CSCI 2073 – Programming Assignment 1 – Game Characters

In a popular game, the world is inhabited by a variety of characters, which you are to represent using an ***abstract*** `GameCharacter` class. Every character has a name, a number of health points, and a location in terms of (x, y) coordinates in an imaginary world. Accordingly, every character should respond to the public methods:

- `String getName()`
- `int getHealthPoints()`
- `int getX()`
- `int getY()`

Characters are either archers, wizards, or knights. All characters are initially created with 100 health points. In addition to the methods above, all characters are capable of performing the following actions:

- `void move (char direction, int distance)`, where direction is either 'N' (increase y position), 'S' (decrease y position), 'E' (increase x position) or 'W' (decrease x position). Wizards can move any number of distance units, whereas archers are limited to five distance units per move, and knights are limited to two distance units per move. For archers and knights, distance values larger than their respective limits should be considered as equal to the limit. Negative distances should be considered as zero and should have no effect on the character's position.

- `boolean attack (GameCharacter target)`, where the argument represents the target being attacked. The return value indicates whether the attack was successful or not. For an archer's attack to be successful, the target must be within a distance of 30 units. For a wizard, the target must be within a distance of 50 units; and for a knight, the target must be within a distance of 5 units. To calculate distance, use the formula for the geometric distance between two points. If successful,
  - The target being attacked loses a number of health points equal to the attack strength of the attacker (30 if attacker is a wizard, 20 if attacker is a knight, 10 if attacker is an archer). However, under no circumstances should the resulting health points be negative.
  - When, as a result of the attack, the target's health points are less than 10, the target is rendered inactive (and therefore, too weak to either attack or move). Therefore, any calls to attack or move by an inactive character should have no effect.

- `String toString()`, although the precise format of the String returned is left up to each student, the contents should be labeled and include name, character type, health points, and (x,y) coordinates.

Your assignment is to create a Java class hierarchy to represent game characters. In addition to the abstract superclass, you should provide subclasses Archer, Knight, and Wizard, to be created using the constructors:

- `Archer (String name, int x, int y)`
- `Knight (String name, int x, int y)`
- `Wizard (String name, int x, int y)`

To test the basic syntax and functionality of your classes, the p1Test program should be downloaded and stored in the same folder as your classes. Compile and execute p1Test. If this test program and your classes do not work well together, you need to modify your classes.

When you are satisfied that your solution works, submit the files below (either individually or in a single .zip file) to Mimir for testing: GameCharacter.java, Archer.java, Wizard.java, and Knight.java.