

Codelite et compilateur C Mingw sous Windows : utilisation de SDL2

Auteur : P. Gaucher – Département Informatique Polytech Tours.

Version 1.0 décembre 2023

1. Préalable

Nous supposons que l'IDE Codelite est installé et correctement paramétré pour permettre la compilation, l'édition de liens et l'exécution de l'exécutable généré en mode console.

Le compilateur C installé est mingw-64, version 64 bits.

Nous supposons également que MSYS2 est installé et que l'installation du compilateur C s'est effectuée à l'aide de cet outil.

2. Librairie SDL

La SDL - **S**imple **D**irect**M**edia **L**ayer est une bibliothèque multimédia qui permet un accès bas-niveau à l'audio, au clavier, à la souris, au joystick, aux graphiques, ... Elle offre la possibilité d'afficher des fenêtres, d'afficher des images, de jouer des sons, de gérer le clavier... Il s'agit d'une bibliothèque libre, multiplateforme. Elle est écrite en C. Elle est compatible avec l'écriture d'application en langage C.

La version stable la plus récente est SDL2 (SDL 2.28.5¹).

Le point d'entrée documentaire est <https://www.libsdl.org/>.

L'objectif de ce tutorial est de fournir les éléments permettant d'interfacer cette librairie au sein de l'IDE CodeLite.

2.1. Installation de SDL2

Nous utilisons l'utilitaire MSYS2 pour installer l'ensemble des fichiers de la librairie. La page de référence d'installation est https://packages.msys2.org/package/mingw-w64-x86_64-SDL2.

Une fois l'utilitaire *mingw64.exe* lancé, la commande :

```
pacman -S mingw-w64-x86_64-SDL2
```

Cela permet d'effectuer l'installation des ressources de la SDL2.



Figure 1 : Commande pacman d'installation de la SDL2

¹ Version la plus récente disponible au 01.12.2023

```
M ~
# pacman -S mingw-w64-x86_64-SDL2
resolving dependencies...
looking for conflicting packages...

Packages (3) mingw-w64-x86_64-vulkan-headers-1.3.268-1  mingw-w64-x86_64-vulkan-loader-1.3.268-1
              mingw-w64-x86_64-SDL2-2.28.5-1

Total Download Size:    2.70 MiB
Total Installed Size:  33.92 MiB

:: Proceed with installation? [Y/n] y
```

Figure 2 : Installation de la SDL2 (1)

```
M ~
# pacman -S mingw-w64-x86_64-SDL2
resolving dependencies...
looking for conflicting packages...

Packages (3) mingw-w64-x86_64-vulkan-headers-1.3.268-1  mingw-w64-x86_64-vulkan-loader-1.3.268-1
              mingw-w64-x86_64-SDL2-2.28.5-1

Total Download Size:    2.70 MiB
Total Installed Size:  33.92 MiB

:: Proceed with installation? [Y/n] y
:: Retrieving packages...
mingw-w64-x86_64-SDL2-2.28... 1443.6 KiB   395 KiB/s  00:04 [#####] 100%
mingw-w64-x86_64-vulkan-he... 1160.2 KiB  313 KiB/s  00:04 [#####] 100%
mingw-w64-x86_64-vulkan-lo...  160.2 KiB   25.2 KiB/s  00:06 [#####] 100%
Total (3/3)                  2.7 MiB   431 KiB/s  00:06 [#####] 100%
(3/3) checking keys in keyring [#####] 100%
(3/3) checking package integrity [#####] 100%
(3/3) loading package files [#####] 100%
(3/3) checking for file conflicts [#####] 100%
(3/3) checking available disk space [#####] 100%
:: Processing package changes...
(1/3) installing mingw-w64-x86_64-vulkan-headers [#####] 100%
(2/3) installing mingw-w64-x86_64-vulkan-loader [#####] 100%
(3/3) installing mingw-w64-x86_64-SDL2 [#####] 100%

M ~
# |
```

Figure 3 : Installation de la SDL2 (fin)

Les principaux éléments à retenir sont :

- C:\msys64\mingw64\bin\ contient les fichiers suivants :
 - SDL2.dll librairie dynamique
 - sdl2-config fichier contenant les option de compilation et de linkage à appliquer
- C:\msys64\mingw64\include\ contient le fichier d'en-tête SDL.h :
 - ...\SDL2\SDL.h
- C:\msys64\mingw64\lib\ contient les librairies statiques :
 - libSDL2.a
 - libSDL2.dll.a
 - libSDL2main.a

<https://manpages.ubuntu.com/manpages/jammy/man1/sdl2-config.1.html> permet d'accéder à des ressources documentaires concernant le fichier *sdl2-config*.

