UML CLASS DIAGRAM

Character

- name :string
- gender :string
- health :int
- attackPower :int

+ specialAbility() abstrct :void
+ takeDamage(int damage) abstract :void
+ attack(Character target) abstract :void
+ ToString() abstract :void

Warrior

- weaponType :string
- rageLevel: int

+ specialAbility() :void
+ takeDamage(int damage) :void
+ attack(Character target) :void
+ ToString() :void

Archer

- quiverSize :int
- arrowsRemaining :int

+ specialAbility() :void
+ takeDamage(int damage) :void
+ attack(Character target) :void
+ ToString() :void

ArcherCommander

- totalUnderPlatoon :int

+ specialAbility() :void
+ takeDamage(int damage) :void
+ attack(Character target) :void
+ ToString() :void

Sorting Algorithm Comparison Charts

Line Graph

Scatter Plot of Grouped Data Points
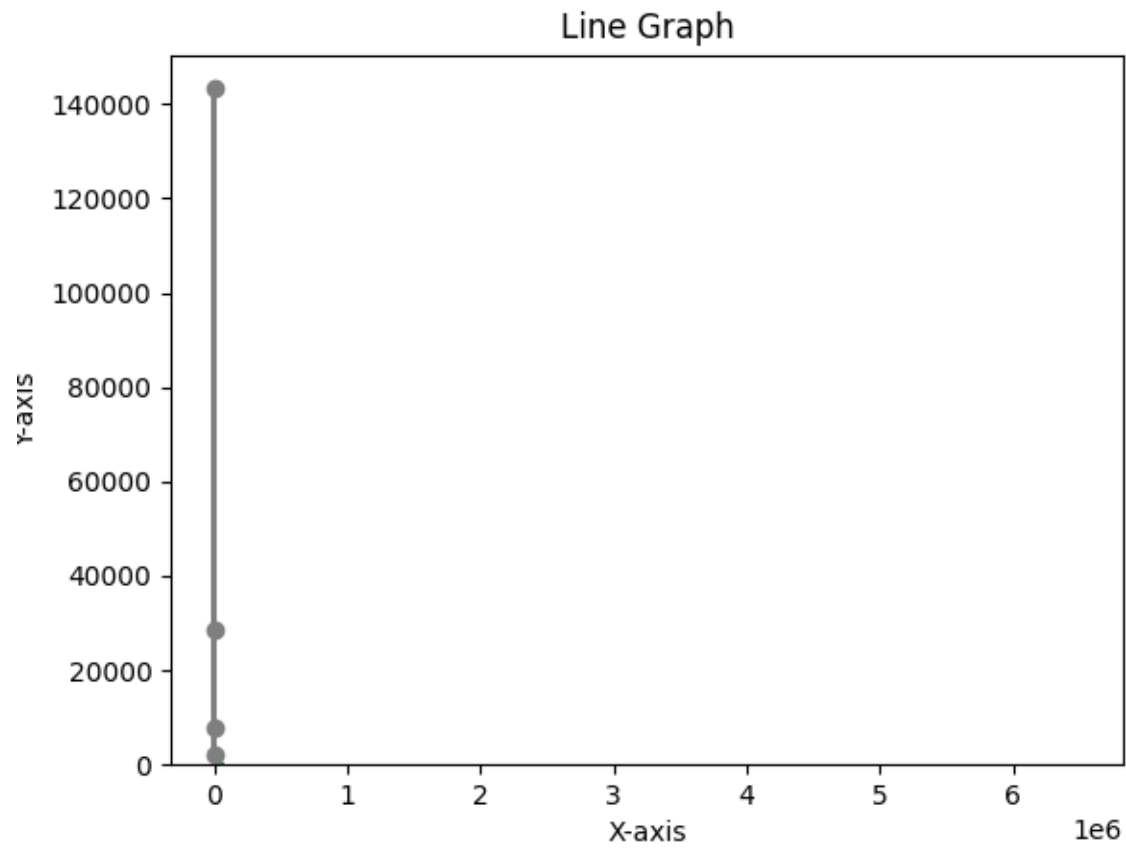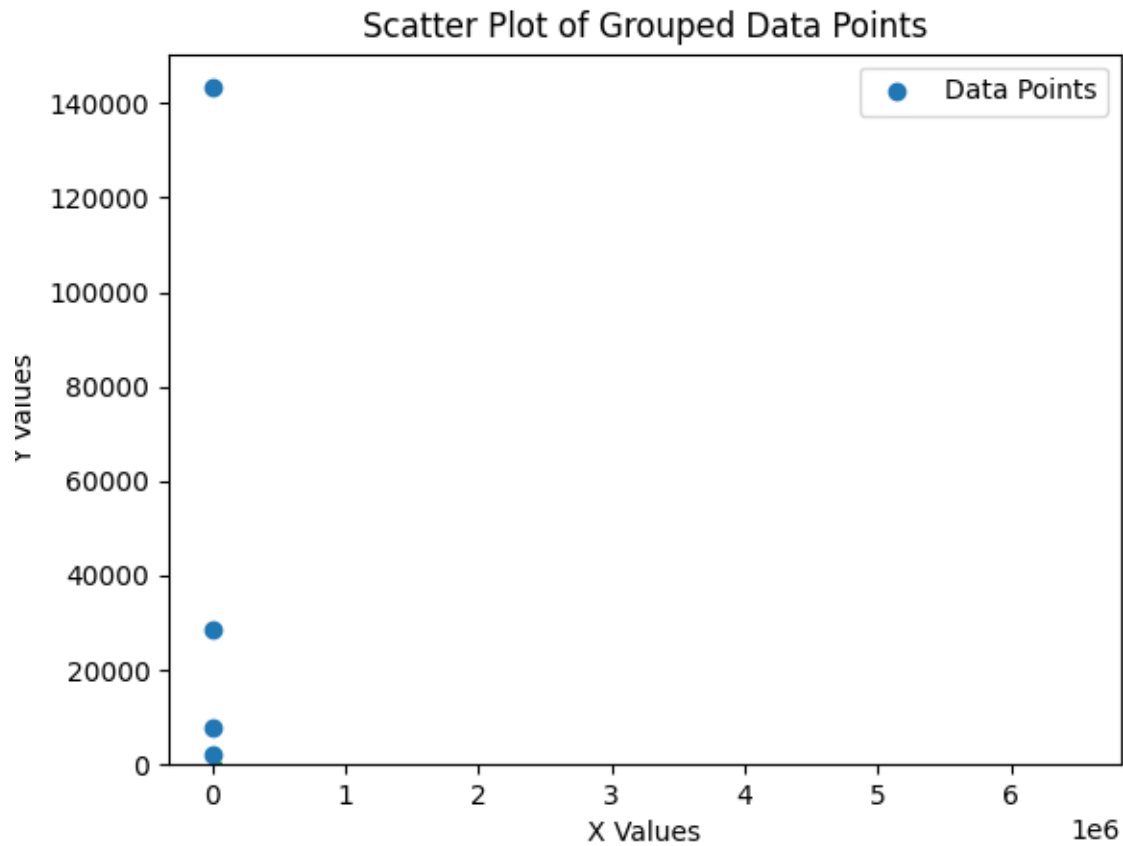
