

# Retro-Memo Test

---

## Contenido

Introducción.....	2
Características principales .....	2
Modos de juego.....	2
Singleplayer.....	2
Multiplayer .....	2
Niveles de dificultad.....	2
Sistema de pantallas .....	2
Gestión de recursos .....	3
Sistema de tablero .....	3
Sistema de entrada de usuario .....	4
Sistema de puntos .....	4
Interfaz gráfica .....	4
Instalación .....	4
IDE Code::Blocks .....	5
Descarga e instalación.....	5
Creación de un proyecto .....	6
Integración de archivos .....	10
SDL.....	12
Documentación oficial.....	18

## Introducción

Retro-Memo test es un videojuego de memoria desarrollado en lenguaje C utilizando la biblioteca multimedia SDL2. El objetivo principal del mismo es desafiar la memoria del jugador a través de la identificación de pares de cartas en un tablero dinámico configurable.

El proyecto fue diseñado bajo una arquitectura modular basada en pantallas, con separación de responsabilidades entre lógica, renderizado, entrada de usuario y gestión de recursos.

## Características principales

### Modos de juego

#### Singleplayer

- Juego individual.
- Registro de intentos.
- Historial del jugador.

#### Multiplayer

- Hasta 2 (dos) jugadores.
- Sistema de turnos.
- Determinación del ganador.

## Niveles de dificultad

El juego permite seleccionar diferentes configuraciones de tablero:

- Dificultad baja: tablero de 3x4.
- Dificultad media: tablero de 4x4.
- Dificultad alta: tablero de 4x5.

Cada dificultad ajusta automáticamente la cantidad de cartas, la distribución de estas dentro del tablero y su inicialización de manera dinámica.

## Sistema de pantallas

El juego se organiza en diferentes estados de aplicación:

- Pantalla principal.
- Pantalla de configuración.
- Ingreso de jugador (nombre del o los jugadores).
- Selección de set de cartas.

- Selección de dificultad.
- Pantalla de juego.
- Pantalla de resultados.

Cada pantalla implementa su lógica individual de inicialización, update (manejo de input), renderización y liberación de recursos, lo cual evita el acoplamiento y pretende mejorar la mantenibilidad.

## Gestión de recursos

El proyecto implementa:

- Carga y destrucción centralizada de recursos reutilizables (imágenes, fuentes y sonidos).
- Creación y destrucción controlada de texturas.
- Liberación de memoria dinámica.
- Separación de carpetas por responsabilidad:
  - Img (Imágenes)
  - Fnt (Fuentes)
  - Snd (Sonidos)
  - Doc (Documentación)

## Configuración

A través de la pantalla principal, se podrá acceder a un menú de configuración desde el cual se podrá modificar el volumen general del juego (el cual perdurará hasta cambiarlo nuevamente) y la opción de establecer nombres por default al momento de iniciar una partida.

Esto último permite o no iniciar un nuevo juego sin completar el nombre del jugador, siendo estos “Player 1” y “Player 2” los predeterminados en caso de no ingresar de ningún nombre.

## Sistema de tablero

El tablero se genera dinámicamente según la dificultad elegida, utilizando un array para las cartas en juego, el cual es mezclado aleatoriamente a fin de mantener un nivel de jugabilidad aceptable.

Por otro lado, se realiza la necesaria validación de coincidencias y el respectivo control de estado por carta (oculta, revelada, emparejada) durante el tiempo de juego, así como también su correcta destrucción en los casos necesarios.

## Sistema de entrada de usuario

El sistema de entrada de usuario está principalmente implementado para recibir todas las acciones necesarias del usuario en este contexto, siendo estas el manejo de eventos dados por mouse y la entrada del texto en los momentos necesarios.

Se tiene en cuenta la detección de colisión a través de coordenadas X e Y, así como también la gestión de los estados de los botones (normal, hover, pressed).

## Sistema de puntos

El juego contempla un sistema de puntos tanto en modo singleplayer como multiplayer, el cual consta de puntos base por adivinar, racha de aciertos y penalidades por fallos.

Actualmente, cada 2 (dos) cartas seleccionadas, se considera un movimiento hecho por jugador. Si ese movimiento resulta en un acierto (es decir, si ambas cartas son iguales), el jugador recibirá 100 puntos en su puntaje total. Si acertó 2 o más movimientos seguidos, los puntos a recibir obtendrán un pequeño bonus. En caso de fallar el movimiento (es decir, si ambas cartas son diferentes), el puntaje total se verá ligeramente decrementado.

Este sistema de puntos ayuda a incluir, a su vez, un ranking de los 5 mejores jugadores.

Este ranking será histórico, tomando en cuenta en ambos modos de juego, y podrá ser visualizado a través de la pantalla principal.

Tanto para el modo singleplayer como multiplayer, al final de la partida se podrán visualizar los resultados del juego, mostrando la cantidad de movimientos y los puntos totales obtenidos por cada jugador.

## Interfaz gráfica

Se hace uso de texturas para todo elemento representado en pantalla. Todos los estados visuales que puedan recibir los elementos de manera individual se encuentran correctamente diferenciados.

A fin de cumplir con los requisitos técnicos mencionados, se destaca que:

- Los sets de cartas están diseñados a través de imágenes PNG, con un tamaño de 256x144px.
- La pantalla de juego se encuentra establecida en una resolución 1366x768px.