<https://doc.ubuntu-fr.org/sdl> donne accès à des ressources pour utiliser *sdl2-config* pour faire le *build* d'un projet

2.2. Mettre en œuvre une application C utilisant la SDL

A titre d'illustration, nous allons nous appuyer sur l'exemple de mise en œuvre disponible sur :

<https://www.matsson.com/prog/platformer.c>

- Créer le projet Test_SDL2 dans l'espace de travail.

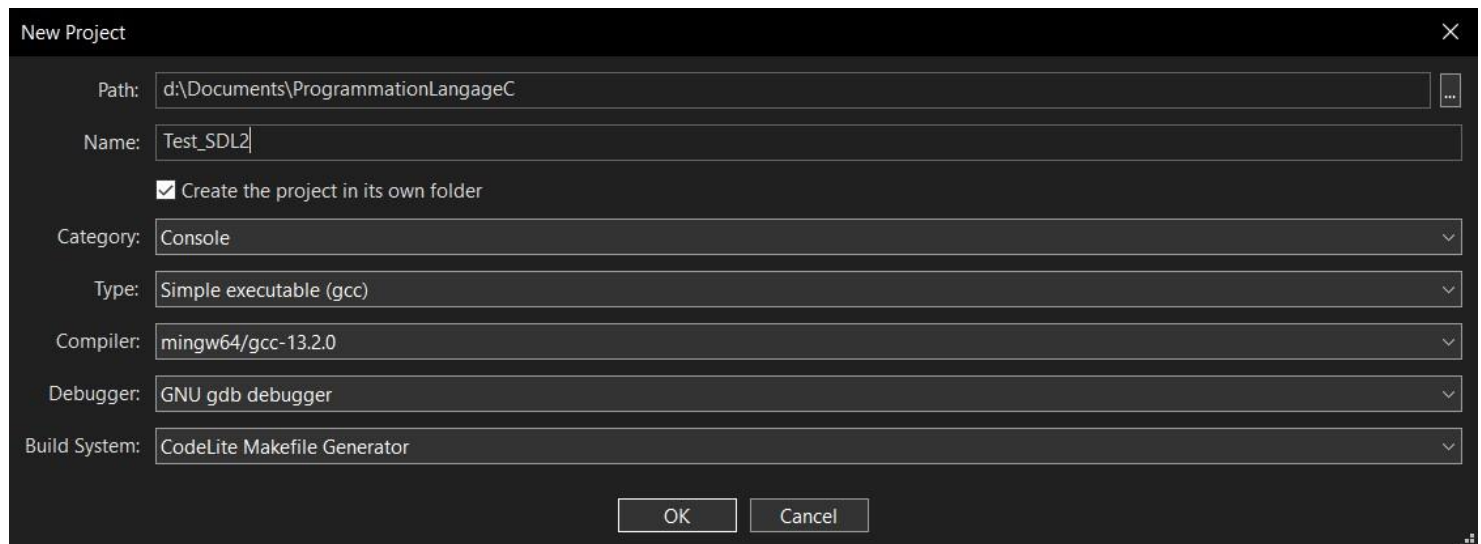


Figure 4 : Création du projet Test_SDL2

- Copier dans *main.c* du projet créé le contenu du code de l'exemple.

```

Test_SDL2\main.c
1  #include <stdio.h>
2  #include <stdbool.h>
3  #include <SDL2/SDL.h>
4  #define WIDTH 640
5  #define HEIGHT 480
6  #define SIZE 200
7  #define SPEED 600
8  #define GRAVITY 60
9  #define FPS 60
10 #define JUMP -1200
11
12 int main(int argc, char* argv[])
13 {
14     /* Initializes the timer, audio, video, joystick,
15     haptic, gamecontroller and events subsystems */
16     if (SDL_Init(SDL_INIT EVERYTHING) != 0)
17     {
18         printf("Error initializing SDL: %s\n", SDL_GetError());
19         return 0;
20     }
21     /* Create a window */
22     SDL_Window* wind = SDL_CreateWindow("Hello Platformer!".

