How to compile SDL_bgi programs with Code-Blocks

These instructions show how to compile an `SDL_bgi` program using CodeBlocks; we will use `fern.c`, provided in the `test/` directory.

Let's assume that CodeBlocks (complete with MinGW and TDM-GCC) is installed in `C:\CodeBlocks`, and that the SDL2 development libraries (MinGW) are installed in `C:\SDL2-2.0.8`. If you installed the software in different directories, change the following instructions accordingly.

The procedure was tested with CodeBlocks 17.12 and SDL2-devel-2.0.8-mingw.

Links:

http://codeblocks.org/ https://libsdl.org/download-2.0.php

- copy `SDL_bgi.dll` to `C:\CodeBlocks\MinGW\bin`

- copy `C:\SDL2-2.0.8\i686-w64-mingw32\bin\SDL2.dll` to `C:\CodeBlocks\MinGW\bin`

- copy `SDL_bgi.h` to `C:\SDL2-2.0.8\i686-w64-mingw32\include\SDL2`

- copy `graphics.h` to `C:\CodeBlocks\MinGW\include`

- start CodeBlocks and click on `Create a new project`

- click on `SDL2 project`

- in the wizard, provide a project name (`fern`), pick a folder to create the project in, then click on `Next>`

- in the next window, pick `C:\SDL2-2.0.8\i686-w64-mingw32`

- in the next window, choose `Create "Release" configuration`, then click on `Finish`

- in the panel `Projects/Workspace/fern/Sources`, delete `main.cpp`

- in the panel `Projects/Workspace/fern`: right click, click on `Add files...`, pick `fern.c`, then click on `Ok`

- from the menu `Project/Build options...` select the `Search directories` tab, click on `Add`, pick the directory `C:\SDL2-2.0.8\i686-w64-mingw32\include`, and select `No` when asked to `Keep this as a relative path?`. Click on `Ok`

- from the menu `Project/Build options...` select the `Linker settings` tab, click on `Add`, pick the files `C:\CodeBlocks\MinGW\bin\SDL_bgi.dll` and `C:\CodeBlocks\MinGW\bin\SDL2.dll`, then click on `Ok`

- from the menu `Build`, select `Build`. The executable `fern.exe` will be compiled.

- run `fern.exe`.

Note 1: if you release your compiled program, you must also provide `SDL2.dll` and `SDL_bgi.dll` in the same directory.

Note 2: if your code uses functions provided by `stdio.h` (e.g. `printf()`, `scanf()`, etc.) you must also run a terminal. From the menu `"Project/Set program's arguments..."`, check the box `Run host in terminal`.