# Lab 3

## Missile Command One

You are to create a research document for the game Missile Command One, a copy of which is in this folder. There will only ever be one incoming asteroid/missile and only one laser/missile to defend with. The longer you wait to fire your laser the higher (altitude not range) the laser will travel. You score 5 points per asteroid destroyed and the frequency of the incoming asteroids increases with your score. Once an incoming asteroid/missile strikes the ground the game is over. All movement, collisions and locations should be dealt with using sf::vector2f and operations from your own myvector2 library{copy files to project directory and add existing items to project}. [Do not use SFML intersects()]

Start with a new project (type SFML Game 2.5.1)

Or

Accept this GitHub classroom assignment (future assignments will be done this way only)  
<https://classroom.github.com/a/zaSArBm->

(you will need to be logged in to your GitHub account first)

Or

Clone <https://github.com/lowwwe/Lab3-Starter> and create your own GitHub repo[remote]

Or

Copy the folder Lab3Starter

You should include a narrative description where the third incoming missile strikes the ground.

In your inner workings you should describe how each the sub-systems for the [power: bar], [laser], [explosion], [asteroid], [collisions] work. Mentioning constants (guessed at this stage) required variables and logic at a level of draw laser, check distance between, increment location by velocity.

This research document is due at the start of next week’s lab make sure to show it to Pete before leaving.

Your coded version is due before Monday **2th of Decemberrreer @10:00**. Create a zip file on your **M drive** named **LAB3**yourName**.zip** that contains one copy of your project.

Anyone adding extra functionality before the basic version works fully will be deducted 25% automatically. [i.e. Textures]

Each work session should be detailed in Game.cpp and the total time calculated.  
Each variable should be commented when declared before use.  
Each method should be summarised when defined before coding.  
**Failure to do so will be heavily punished.**

When you have the basic game working you should add extra functions (20%) accessible by pressing “T” during the game. These could include

* Textures
* Sounds
* Advanced scoring
* Ramping up of speeds / difficulty
* Themes
* Animations
* Not multiple missiles or lasers or ground targets