## Instalación

Para la correcta jugabilidad del juego en esta etapa, se deberá tener en cuenta que será necesario compilar correctamente el proyecto de manera individual y autónoma. Es por

este motivo que, dentro de esta sección, se desglosarán todos los pasos necesarios para su ejecución.

## IDE Code::Blocks

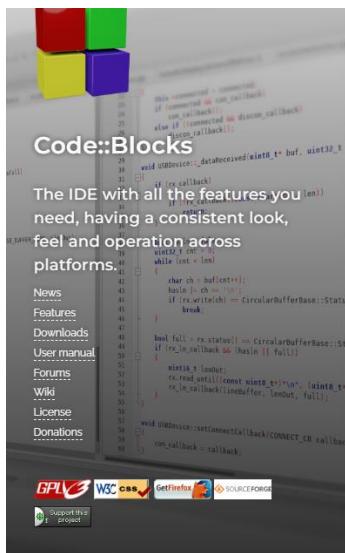
El proyecto fue desarrollado, compilado y ejecutado utilizando el IDE Code::Blocks.

Podrá encontrar información sobre el mismo a través de su página oficial:

<https://www.codeblocks.org/>

## Descarga e instalación

Dentro de la página oficial, podrá encontrar los archivos de descarga en la sección “Downloads”.



Code::Blocks / Downloads

## Downloads

There are different ways to download and install Code::Blocks on your computer:

- Download the binary release**  
This is the easy way for installing Code::Blocks. Download the setup file, run it on your computer and Code::Blocks will be installed, ready for you to work with it. Can't get any easier than that!
- Download a nightly build**  
There are also more recent so-called nightly builds available in the forums. Please note that we consider nightly builds to be stable; usually, unless stated otherwise.
- Other distributions** usually follow provided by the community (big "Thank you!" for that). If you want to provide some, make sure to announce in the forums such that we can put it on the official C-B homepage.
- Download the source code**  
If you feel comfortable building applications from source, then this is the recommend way to download Code::Blocks. Downloading the source code and building it yourself puts you in great control and also makes it easier for you to update to newer versions or, even better, create patches for bugs you may find and contributing them back to the community so everyone benefits.
- Retrieve source code from SVN**  
This option is the most flexible of all but requires a little bit more work to setup. It gives you that much more flexibility though because you get access to bug-fixing we do at the time we do it. No need to wait for the next stable release to benefit from bug-fixes!  
Besides Code::Blocks itself, you can compile extra plugins from contributors to extend its functionality.

Diríjase a la sección “Download the binary release”. Seleccione dentro de la pantalla el sistema operativo que posea.

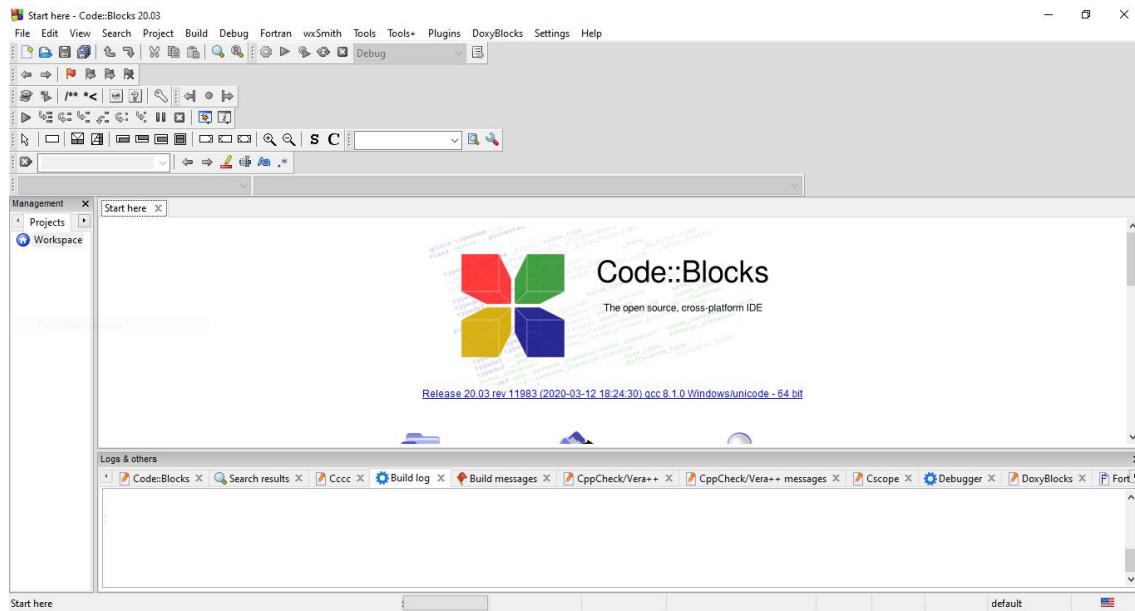
*Nota: Si utiliza un entorno Windows, podrá descargar el IDE junto al compilador mingw; seleccione esa opción de descarga para incluirlo dentro de la instalación. Si descarga el IDE sin compilador, o bien se encuentra en otro entorno, deberá encargarse personalmente de instalar y utilizar el compilador de su preferencia de manera individual.*

Podrá elegir la opción instalable tanto como la portable en este caso, dependiendo únicamente de sus preferencias.

