

Actividad 1. Manejo del entorno de desarrollo (IDE).

Objetivos:

- Adquirir soltura en el manejo del entorno de desarrollo (*IDE –Integrated Development Environment*) Eclipse for C/C++ developers junto con el compilador para C++ MinGW.
- Aprender a usar el depurador (*debugger*) de programas.

NOTA:

Previamente deberá haber descargado e instalado tanto el “compilador” (*toolchain*) como el propio entorno de desarrollo (*IDE*). Para más información consultar:

- Compilador para C++ MinGW:
 - Descarga: <http://sourceforge.net/projects/mingw/files/>
 - Pasos a seguir: <http://www.ics.uci.edu/~pattis/common/handouts/mingweclipse/mingw.html>
- Eclipse for C++ developers:
 - <https://eclipse.org/downloads/packages/release/Neon/R>

Desarrollo de la práctica:

1. El “banco de pruebas” (*Workbench*).

1.1. Interfaz de usuario.

Inicie el entorno Eclipse. La pantalla deberá mostrar el aspecto de la figura 1. Observe el icono **Workbench** que le permitirá acceder al entorno de trabajo.



Figura 1. Acceso a Eclipse

La figura 2 muestra el aspecto del entorno de trabajo, de momento vacío. Dedique unos minutos a familiarizarse con él¹.

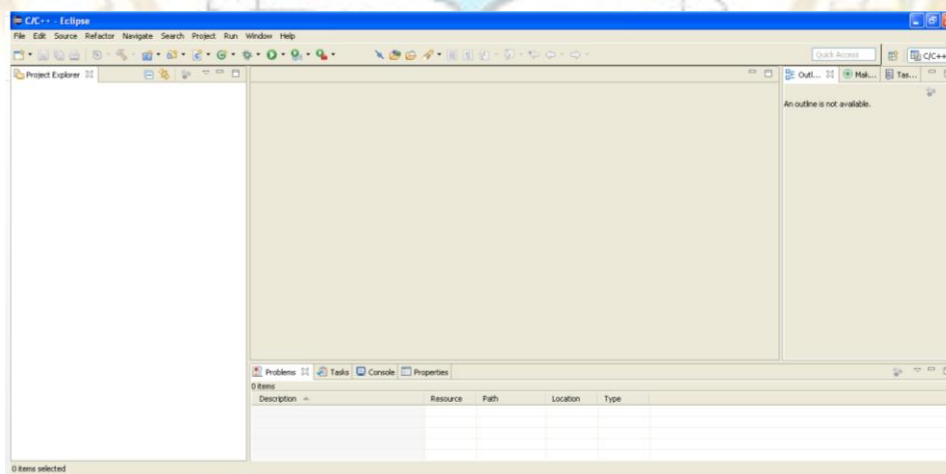


Figura 2. Entorno de trabajo.