```

Figure 5 : Fichier main.c du projet Test_SDL2

Remarque : Dans un projet impliquant l'utilisation de la librairie SDL2, l'en-tête de la fonction main doit obligatoirement être défini avec les paramètres *argc* et *argv[]*.

- Lancer le build du projet et son exécution.

```

Build Search Replace References Output Terminal
C:/msys64/mingw64/bin/mingw32-make.exe -j12 -e -f Makefile
-----Building project: [ Test_SDL2 - Debug ]-----
mingw32-make[1]: Entering directory 'D:/Documents/ProgrammationLangageC/Test_SDL2'
C:/msys64/mingw64/bin/g++.exe -o ../build-Debug/bin/Test_SDL2.exe @../build-Debug/Test_SDL2/ObjectsList.txt -L.
C:/msys64/mingw64/bin/./lib/gcc/x86_64-w64-mingw32/13.2.0/../../../../x86_64-w64-mingw32/bin/ld.exe: ../build-Debug/Test_SDL2/main.o: in function 'SDL_main':
d:/Documents/ProgrammationLangageC/Test_SDL2/main.c:16:(.text+0x6c): undefined reference to 'SDL_Init'
C:/msys64/mingw64/bin/./lib/gcc/x86_64-w64-mingw32/13.2.0/../../../../x86_64-w64-mingw32/bin/ld.exe: D:/Documents/ProgrammationLangageC/Test_SDL2/main.o:18:(.text+0x73): undefined
C:/msys64/mingw64/bin/./lib/gcc/x86_64-w64-mingw32/13.2.0/../../../../x86_64-w64-mingw32/bin/ld.exe: D:/Documents/ProgrammationLangageC/Test_SDL2/main.o:22:(.text+0xc1): undefined
C:/msys64/mingw64/bin/./lib/gcc/x86_64-w64-mingw32/13.2.0/../../../../x86_64-w64-mingw32/bin/ld.exe: D:/Documents/ProgrammationLangageC/Test_SDL2/main.o:28:(.text+0x1d1): undefined
C:/msys64/mingw64/bin/./lib/gcc/x86_64-w64-mingw32/13.2.0/../../../../x86_64-w64-mingw32/bin/ld.exe: D:/Documents/ProgrammationLangageC/Test_SDL2/main.o:30:(.text+0xe8): undefined
C:/msys64/mingw64/bin/./lib/gcc/x86_64-w64-mingw32/13.2.0/../../../../x86_64-w64-mingw32/bin/ld.exe: D:/Documents/ProgrammationLangageC/Test_SDL2/main.o:34:(.text+0x110): undefined
C:/msys64/mingw64/bin/./lib/gcc/x86_64-w64-mingw32/13.2.0/../../../../x86_64-w64-mingw32/bin/ld.exe: D:/Documents/ProgrammationLangageC/Test_SDL2/main.o:37:(.text+0x120): undefined
C:/msys64/mingw64/bin/./lib/gcc/x86_64-w64-mingw32/13.2.0/../../../../x86_64-w64-mingw32/bin/ld.exe: D:/Documents/ProgrammationLangageC/Test_SDL2/main.o:38:(.text+0x13e): undefined

```

Figure 6 : Build du projet (avec erreur du linker)

On constate alors que des erreurs surviennent lors de l'appel au linker. Il faut alors le paramétrer afin de prendre en compte les librairies de la SDL2, situées dans C:\msys64\mingw64\lib².

Ouvrir la fenêtre de settings du projet Test_SDL2 :