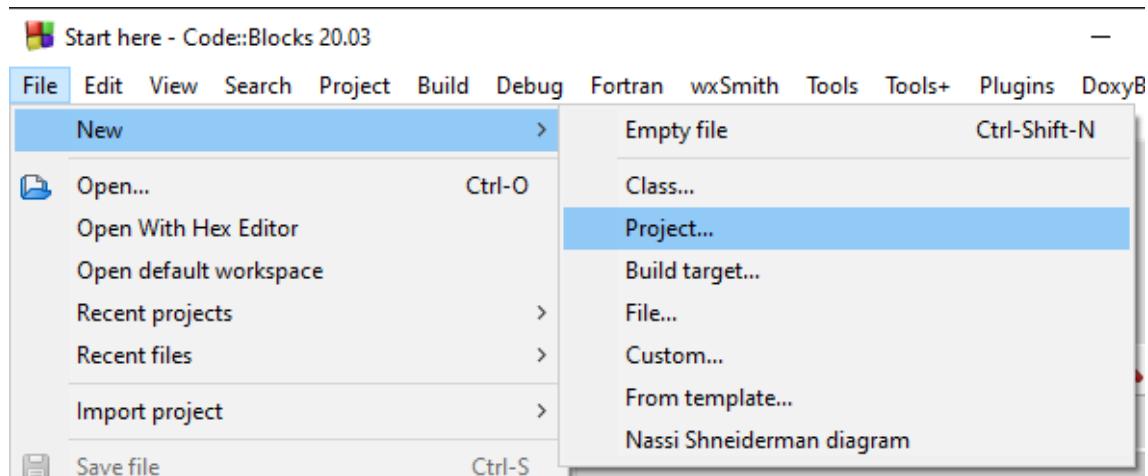
En caso de elegir la opción instalable, siga las instrucciones del asistente de instalación; asegúrese de realizar una instalación Full (“Full: All plugins, all tools”) para evitar problemas en el futuro.

## Creación de un proyecto

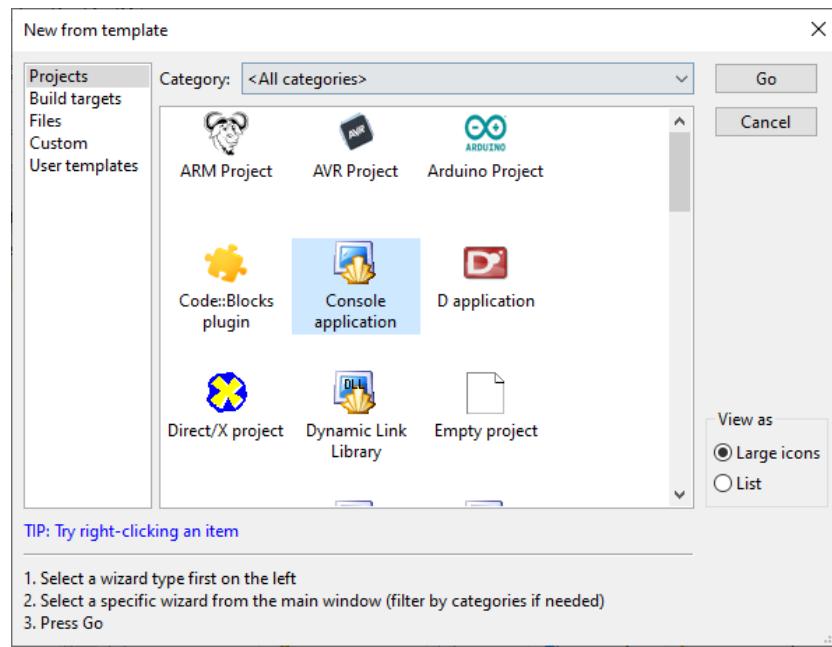
Ejecute el IDE Code::Blocks.



Diríjase a la sección “File”, ubicada en la parte superior izquierda. Luego, hacia “New” y seleccione “Project”.

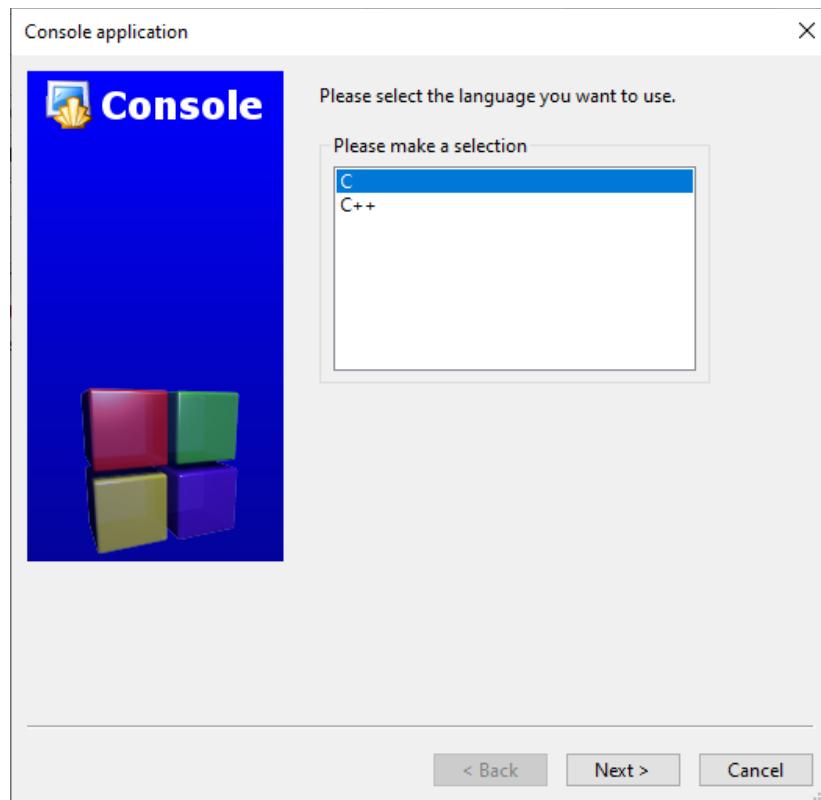


Se abrirá una ventana donde podrá elegir el tipo de proyecto a crear. Seleccione la opción “Console application”.

