***Reflection***

**Changes made to project 2-A**

* I removed all the interfaces as I could not find an appropriate way of using them and they were easily replaced with some public methods.
* In project 2-A the explosion was being rendered using a class ‘effect’ but while coding I found it easier to just replace the cracked wall with the effect for the specified time.
* I had to add some more functions to the ‘World’ class; (+ static isBlock(float,float):boolean, + static isUnit(float,float):boolean,

+ static isTargetCovered(float,float):Boolean). These functions made sense to be in the ‘World’ as it contains the sprite array which we can loop through to check for information.

* I used a class called ‘Enemy’ to be a parent to the all the enemy units (rogue, mage, skeleton) part but my project 2-b did not make use of it. The three enemies did not have much in common as only the rogue can push; the mage follows an algorithm and the skeleton moves mindlessly. Therefore, they have their own interfaces and it made more sense to just keep them as a child of the sprite class.
* A lot of changes have been made to the children classes of sprite as well.
* The functions ‘isCovered(sprite)’ and ‘unlockDoor(sprite)’ also had to be removed as I just passed a boolean that showed if the door should render or not.

**Difficulties faced during the project**

The most difficult part was to get my head around the object oriented way of programming, which I do not think I have done successfully and there is a lot to learn. And because of not using the object oriented programming properly I got a number of errors when I add new elements to my game.

If you move the stones in level ‘0’ then they stop rendering but still exist on the screen and I cannot find a solution to this problem.

Moving the ice after it stopped (giving it that push after it collided with a block or wall) was difficult to get my head around.

**Key piece of knowledge**

Object oriented programming is a very powerful tool and an important skill to have, the project helped me get my head around inheritance and how every class has a special role. Also making changes to the program is also easy when you have specific tasks assigned to different classes.

**Things I would do differently in a similar project**

Using interfaces in a project in what I want to do and also make better use of object oriented programming.