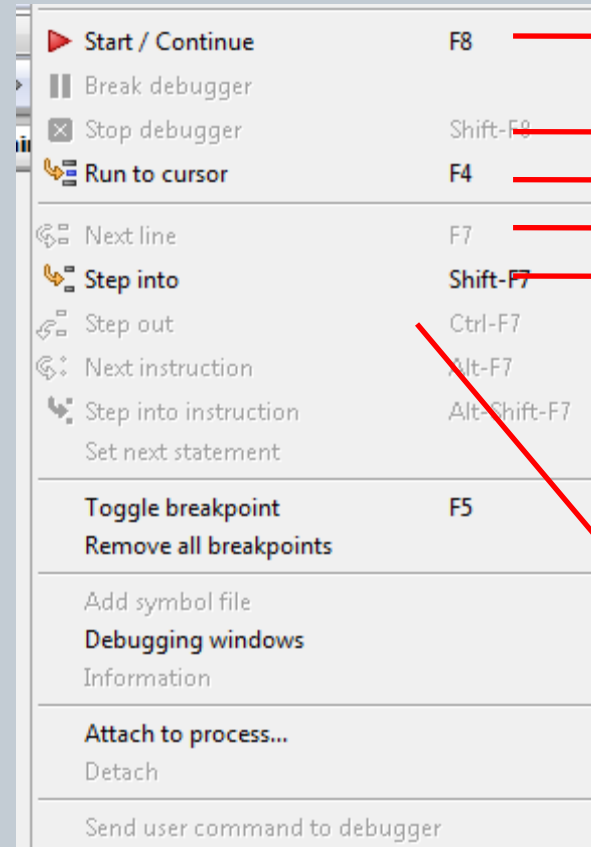
# DEPURAR UN PROGRAMA

1. Para el debug pulsamos botón o menú Debug→Start
2. Para ver los valores de las variables ponemos en el menú de Debug y en la opción Debugging Windows marcar Watches. Esto nos abre una ventana donde podemos ver las variables y sus valores
3. Descripción de los comandos de debug



# DEPURAR UN PROGRAMA

1. Para el debug pulsamos botón o menú Debug→Start
2. Para ver los valores de las variables ponemos en el menú de Debug y en la opción Debugging Windows marcar Watches. Esto nos abre una ventana donde podemos ver las variables y sus valores
3. Descripción de los comandos de debug



Para comenzar/continuar el debug

Termina la depuración

Ejecuta el programa hasta el cursor

Avanza una línea la ejecución.

En caso de estar en una línea que es una llamada a una función, con esta opción podemos hacer el seguimiento dentro de esta función

Considera las llamadas a funciones como una única instrucción