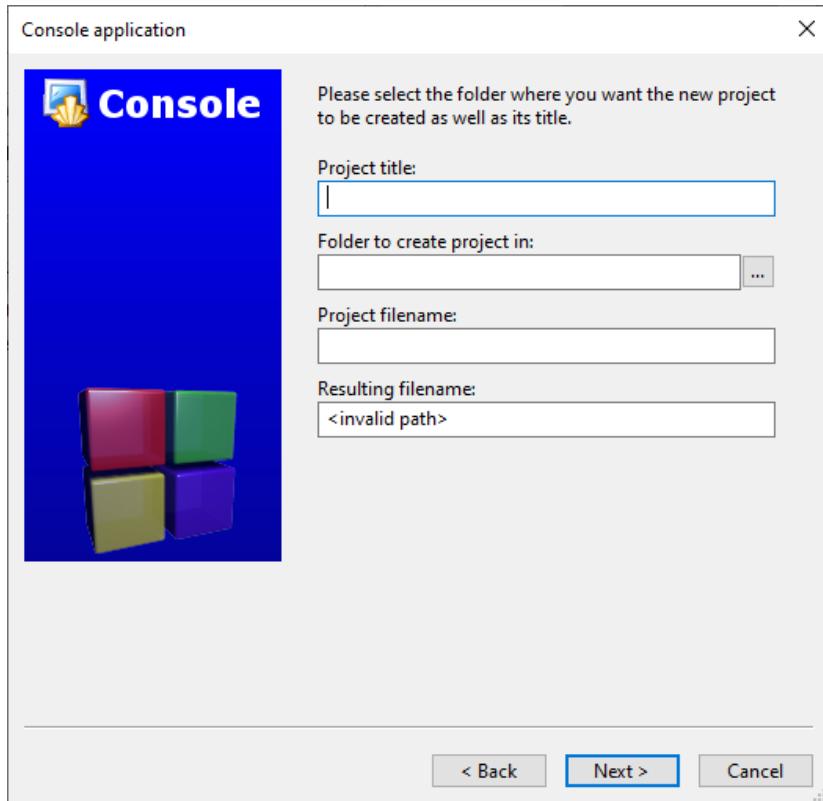


Haga clic en el botón “Go” para continuar.

Seleccione, como lenguaje a utilizar, “C”. Luego, haga clic en “Next”.



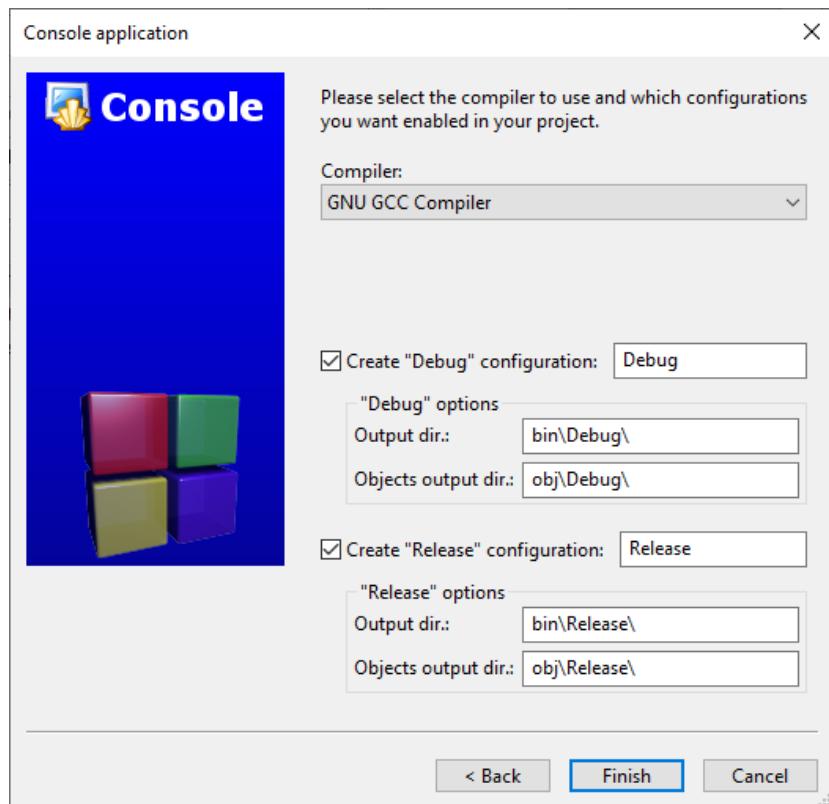
En la siguiente ventana, deberá indicar el nombre del proyecto y su correspondiente ruta.



Primero, ingrese la ruta donde desea crear y guardar el proyecto, cliqueando el botón de la derecha sobre el campo “Folder to create project in”. Luego, ingrese el nombre que desea darle al proyecto sobre el campo “Project title”.

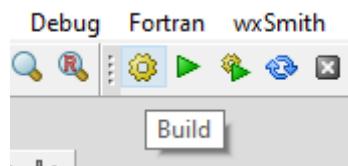
Cuando haya finalizado, haga clic en “Next” para continuar.