As we can see from the above results quick sort performs 7800 swap operations, Insertion sort perform 143307 swap operations, Merge sort performs 2068 comparison operations, Selection sort performs 28680 swap operations.
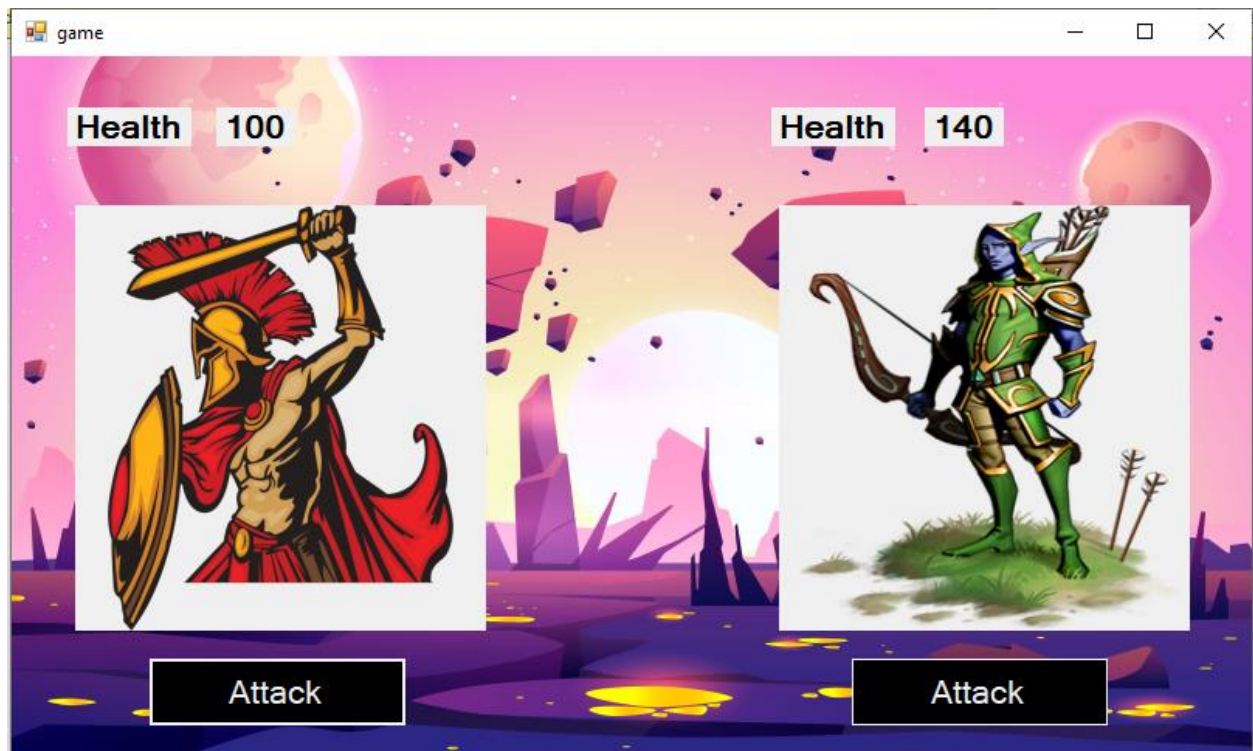
So here complexity of Quick sort is **O(n(log n)),** Merge Sort is **O(n(log n)),** Selection sort is **O(n^2),** Insertion sort is **O(n^2)**
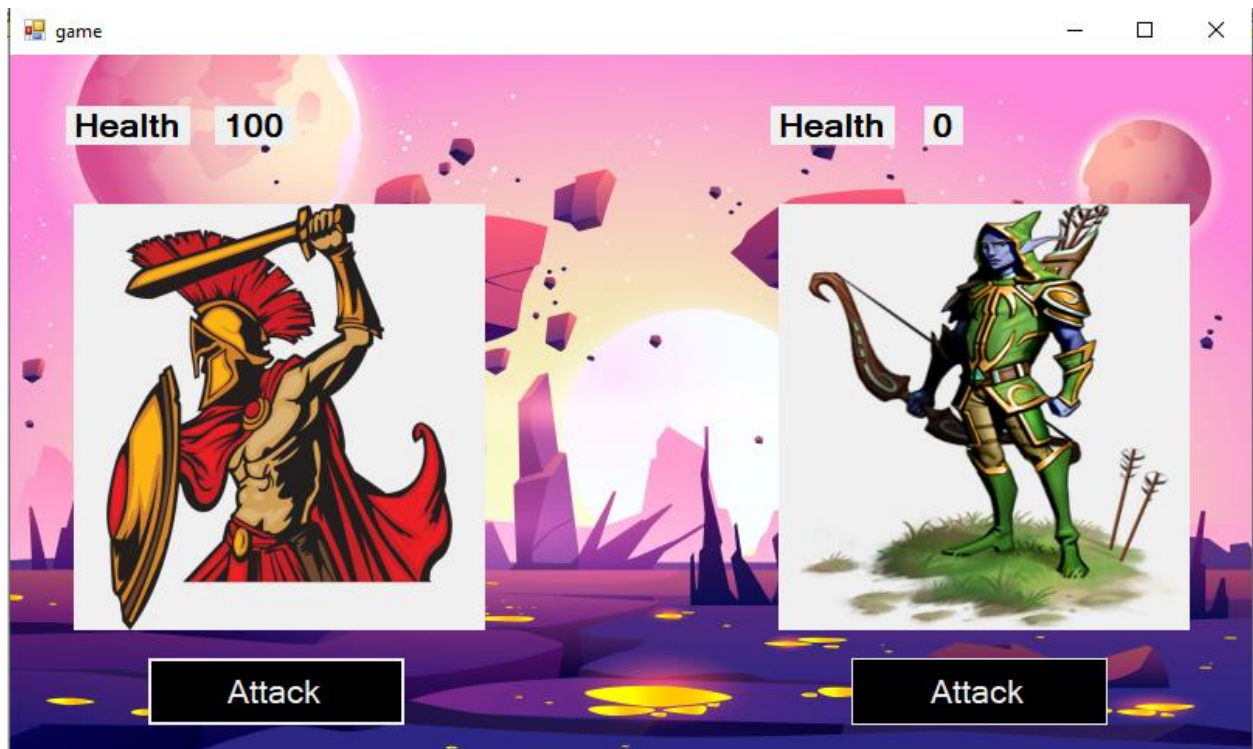
Unit Testing

