Assignment 3 Report

In this assignment, interface and abstract class and polymorphism structure is used. Every jewel object is created by using "Jewels" superclass and specific subclass.

For jewels, every jewel type has different matching type and score, so I prefer using a super abstract class for them. It contains the abstract matching method and if any match occurs, an "addScore" method applies and the variable "point" increases.

For math signs, all type of signs have the same score, so I prefer using an interface "MathSigns" and have static method "addScore". Again, every sign has different matching type so I used abstract method for this and override it in every sign class.

Although every jewel has different matching restirictions, the removing process was the same, so I used an interface called "Remove" and implement this interface to all the jewel classes.

Harun Harman

2200765008

Class Diagram:

Main
+ main(String[] args);

Command
+ command();

Jewels
- point int;

+ setPoint(point; int)
+ getPoint();   int

Extends

«abstract»
Game
+findKind(y: int, x: int, gameGrid :ArrayList<ArrayList<String>>, jewel :String);

+ addScore(jewel1 :String , jewel2: String);

Extends

Extends

Diamond
+ findKind();
+addScore();

Square
+ findKind();
+addScore()

Triangle
+ findKind();
+addScore()

Wildcard
+ findKind();
+addScore()

«interface»
Remove
+ removeJewels();

RightSlash
+ findKind();

LeftSlash
+ findKind();

Stick
+ findKind();

Plus
+ findKind();

Minus
+ findKind();

«interface»
MathSigns
+ operators; Arraylist;

+ findKind();
+addScore();   int