En la siguiente ventana, asegúrese de tener habilitadas las opciones “Debug” y “Release”.



Haga clic en “Finish” para crear el proyecto.

Compile el proyecto recién creado para asegurarse que todo haya sido correctamente configurado a través del botón “Build”.



La compilación del proyecto deberá ser exitosa para poder continuar. Asegúrese de esto mirando detenidamente la sección “Build log”.

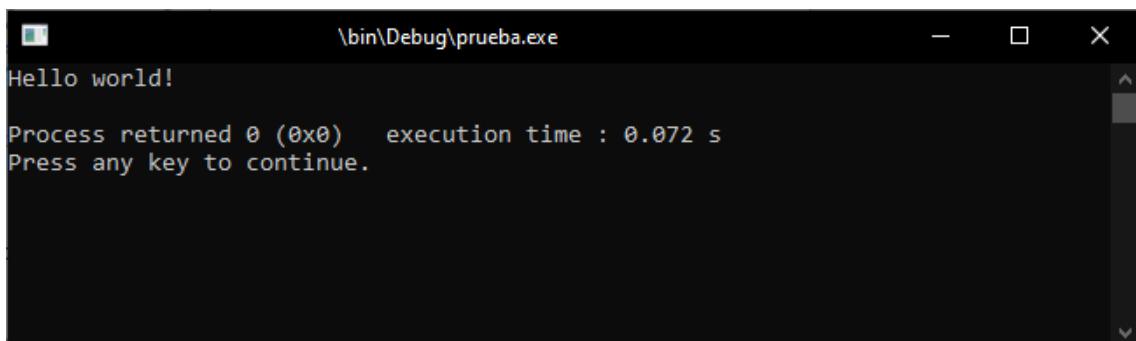
```
Logs & others
Code::Blocks X Search results X Ccc X Build log X Build messages X
gcc.exe -Wall -g -pedantic-errors -c          Desktop\prueba\main.c -o
obj\Debug\main.o
gcc.exe -o bin\Debug\prueba.exe obj\Debug\main.o
Output file is bin\Debug\prueba.exe with size 53.93 KB
Process terminated with status 0 (0 minute(s), 0 second(s))
0 error(s), 0 warning(s) (0 minute(s), 0 second(s))
```

Todo proyecto que usted compile deberá tener, dentro de las posibilidades, 0 (cero) errores y 0 (cero) advertencias.

Ejecute el proyecto, si así lo desea, a través del botón “Run”.



Si Code::Blocks fue correctamente instalado, al momento de compilar y ejecutar el código main por defecto, verá la siguiente pantalla.



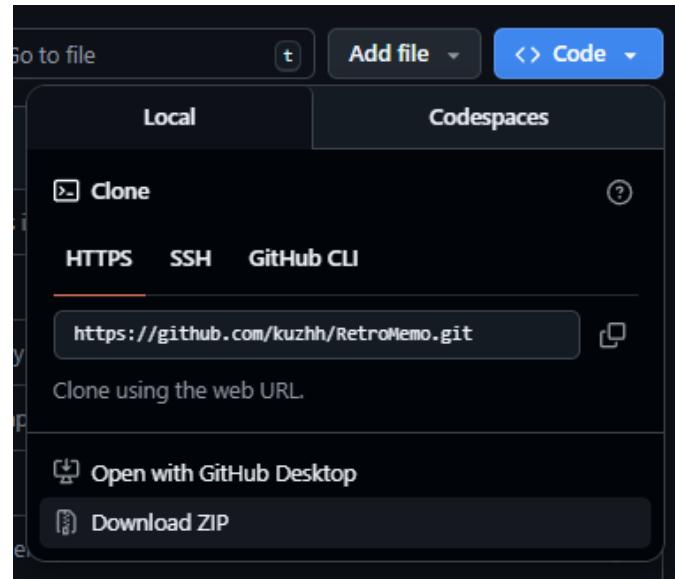
Termine la ejecución del programa presionando cualquier tecla dentro de esta pantalla.

Evite al máximo posible cerrar la ventana desde la “X”.

### Integración de archivos

Una vez listo y configurado el entorno, ya habiéndonos asegurado de que los programas se ejecutan correctamente, descargue los archivos correspondientes al juego Retro-Memo Test desde el siguiente enlace: <https://github.com/kuzhh/RetroMemo>

Podrá ver diferentes opciones de descarga haciendo clic en el botón “Code”.

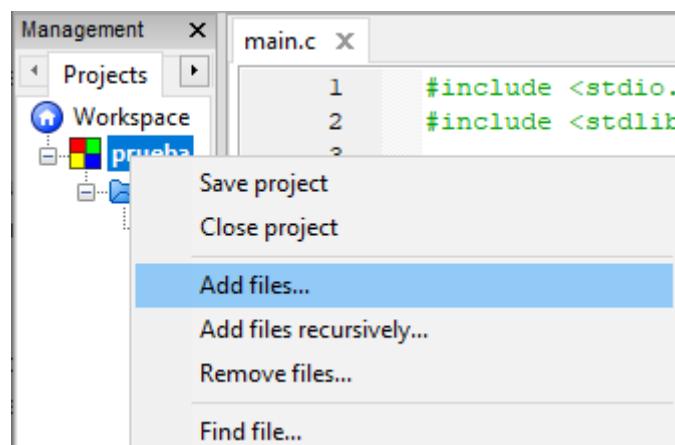


Utilice la más cómoda para usted.

