You are given a game scenario with classes Goblin and GoblinKing. Please implement the following rules:

* A goblin has base 1 attack/1 defense (1/1), a goblin king is 3/3.
* When the Goblin King is in play, every other goblin gets +1 Attack.
* Goblins get +1 to Defense for every other Goblin in play (a GoblinKing is a Goblin!).

Example:

* Suppose you have 3 ordinary goblins in play. Each one is a 1/3 (1/1 + 0/2 defense bonus).
* A goblin king comes into play. Now every goblin is a 2/4 (1/1 + 0/3 defense bonus from each other + 1/0 from goblin king)

The state of all the goblins has to be consistent as goblins are added and removed from the game.

Here is an example of the kind of test that will be run on the system:

class FirstTestSuite(unittest.TestCase):

def test(self):

game = Game()

goblin = Goblin(game)

game.creatures.append(goblin)

self.assertEqual(1, goblin.attack)

self.assertEqual(1, goblin.defense)

**Note:** creature removal (unsubscription) does not need to be implemented.