I defined my getWorld function in class Actor because every actor needs a way to access the pointer back to the world they exist in. Since this is the same for every actor, it need not be virtual.

Actors doSomething is pure virtual because you should not be able to make a plain “Actor”, you should only be able to make types of actors. Also, what a specific actor does during its tick depends on its type. I defined doSomething in this class because every actor does something, but not every actor does the same thing.

blockPath is defined in Actor because every actor shares this behavior. It is virtual because some actors do block the pathway while others do not.

blocksBullet is defined in actor because every type of actor either blocks a bullet’s path or does not. For this reason the function is also virtual.

stillAlive is defined in Actor because every actor needs a way to access the private member variable that tells them if they are alive or not. Since this is the the same for every actor it need not be virtual.

setDead is defined in actor and is not virtual because every actor needs a way to change its state of being from alive to dead. This is done in the same way for every actor.

2. My code seems to sporadically crash in level 2. I have been working on this problem all day, but I cannot seem to find the problem. I speculate that somewhere my code may be trying to delete a null pointer.

3. I assumed that if a robot shoots and a player moves out of the way, the bullet can hit and possibly damage another object. I also assumed that if a robot and a jewl are on the same space, you cannot move onto that space. Also, I assumed that no actors are placed outside of the 15x15 grid.

4. I tested my classes through thorough gameplay of each level and by making levels to create situations in which I thought my code might experience problems.