Si realiza la descarga del ZIP, descomprima el archivo dentro de la carpeta del proyecto que acaba de crear.

Si desea clonar el repositorio con Git o GitHub Desktop, quedará en sus propios medios establecer las rutas adecuadas para la integración de los archivos al proyecto creado.

Una vez haya movido todos los archivos descargados a la carpeta del proyecto, vuelva a entrar al IDE. Sobre la sección izquierda, verá el nombre del proyecto; haga clic derecho sobre él y seleccione la opción “Add files”.



Seleccione todos los archivos descargados. Cuando finalice, haga clic en “Abrir”.

Verá, con esto, que se han integrado todos los archivos a su proyecto. Si compila el programa en este momento, no podrá ejecutarlo; prepare los archivos correspondientes a la librería SDL antes de continuar.

## SDL

A fin de poder renderizar el programa correctamente, le solicitamos que siga las instrucciones en el orden establecido. Esta biblioteca resulta muy delicada de manejar, y cualquier error al momento de su integración podrá derivar en errores de compilación (y dolores de cabeza).

Deberá descargar la carpeta principal que contiene la biblioteca SDL, así como todas las extensiones necesarias nombradas, a fin de incluir correctamente todos los recursos utilizados.

- SDL2

Podrá realizar la descarga desde el siguiente enlace: <https://github.com/libsdl-org/SDL/releases/download/release-2.32.8/SDL2-devel-2.32.8-mingw.zip>

- SDL2 mixer

Podrá realizar la descarga desde el siguiente enlace: [https://github.com/libsdl-org/SDL\\_mixer/releases/download/release-2.8.1/SDL2\\_mixer-devel-2.8.1-mingw.zip](https://github.com/libsdl-org/SDL_mixer/releases/download/release-2.8.1/SDL2_mixer-devel-2.8.1-mingw.zip)

- SDL2 image

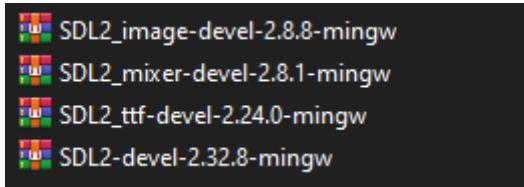
Podrá realizar la descarga desde el siguiente enlace: [https://github.com/libsdl-org/SDL\\_image/releases/download/release-2.8.8/SDL2\\_image-devel-2.8.8-mingw.zip](https://github.com/libsdl-org/SDL_image/releases/download/release-2.8.8/SDL2_image-devel-2.8.8-mingw.zip)

- SDL2 ttf

Podrá realizar la descarga desde el siguiente enlace: [https://github.com/libsdl-org/SDL\\_ttf/releases/download/release-2.24.0/SDL2\\_ttf-devel-2.24.0-mingw.zip](https://github.com/libsdl-org/SDL_ttf/releases/download/release-2.24.0/SDL2_ttf-devel-2.24.0-mingw.zip)

*Nota: Tenga en cuenta que estos enlaces son únicamente de descarga. SDL2 ya fue reemplazado por su nueva versión (SDL3), por lo que podrá ser difícil para usted encontrar la página GitHub original.*

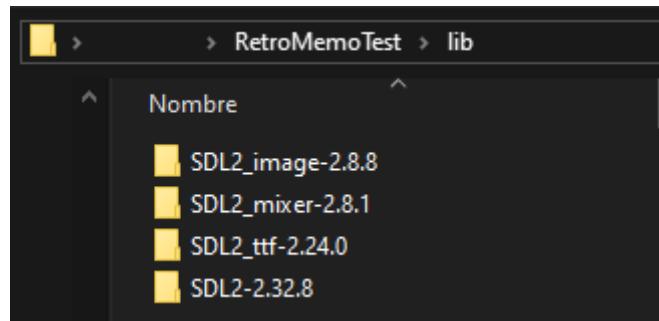
Estos enlaces habilitarán la descarga de 4 (cuatro) archivos .zip, nombrados: “*SDL2-devel-2.32.8-mingw*”, “*SDL2\_mixer-devel-2.8.1-mingw*”, “*SDL2\_image-devel-2.8.8-mingw*” y “*SDL2\_ttf-devel-2.24.0-mingw*”.



Dentro de la carpeta de su proyecto, cree una nueva carpeta con el nombre “*lib*”.

Una vez hecho esto, descomprima los 4 (cuatro) archivos .zip dentro de la misma; obtendrá 4 (cuatro) carpetas, nombradas: “*SDL2-2.32.8*”, “*SDL2\_mixer-2.8.1*”, “*SDL2\_image-2.8.8*” y “*SDL2\_ttf-2.24.0*”.

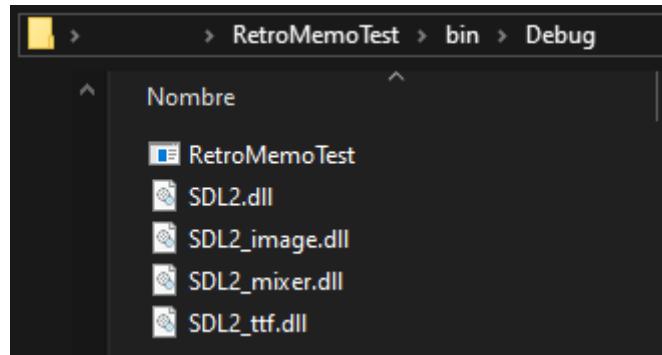
Puede eliminar los archivos .zip o moverlos de lugar acorde a su conveniencia.