² Répertoire spécifique selon l'installation de la SDL et du compilateur mingWc

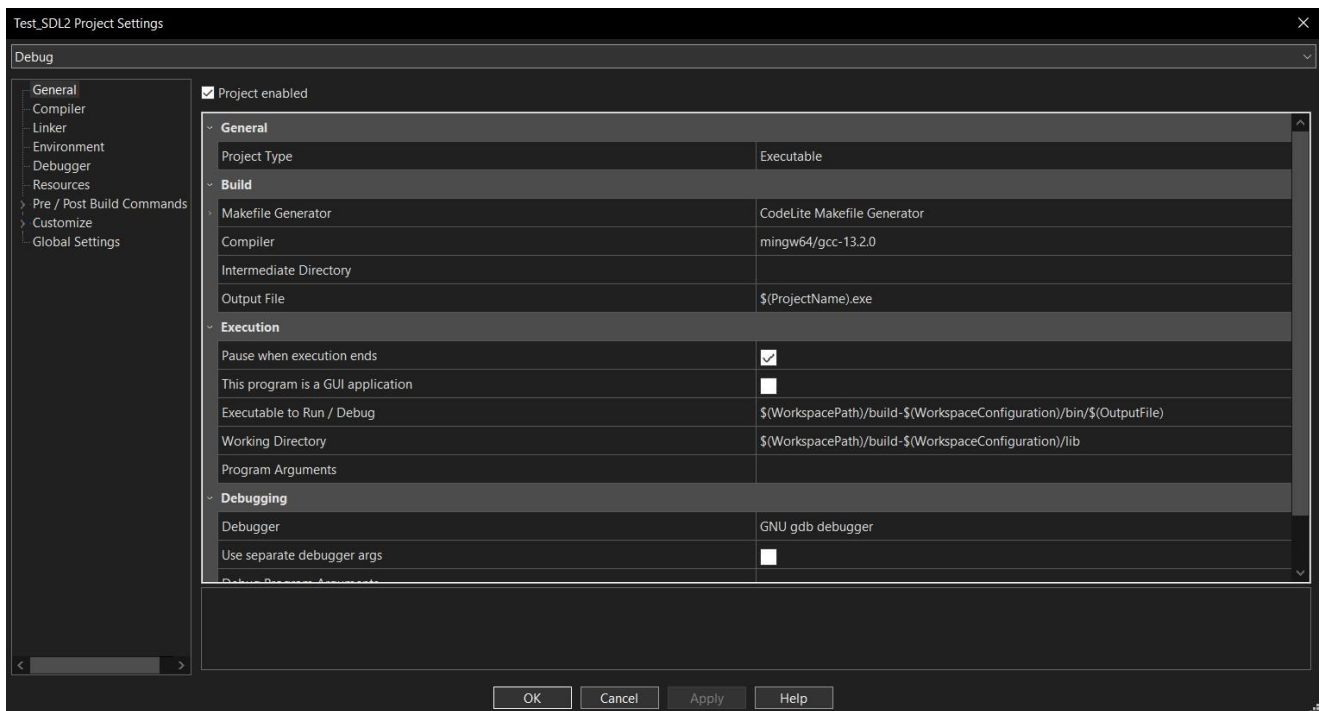


Figure 7 : Fenêtre de paramétrage du projet Test_SDL2

Sélectionner Linker puis Linker options :

