Week 2: Day 003.1 – Peer Critique: "Simple" Dice Game:

Pair up with a fellow student, and critique each other's design and implementation of Day 001.1 and 001.3's "Simple" Dice Game.

Critique the project:

- Was a flowchart created?
- Did the flowchart follow standard flowchart rules and syntax?
- Has the game's functionality been implemented correctly?
- Are the required game design rules met?
- How was the 70% AI winning achieved?
- Have the Statistic Reporting requirements been implemented?
- Are there areas for improvement?

Also, discuss your impressions of using C++ and Visual Studio thus far to implement the "Simple" Dice Game.

Week 2: Day 003.2 – Peer Critique: Noughts and Crosses:

Pair up with a different student in the class, and critique each other's design and implementation of Day 001.2's Noughts and Crosses Game.

Critique the project:

- Was a flowchart created?
- Did the flowchart follow standard flowchart rules and syntax?
- Has the functionality been implemented correctly?
- Are the required game design rules met?
- Can a player win with three horizontal pieces in a row?
- Can a player win with three vertical pieces in a row?
- Can a player win with three diagonal pieces in a row?
- Is the end game scenario announced to the players, do you know who won and who lost, or if there was a draw?
- Are there areas for improvement?

Also, discuss your impressions of using C++ and Visual Studio thus far to implement the Noughts and Crosses Game.

Week 2: Day 003.3 – Space Invaders UML Class Diagram:

Using Microsoft's Visio, create a UML Class Diagram to represent the technical design of a Space Invaders (Taito, 1978) clone. Focus on the important classes, relationship, fields and methods.

Week 2: Day 003.4 – Weapon System Technical Design:

As a game programmer, you have been allocated the task to implement a weapons system for your company's new space shooter game.

In the space shooter game:

- The player can fit different kinds of weapons to their space ship, such as a mini-gun, missile launcher, laser turret and a mine planter.
- The weapons can be loaded with ammo.
- The ammo is specific to each weapon, for example the mini-gun ammo only works with the mini-gun.
- There are different kinds of ammo per weapon, for example a homing missile and a large-payload missile for the missile launcher.
- A weapon can be fitted with additional upgrades, which again is tied to the weapon type, for example a 3X-Laser cannot be fitted to a mini-gun.

Using Microsoft's Visio, create UML which decomposes the above game weapon system into a class hierarchy.