Dentro de cada una de las carpetas, podrá encontrar las bibliotecas dinámicas correspondientes a cada módulo: “*SDL2.dll*”, “*SDL2\_image.dll*”, “*SDL2\_mixer.dll*” y “*SDL2\_ttf.dll*”.

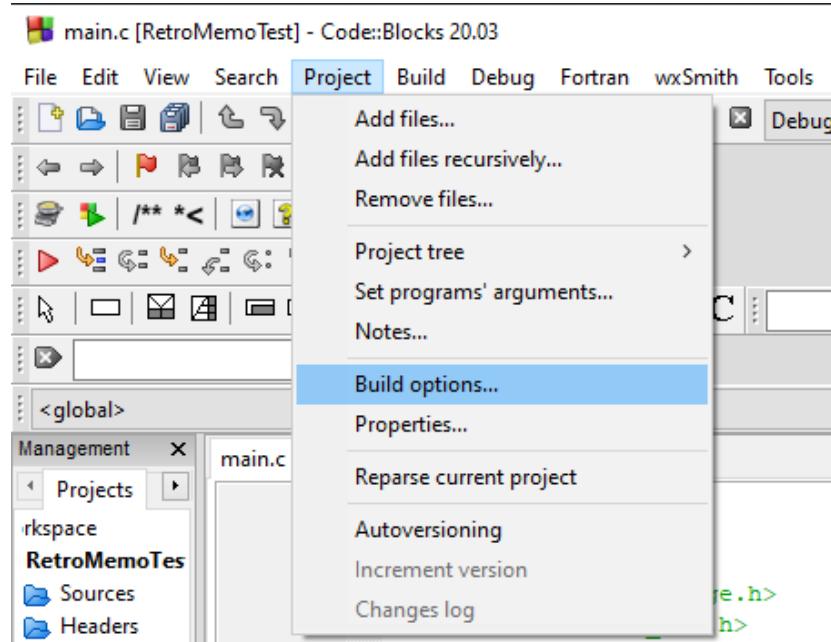
Dichas bibliotecas, para entornos Windows de 64 bits, se encuentran en la ruta: `../x86_64-w64-mingw32/bin` dentro de cada carpeta.

Copie todos los archivos .dll mencionados, y péguelos dentro de su proyecto en la siguiente ruta: `../bin/Debug`. Asegúrese que el ejecutable (archivo .exe) del proyecto se encuentre junto a ellos.



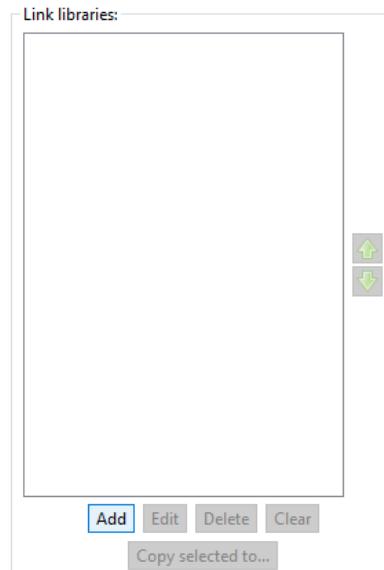
Ejecute el IDE Code::Blocks y abra el proyecto anteriormente creado.

Una vez dentro del mismo, en la barra de herramientas superior, podrá encontrar la pestaña “Project”. Haga clic sobre ella, y luego sobre la opción “Build options...”.

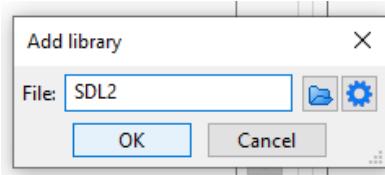


Dentro de dicha ventana, diríjase a la pestaña “Linker settings”. Ahí, usted deberá incluir la librería SDL2 y sus extensiones descargadas de la siguiente manera:

1. Dentro de la sección “Link libraries”, presione el botón “Add”.



Agregue la librería “SDL2” dentro de la ventana emergente, y luego presione “OK”.



Podrá ver, en la nombrada sección, que acabamos de agregar la librería a utilizar.

2. Dentro de la sección “Other link options”, ingrese:

- -lmingw32
- -lSDL2main
- -lSDL2
- -lSDL2\_mixer
- -lSDL2\_ttf
- -lSDL2\_image



Asegúrese de que los primeros 3 (tres) parámetros siempre se encuentren en el orden correcto, a fin de evitar errores de lindeo en las bibliotecas agregadas.