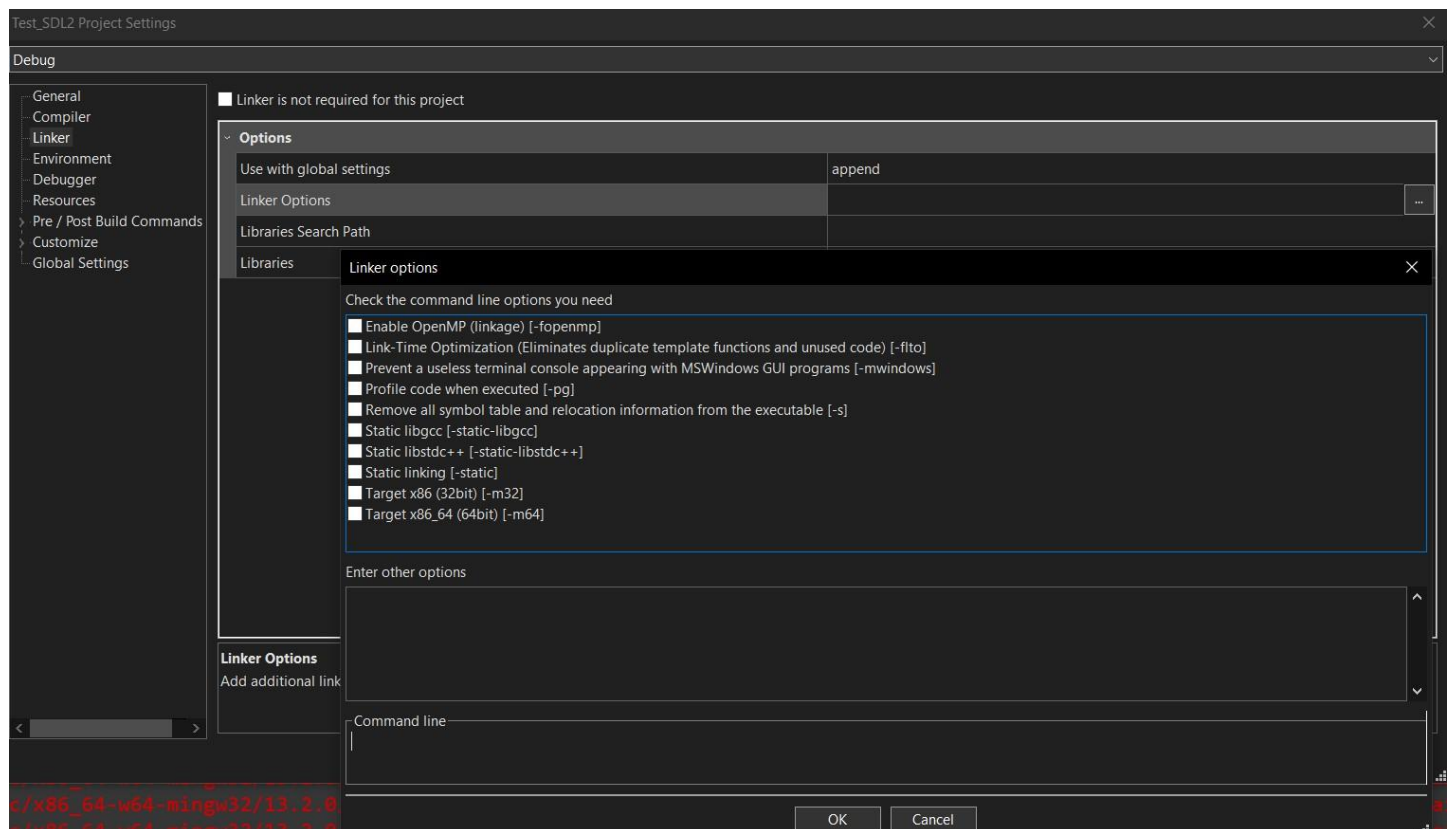


Figure 8 : Paramétrage du linker (1)

Insérer alors la ligne de commande suivante dans la partie Enter other options :