In this unit testing we are testing attack function of both game characters. Initially both powers are displayed.
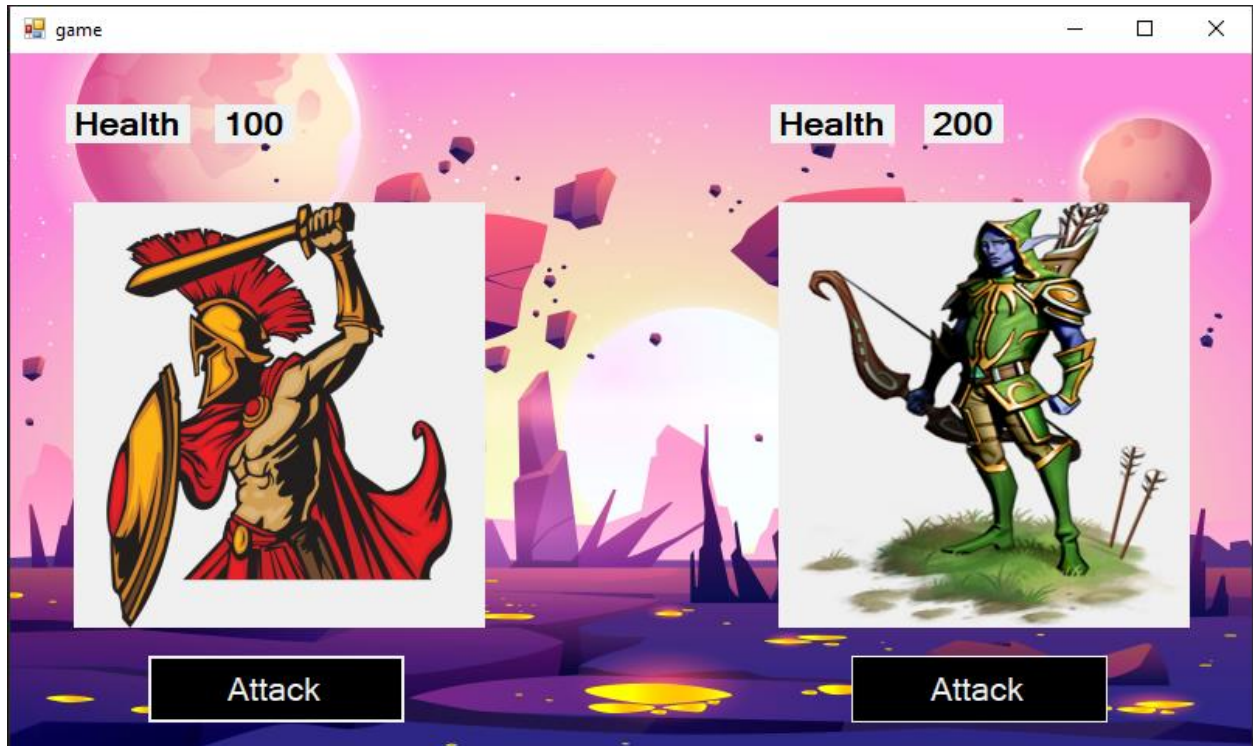
Now this Screen shot is taken when Warrior put 2 attacks on Archer with attack power of 30

Now after Archer died Warrior cannot hit Archer game character back as its power is already 0

So, the minimum power of both characters can be 0 i.e if they are dead.