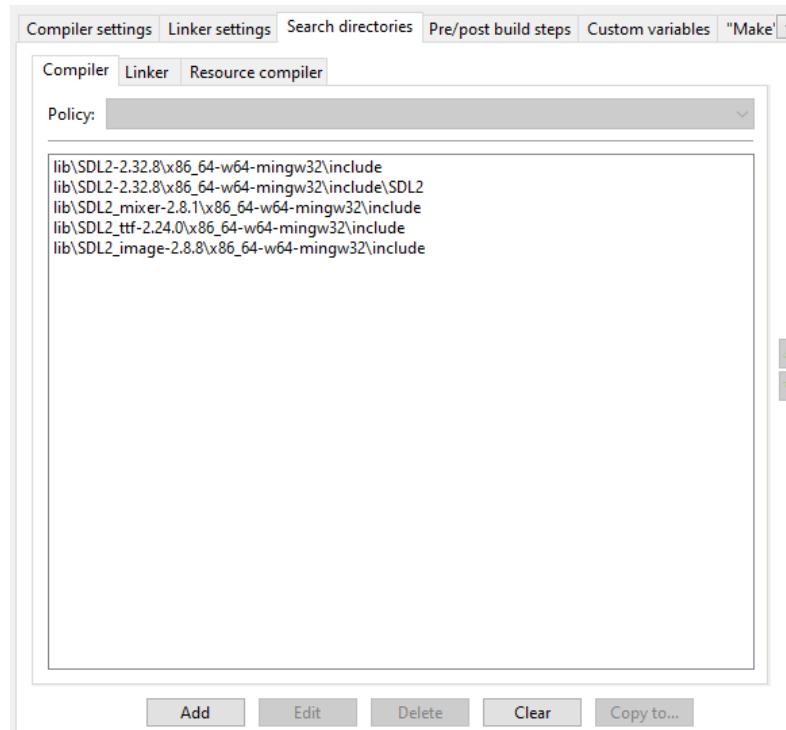
*Nota: Resulta importante a destacar, ya que no puede lograr diferenciarse correctamente en algunas fuentes, que la primera letra de cada parámetro es una letra 'L' minúscula. Asegúrese de escribir correctamente cada parámetro.*

3. Sobre la misma ventana, diríjase a la pestaña “Search directories”. Dentro de esta, en la pestaña “Compiler”, deberá ingresar cada ruta destinada a la carpeta “include” de las bibliotecas, utilizando el botón “Add” en cada oportunidad.

*Nota: Tenga en cuenta que, únicamente para la carpeta “SDL2-2.32.8”, deberá incluir tanto la carpeta “include” como “include/SDL2”.*

Las rutas de acceso deberían resultar ser las siguientes:

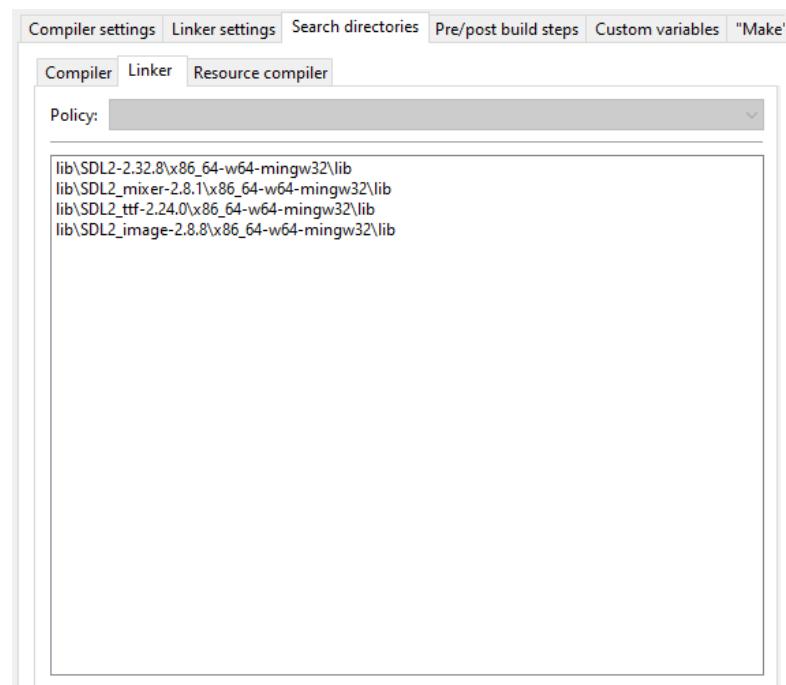
- *lib\SDL2-2.32.8\x86\_64-w64-mingw32\include*
- *lib\SDL2-2.32.8\x86\_64-w64-mingw32\include\SDL2*
- *lib\SDL2\_mixer-2.8.1\x86\_64-w64-mingw32\include*
- *lib\SDL2\_ttf-2.24.0\x86\_64-w64-mingw32\include*
- *lib\SDL2\_image-2.8.8\x86\_64-w64-mingw32\include*



4. Por último, diríjase a la pestaña “Linker”. Dentro de esta, deberá ingresar cada ruta destinada a la carpeta “lib” de las bibliotecas, utilizando el botón “Add” en cada oportunidad.

Las rutas de acceso deberían resultar ser las siguientes:

- *lib\SDL2-2.32.8\x86\_64-w64-mingw32\lib*
- *lib\SDL2\_mixer-2.8.1\x86\_64-w64-mingw32\lib*
- *lib\SDL2\_ttf-2.24.0\x86\_64-w64-mingw32\lib*
- *lib\SDL2\_image-2.8.8\x86\_64-w64-mingw32\lib*



***Por favor, absténgase de utilizar las rutas brindadas en la guía para la localización de las carpetas; las mismas dependen exclusivamente de su entorno y organización, por lo cual podrían variar.***

***Agregue manualmente cada ruta examinando su propio explorador de archivos en cada caso.***

Una vez realizados los cambios, haga clic en “OK” para guardarlos. Ya se encuentra en posición de compilar y ejecutar el programa correctamente.

## **Documentación oficial**

Podrá ingresar a la web oficial de SDL2 para hacer uso de su extendida documentación oficial, a fin de aprender a utilizar la librería si así lo desea.

Podrá encontrar la documentación sobre la librería SDL2 en este enlace:

<https://wiki.libsdl.org/SDL2/FrontPage>