-lmingw32 -lSDL2main -lSDL2

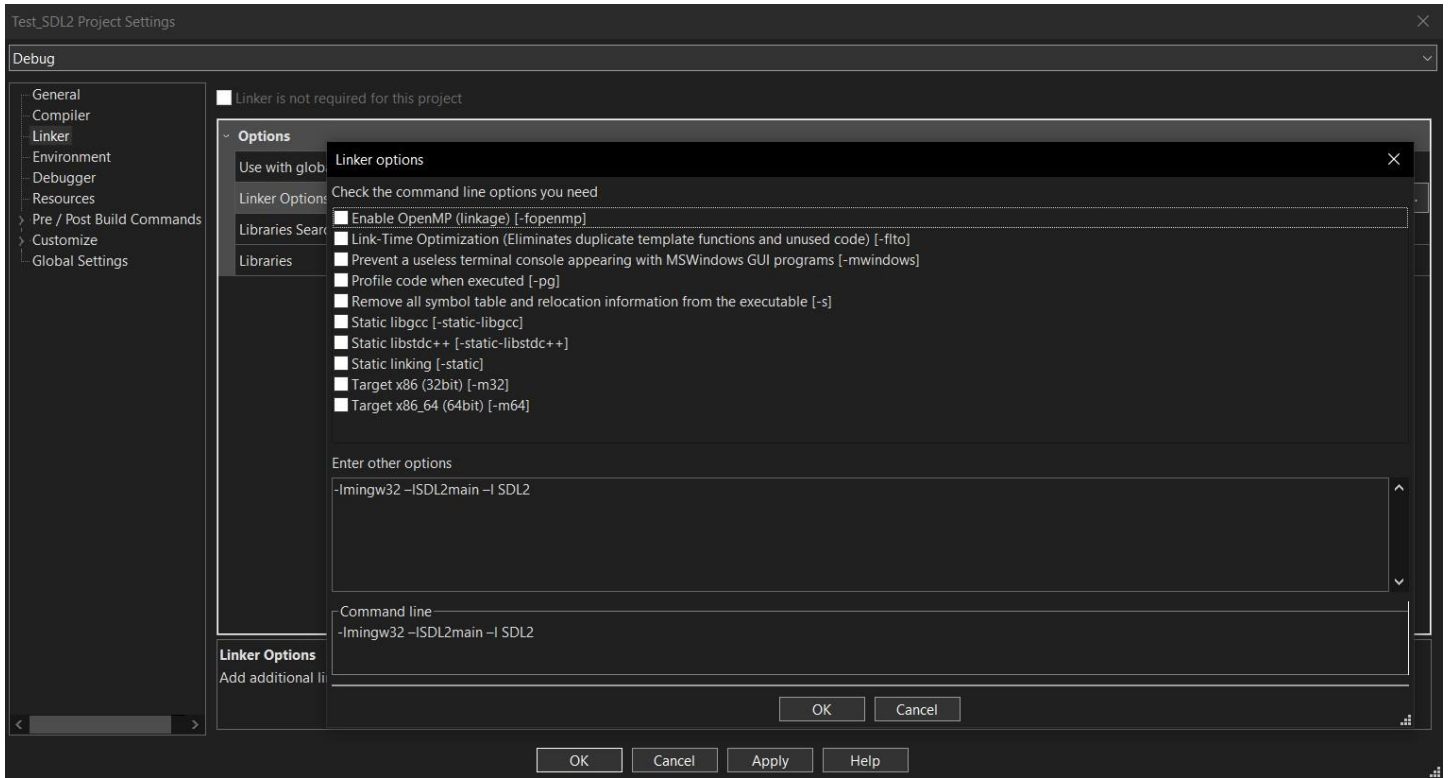


Figure 9 : : Paramétrage du linker (2)

Puis valider par Ok.