1.2. Creación del “banco de pruebas” personal.

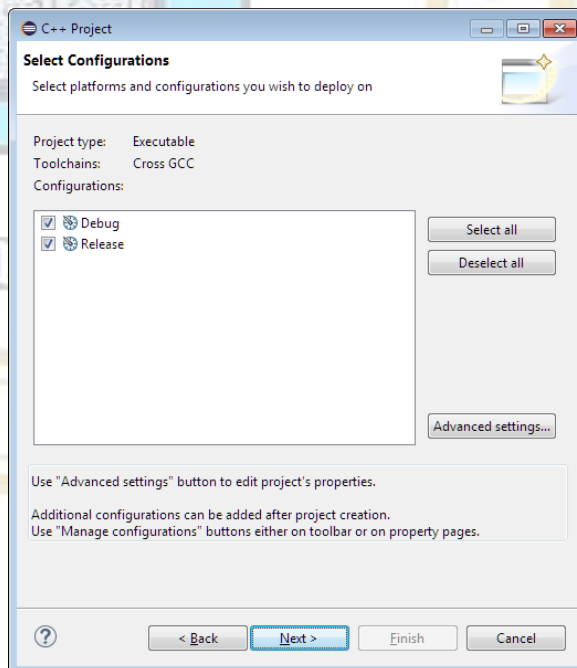
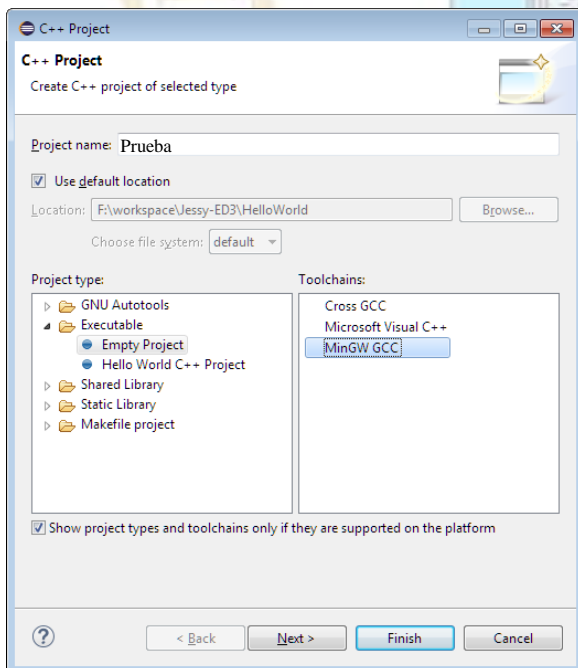
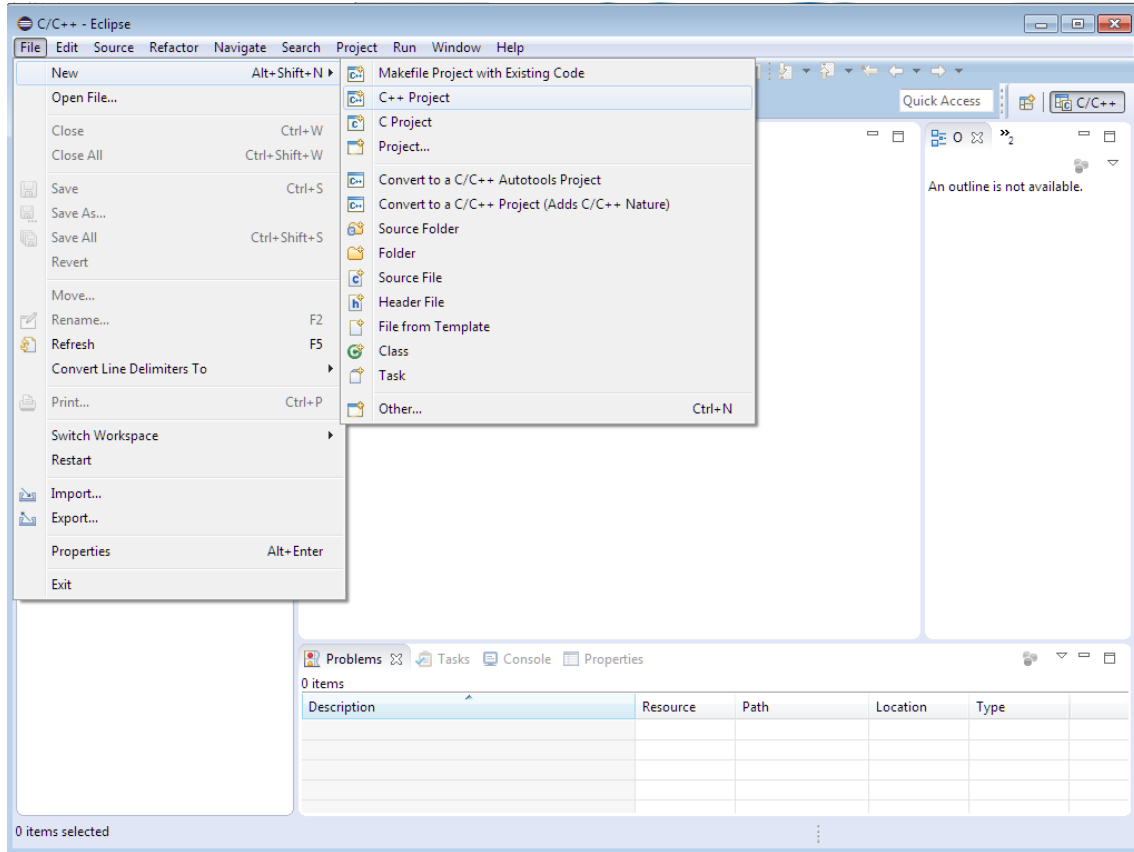
Seleccione *File -> Switch Workspace -> Other... TallerED* (y preste atención a dónde lo deja).

¹ Es posible que como consecuencia de manipulaciones pierda esta configuración. Siempre podrá volver a ella con *Window -> Reset Perspective*.



