

## MazeItem

```
Public:  
virtual void render()  
virtual bool passThrough()  
virtual bool pickUp()  
virtual bool edible()
```

## MazeWall

```
bool mHorizontal  
bool mSecretPassage  
MazeWall(bool horz)  
virtual void render()  
virtual bool passThrough()  
void setPassage(bool p)
```

Shield, Sword, Grape, Apple, CandyCane

```
Public:  
void render()  
bool passThrough()  
bool pickUp()
```

## moveableItem

```
int x  
int y  
int attack  
int health  
moveableItem()  
int getX()  
void setX(int x)  
int getY()  
void setY(int y)  
void moveLeft(Maze* maze)  
void moveRight(Maze* maze)  
void moveUp(Maze* maze)  
void moveDown(Maze* maze)  
virtual void updatePosition(Maze*)  
virtual int getAttackValue()  
virtual int getHealth()  
void setAttack(int num)
```

## Character

```
vector<MazeItem*> inventory  
int numItemsEaten  
int armor  
Character()  
~Character()  
int getArmor()  
virtual void render()  
void addItemToInventory(MazeItem* item)  
void renderInventory()  
void renderHealth()  
int getHealth()  
void attack(moveableItem* mazeItem, Maze* Maze)
```

## Boss

```
Public:  
Boss()  
virtual void render()  
virtual bool passThrough()  
int getX()  
void setX(int x)  
int getY()  
void setY(int y)  
void attack(moveableItem* mazeItem, Maze* Maze)  
int getHealth()
```

## Ghost

```
int movement;  
Ghost(int movement)  
virtual void render()  
virtual bool passThrough()  
virtual void updatePosition(Maze*)  
int getHealth()  
void attack(moveableItem* Hero, Maze* Maze)
```

### Maze

```
int mazeHeight;  
int mazeWidth;  
vector<moveableItem*>  
void heroDidMove()  
MazeItem* constructItemForChar()
```

```
MazeItem*** maze  
Ghost* ghosts  
Character* hero  
Boss* boss  
Maze(string* initialMaze, int numLines)  
~Maze()  
bool canmove(int x,int y)  
void render()  
void updateMovableItemPositions()  
void removeItem(moveableItem* Item)
```