Puis appliquer les modifications en validant par Apply puis Ok

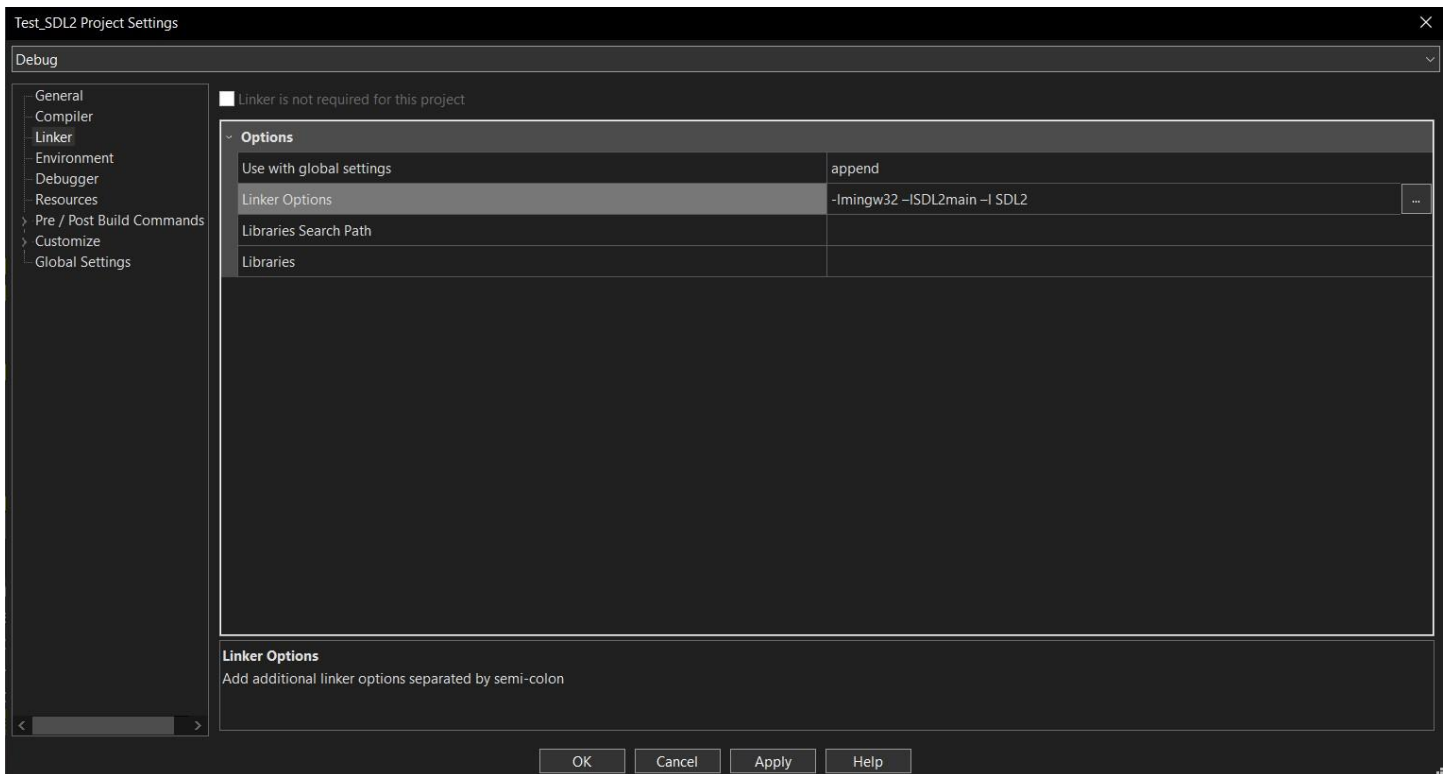


Figure 10 : Paramétrage du linker (fin)

Relancer le build du projet pour aboutir à :

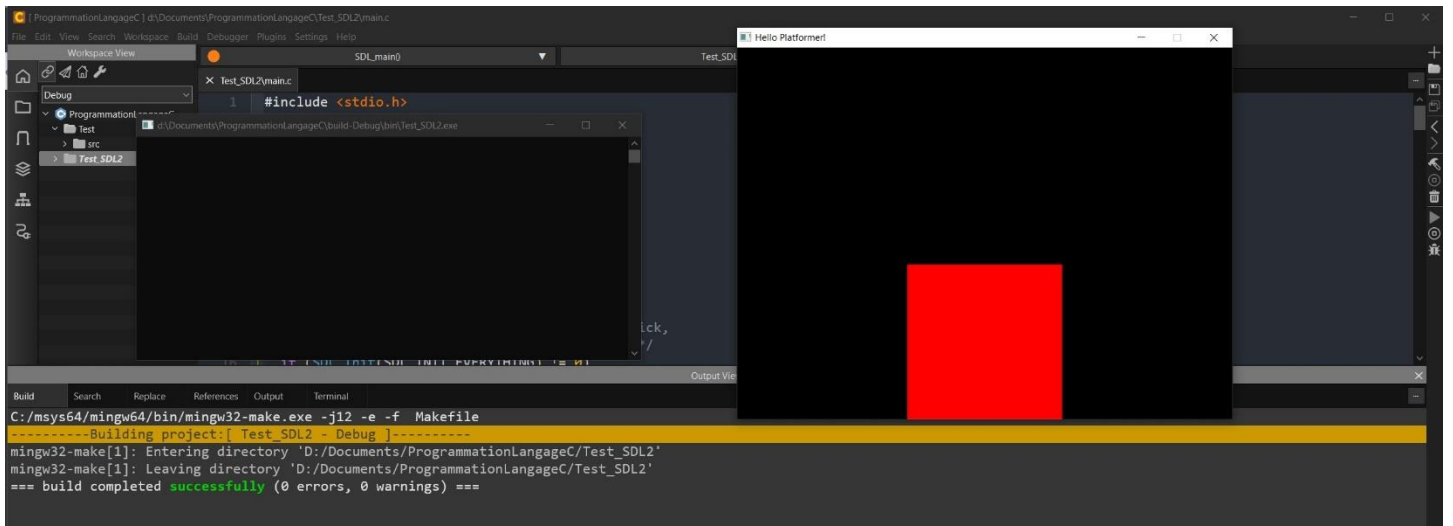


Figure 11 : Build du projet et exécution du .exe

Appuyer sur la barre d'espace ou sur les touches ← ou → et observer l'impact sur le carré rouge.

La prise en compte de la SDL2 est alors opérationnelle.

