The goblin function is not recursive.

* There was an attempt to create a stack of possible routes in which the goblin can take to get to the player by first taking into the goblin smell distance and randomly feeding in directions to find the path that required the least amount of steps.

Description of Design

* The classes are divided into 3 main groups. Dungeon, Actors, and Game Objects. The Game class connects the dungeon class (map ) and Actor class together. The Dungeon class is responsible for updating the arena by adding/removing items as the game progresses. The Dungeon class also has rooms and regions in which it is split up. The dungeon class also has lists for the currently existing actors and objects that have spawned on the map. In addition, there is a vector for the action strings to be displayed and are triggered when actor classes come into contact with each other.
* Actors are split into player class and monster classes. The player class has the ability to interact with the other Game object classes. It has access to it’s inventory and can move/attack/and interact with items.
* Monster classes all have a doAction() function that will have it move around as long as it it within range of the player class. The game is controlled input by input based on what the player puts in.
* Game Objects are split into weapons and scrolls. Weapons increase the stats of the player and scroll classes can increase a player’s stat’s when read. All game objects have a name and action string attached to it upon construction because there is a message that is outputted when it is interacted with.
* The game::Play() function brings everything together and progession of the game depends on player input. The player class is the only class that calls upon other gameobjects as it has the ability to do the most things. Other monsters simply wait until they are triggered to move.
* Aside from the goblin class not being able to move as efficiently as I would have liked it to, there are bugs with the program where sometimes the levels would fail to create and monsters. My initial function to create rooms and corridors had trouble linking with the rest of the project so I had to opt for a method that created regions and separated the arena in sections in which to be connected by corridors.