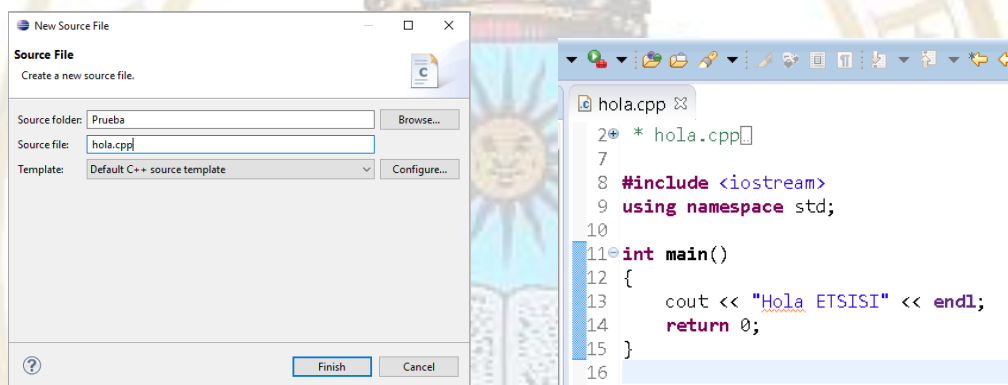
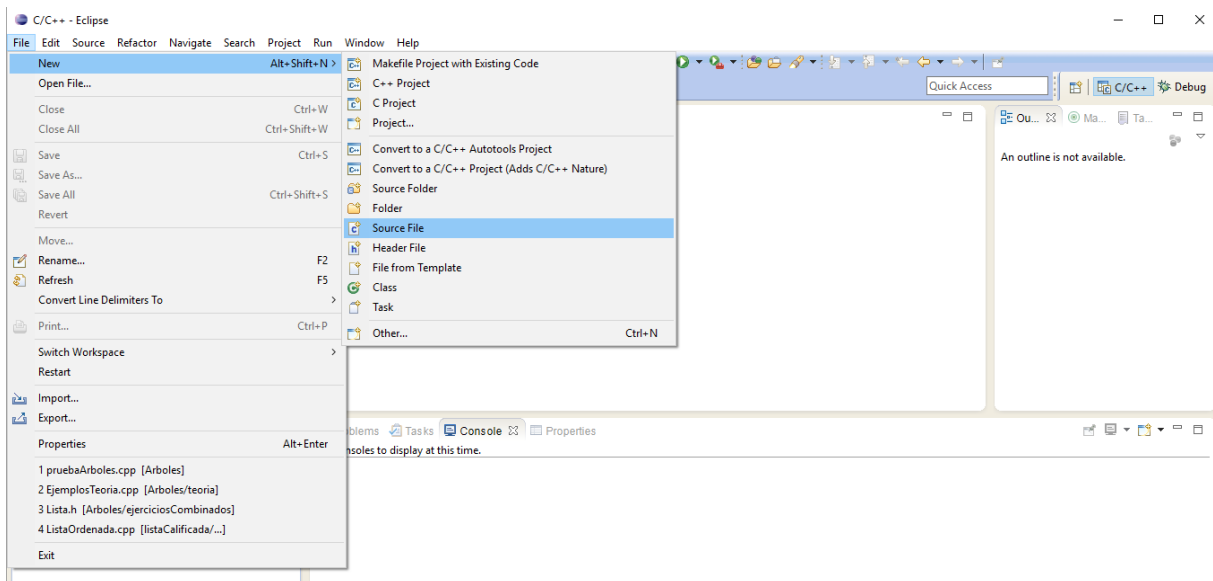
2. Construcción y prueba de un proyecto.

2.1. Cree (en *TallerED*) un proyecto C++ de nombre "Actividad1".





2.2. Incluya en el proyecto el código fuente *hola.cpp*.



2.3. Pulse guardar y observe el nuevo contenido en el *Project Explorer*.

2.4. Pruebe el proyecto.

Se requieren dos etapas:

- Construir el "proyecto"**. (Botón derecho sobre el proyecto -> *Build Project*, o icono "martillo"). El resultado será un fichero de extensión *.exe*
- Ejecutar el programa**. (Botón derecho sobre el proyecto -> *Run As -> Local C/C++ Application*, o icono "run").

3. Ejercicio propuesto.

3.1. Construir y probar un proyecto en C++ que, dado un número recogido desde el teclado, genere un listado de sus divisores (sean o no primos).

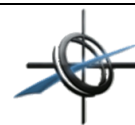
Plantilla:

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

void divisor (int n);

int main(){
    int i;
    cout << "Introduzca un valor entero: " << endl;
    cin >> i;
    cout << "El valor introducido es " << i << " y sus divisores son:";
    divisor (i);
    cout << endl;
    return 0;
}