Le même résultat peut être obtenu en générant l'exécutable en ligne de commande par :

```
gcc -o test_sdl2 main.c -lmingw32 -lSDL2main -lSDL2
```

ce qui conduit à générer le fichier exécutable Test_SDL2.exe.

Le résultat de son exécution dans la console Windows de Windows est illustré ci-après.

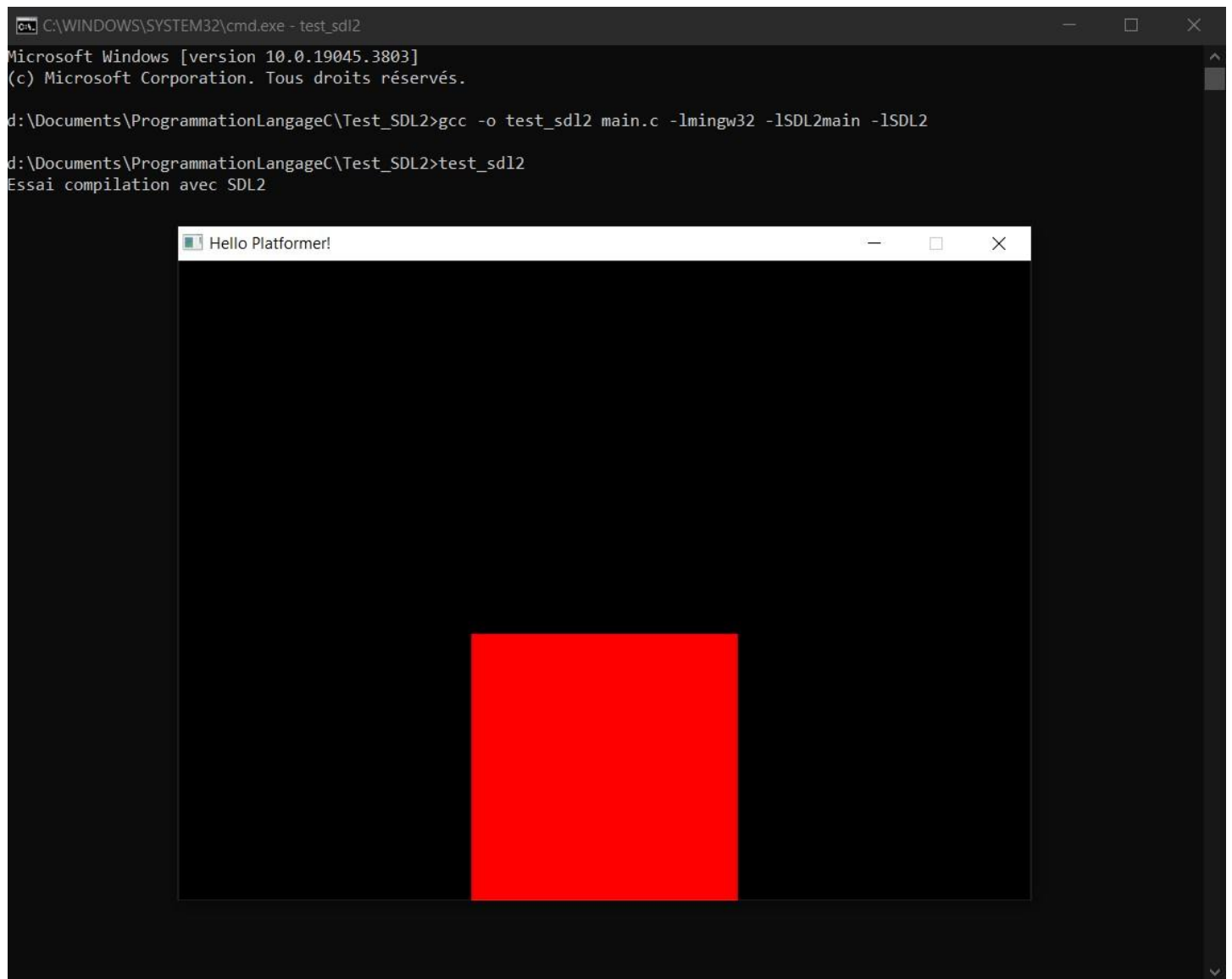


Figure 12 : Compilation en ligne de main.c et exécution de .exe généré dans la console Windows