

GARA GRA i LABIRINTI

Compilazione

Compilazione

L'applicazione è stata compilata e testata con successo sulle seguenti piattaforme:

- **OS X – 10.8.4**
Compilatore Apple LLVM version 5.0 (clang-500.1.63)
- **Ubuntu Linux – 13.04 64bit – Kernel 3.8.0-29**
Compilatore gcc 4.7.3
- **Windows 7 SP1**
Compilatore MinGW 4.7.2

Dipendenze

Sono necessarie alcune dipendenze al fine di compilare l'applicazione:

1. CMake (2.8 o superiore) — <http://www.cmake.org>
2. Allegro (5.0.0 o superiore) — <http://alleg.sourceforge.net>
3. Dumb (0.9.3) — <http://dumb.sourceforge.net> — facoltativa, per la riproduzione audio
4. Compilatore C compatibile con lo standard c99

Compilazione libreria dumb

La libreria dumb non è fondamentale per il funzionamento ma è necessaria per la riproduzione delle tracce audio principali.

OSX/Linux

```
$ curl -o dumb.tar.gz http://kent.dl.sourceforge.net/project/dumb/dumb/0.9.3/dumb-0.9.3.tar.gz
$ tar xf dumb.tar.gz
$ cd dumb-0.9.3
$ make
Would you like support for Allegro (Y/N)? N
Please specify an installation prefix (default /usr/local) >
Configuration complete.
Run 'make config' to change it in the future.
Press Enter to continue ...
$ make install
```

Windows

Una versione precompilata della libreria è inclusa nei binari della libreria Allegro.

Compilazione della libreria Allegro¹

Linux

Per compilare la libreria sono necessarie una serie di dipendenze.

- Build Essentials (build-essential) - GCC, make, binutils

¹Per ulteriori, e più approfondite, informazioni è possibile fare riferimento a:
http://wiki.allegro.cc/index.php?title=Allegro_5#Installing_and_Building_Allegro_5

- X11 (xorg-dev) - Development libraries for X11
- OpenGL (libgl1-mesa-dev, libglu-dev) - Devel libs for OpenGL

Nel caso di Ubuntu, ad esempio:

```
$ sudo apt-get install libgl1-mesa-dev libglu1-mesa-dev cmake
build-essential make libxcursor-dev libdumb1-dev
```

installa le dipendenze di base.

OSX/Linux

```
$ curl -o allegro.tar.gz http://heanet.dl.sourceforge.net/project/alleg/allegro/
5.0.10/allegro-5.0.10.tar.gz
$ tar xf allegro-5.0.10.tar.gz
$ cd allegro-5.0.10
$ mkdir build && cd build
$ cmake ..
$ make install
```

Windows

Su Windows, la soluzione più semplice e rapida è sicuramente fare affidamento su una versione pre-compilata della libreria (per MinGW) scaricabile da:

<http://alleg.sourceforge.net/download.html#binary-version-only-for-windows>

Compilazione dell'applicazione

OSX/Linux

Supponendo che i sorgenti si trovino nella cartella *lasd/src*:

```
$ cd lasd/src
$ mkdir build && cd build
$ cmake ..
$ make
```

Se la compilazione è completata senza errori, rimanendo nella cartella *build*, è possibile avviare l'applicazione:

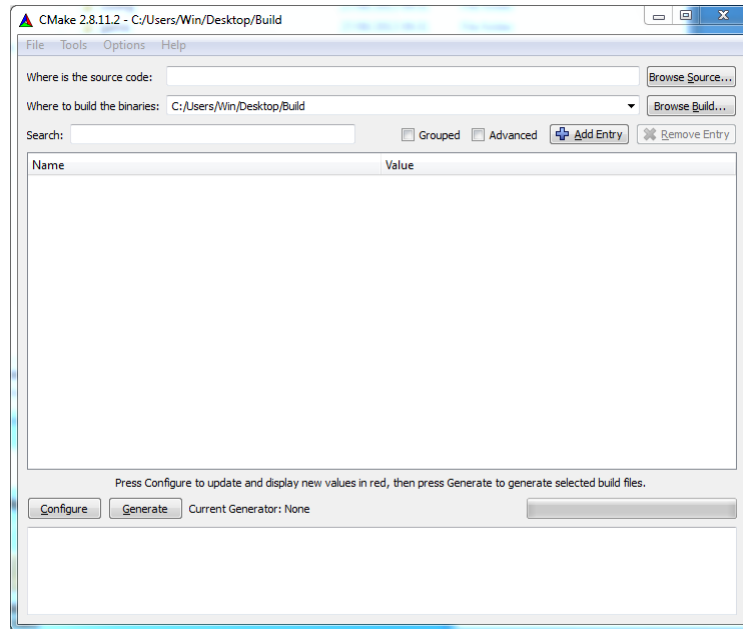
```
$ ./Gara
```

Windows²

Supponendo che binari e headers della libreria Allegro si trovino in *C:\allegro-5.0.10-mingw-4.7.0*

Generazione makefile Utilizzando CMake va generato un makefile per la compilazione dell'applicazione.

²Utilizzando come compilatore MinGW 4.7.2



- *Browse Source* e si seleziona la cartella contenete i sorgenti.
- *Browse Build* e si seleziona la cartella dove creare il makefile.
- A questo punto vanno definite alcune variabili cliccando su *Add Entry*
 1. Name: CMAKE_INCLUDE_PATH
Type: PATH
Value: C:\allegro-5.0.10-mingw-4.7.0\include
 2. Name: CMAKE_LIBRARY_PATH
Type: PATH
Value: C:\allegro-5.0.10-mingw-4.7.0\lib
- *Configure*, si seleziona MinGW Makefile come generatore e si prosegue.
- *Generate*, per generare il makefile.
- A questo punto tramite prompt dei comandi:

```
C:\Users\Win\> cd Desktop\Build
C:\Users\Win\Desktop\Build> mingw32-make
```

A compilazione avvenuta, prima di poter eseguire l'applicazione è necessario copiare i files:

- allegro_audio-5.x.x-mt.dll
- allegro_acodec-5.x.x-mt.dll
- allegro_font-5.x.x-mt.dll
- allegro_images-5.x.x-mt.dll
- allegro_primitives-5.x.x-mt.dll
- allegro-5.x.x-mt.dll

Nella cartella dell'eseguibile.

Binari precompilati

OSX <https://dl.dropboxusercontent.com/u/53929534/GaraOSX.zip>

Windows <https://dl.dropboxusercontent.com/u/53929534/GaraWin.zip>