**KrisBry Engine README**

Artificial Intelligence Project 3

To run the project, do the following:

* Ensure that KrisBryGame is set as the start up project
* Build the solution in Debug or Release in x86 or x64
  + The SimplePhysics or BulletPhysics Project should build automatically, but if not build separately before running.
* Run the Project (CTRL + F5)

Notes on specific implementation

* **PLEASE NOTE: BECAUSE OF THE WAY I AM CURRENTLY DOING THE INSTANCED DRAWING (will fix the hack and put it in its own shader eventually) PLEASE DO NOT USE YOUR INTEGRATED GPU AS THE SHADER WILL CRASH. PLEASE USE A DEDICATED GPU ONLY (I use the NVIDIA GTX-1050TI**
* You start the program with the bitmap loaded called resourceMap.bmp
  + This is Located in KrisBryGame/assets/bitmaps
  + To change the map, either overwrite the one in the location above with the same name, or place another one in that folder and change the file name in KrisBryGame/src/loadScene.cpp line 87
* There are 10 spawn points (so 10 gatherers) and 13 possible home bases to return to (More than one gatherer can occupy the same home base at the same time since the spec didn’t say otherwise)
* See grid on last page for all controls

Running PathFindingDemoSetup.exe will install a playable demo on your machine

* Once installed a short cut named Pathfinding Demo should be created in the Start Menu
* Or it is installed at C:/cnd/KerriganKristian/videogame
  + This is also where the uninstaller is found

Notes on each project in the solution:

* glm project contains the headers for the GLM math library (Use show all files in VS)
* PhysicsInterfaces contains all the interfaces used by the engine (Use show all files in VS)
* SimplePhysics contains the implementation of the interfaces above (Use show all files in VS)
* BulletPhysics contains the implementation of the interfaces using the Bullet engine (Use show all files in VS)
* KrisBryEngine contains the core OpenGL game engine (Use regular filters in VS)
  + Used to render graphical objects and lights to the screen
  + Controls the game loop
  + Contains managers for things like the Scene, Shaders, etc
  + Has factories and builders to create the entities and components needed in the engine
* KrisBryGame references KrisBryEngine and uses its system to implement the actual gameplay (Use regular filters in VS)
  + Loads JSON scene files
  + Controls player input
  + Loads the needed shaders, models, textures, etc.

The files to change the objects in the game are found in the KrisBryGame project under assets/maps  
Some notes about each type of JSON file there

config.json contains:

* Window width and height
* Window title
* Physics DLL being used (change between SimplePhysics.dll or BulletPhysics.dll)

cameras.json contains:

* Selected light and entity at the start of the game
* Eye, at, and up location of the camera at start
* Orientation of the camera at start

lights.json contains information for each light such as position, colour, type, cone angles, on/off, etc

sounds.json contains information for any background music being played

entities.json contains all the information about the objects in the game (planes and spheres)

* Each entity object has a unique ID and a type that is used by the builder to create the right object
* It also contains an array of components with a type defining its use/behaviour

**Controls**

|  |  |
| --- | --- |
| Key | Action |
| UP / DOWN arrow | Change selected rigid body (will turn red) |
| W | Move rigid body forward |
| S | Move rigid body backwards |
| A | Move rigid body left |
| D | Move rigid body right |
| I | Move camera forward |
| K | Move camera backward |
| J | Move camera left |
| L | Move camera right |
| Q | Move camera up |
| E | Move camera down |
| CTRL + Q | Roll camera to the left |
| CTRL + E | Roll camera to the right |
| Clicking the left mouse will increase camera move speed | |