There are couple of classes that I call big parents. For now they are not really very much effective as the name suggests.

Right now, the most effective one is Moveable class that will be inherited by every object which have dynamic movement. It makes command interface for the movements. Either the child class is having movements physically or not, it does not matter! Caller only needs to now it is Moveable…

Runners are the contains classes that packs the world up and creates it and a console class that used in the game to give commands.

GObject class is a parent that includes all the necessary proprieties every graphics object will need, such as lightening information or shader programs that graphic object will use when shadow mapping must be done (only depth calculation program).

Every Specialized class of GObject aka inherited from, will correspond a OpenGL shader program BUT it can also take a program from child of this concrete class so long as it obeys the layout location.

Currently Working on:

My own bullet file importer.

Texture splatting for terrain.

Shadow mapping.

Feature:

Normal mapping

Tri-Planer terrain texturing.

Every light may have mapping for shadow by simply adding sampler uniform data and binding their calculated maps. That also means for every light there must be a matrix push to the shader. Not so bright.