void divisor (int n){
    //COMPLETAR
}
```



4. Manejo del depurador (*debugger*).

El siguiente código en C++ calcula el factorial de un número introducido por pantalla.

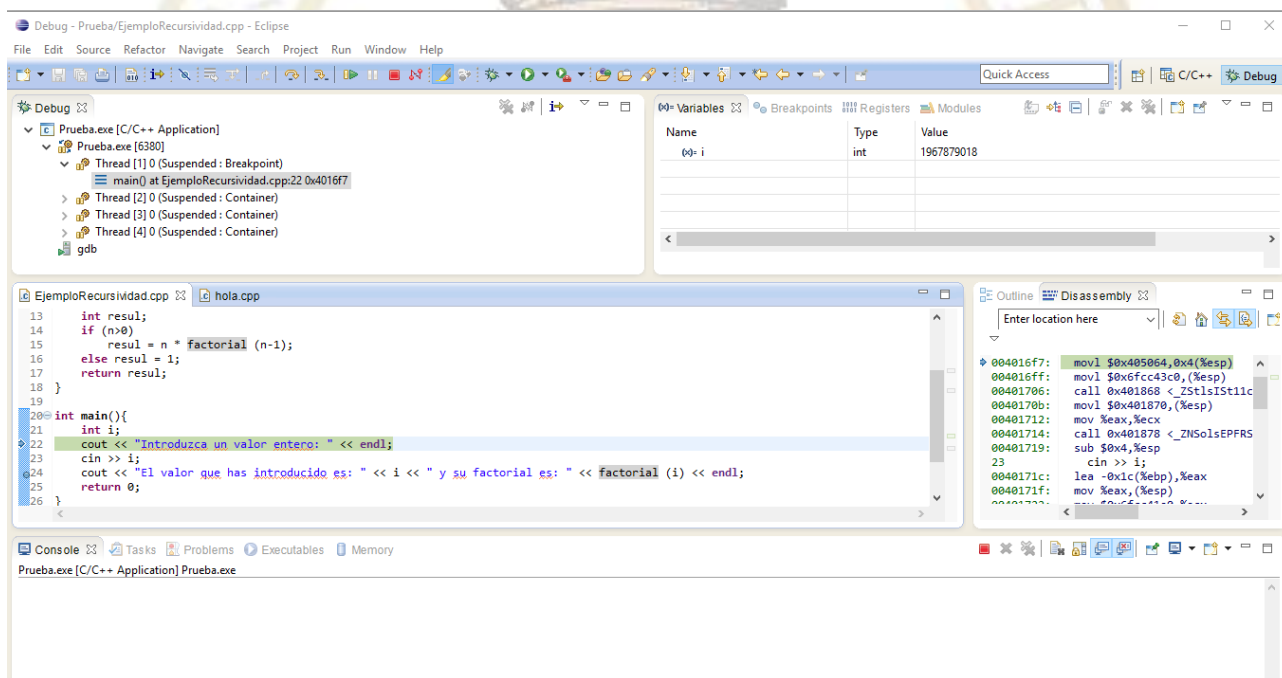
```
#include <iostream>
using namespace std;

int factorial (int n){
    int resul = 0;
    if (n > 0)
        resul = n * factorial (n-1);
    else resul = 1;
    return resul;
}

int main() {
    int i;
    cout << " Introduzca un valor entero: " << endl;
    cin >> i;
    cout << "El valor que has introducido es: " << i << endl;
    cout << "El valor de " << i << " es: " << factorial (i) << endl;
    return 0;
}
```

4.1. Construya el proyecto.

4.2. Defina un punto de ruptura (*breakpoint*) en la línea en que se realiza la llamada recursiva: "*factorial(i)*" usando botón derecho en la línea de código y "*Toggle Breakpoint*". Ejecute en modo *Debug*. La pantalla deberá mostrar el aspecto de la figura.



4.3. Familiarícese con la paleta de herramientas para ejecutar el *debug* paso a paso, introducirse en la ejecución de una función, continuar hasta el siguiente *breakpoint*, o parar la ejecución.

4.4. Ejecute el programa paso a paso (F5 o). Responda "3" al dato solicitado para calcular el factorial. Observe la pantalla y localice:

- La pila de *threads* (o instancias) de ejecución
- El instante (sentencia) actual de ejecución
- La zona de variables para inspeccionar sus valores.
- Observe y trate de entender lo que sucede.



The screenshot shows the Eclipse IDE in debug mode. The main window displays the source code of a recursive factorial function. The current execution point is at line 14, where the condition `if (n > 0)` is being evaluated. The Variables window on the right shows the current state of the function: `n` is 3 and `resul` is 1875655168. The Disassembly window on the right shows the assembly code for the `factorial` function, with the instruction `cmp $0x0,0x8(%ebp)` highlighted, corresponding to the `if (n > 0)` condition. The Console window at the bottom shows the output of the program: `Introduzca un valor entero: 3`.

5. Operaciones finales.

- Mediante el Explorador de Windows analice la carpeta que contiene su *workbench* (**TallerED**).
- Guarde su *workbench* en una ubicación externa (*pen drive*, dispositivo móvil, *cloud*...).
- Salga de Eclipse.
- Vuelva a iniciar Eclipse y recupere su *workbench* desde la ubicación externa utilizada en el paso b). **Deberá encontrar las cosas tal como las dejó.**

Asegúrese de tener su *workbench* (TallerED) disponible para posteriores sesiones prácticas y en cualquier equipo que tenga instalado Eclipse.