

Project Name: Final_Project
Name: Tanya Yeu
Date: 6/11/2019

Requirements

1. Space class - represents the space the player can be in
 - a. Abstract class that has pure virtual functions
 - b. 4 pointers - top, right, left, bottom
 - c. user the class to create a game with the structure of linked space
 - d. create a grid map space?
2. Any unused pointers point to NULL
3. At least 3 derived classes that are derived from Space
 - a. Each represents a different type of space
 - i. dungeon
 - ii. forest
 - iii. town
 - b. Needs to have a special action for the player to interact with
 - i. battle
 - ii. buy potions
 - iii. rest
4. At least 6 spaces
 - a. Fangor Forest
 - b. Deadwood Forest
 - c. Mirkwood Forest
 - d. Kirkwall town
 - e. Rivendell town
 - f. Moria Dungeon
5. Theme
 - a. Fantasy
6. Goal
 - a. Defeat the wizard in the dungeon
 - b. dont die
 - c. dont let the town die
7. Keep track of where the player is in the game
 - a. print text describing surrounds
 - b. print map
8. Container to carry items
 - a. container must have capacity limit
 - i. backpack = 6 items
 - b. Game must contain items for the player to obtain
 - i. one or more must be required as part of the solution
 - ii. potions
 - iii. KEY to open moria
 1. 45% drop rate off creatures in the forests
9. Time limit

a. town health

Requirements	Implementation
Spaces	Dungeon, Town, Forest
Theme	Fantasy / ye olden times
Goal	Defeat the wizard in the dungeon before the town is completely overrun
Time limit	Town health goes down
Status/Location	Space *playerLoc
Container for items	string backpack. character to show what each item is P - potion H - hi-potion K - key

Brainstorm

1. Goal
 - a. Escape the dungeon before it collapses?
 - i. there are monsters
2. Story -
3. Theme - DND fantasy style?
4. Environment
 - a. Spaces
 - i. forests
 - ii. towns - shops and gossip?
5. Items
 - a. Potion - Restore HP
 - i. revive - can save from death
 - ii. Hi Potion
 - iii. Potion
 - b. Key - to open dungeon - must have
 - c.
6. Character design
 - a. HP (Health Points)

Program Design

1. Character
 - a. HP
 - b. maxHP

- c. backpack
 - i. open backpack
 - ii. use item from backpack
 - iii. get number of items in backpack
 - iv. has revive in backpack
 - v. use revive on death
 - d. recover health
 - e. gold
 - f. is alive
 - g. has key to enter dungeon
- 2. Knight - derived
- 3. Enemy - derived
- 4. Space
 - a. pointers: top, bottom, right, left
 - b. interact
- 5. Forest
 - a. battle creatures
 - i. +5 gold
 - ii. 45% drop rate for key
 - iii. if town is almost dead, 100% drop rate
 - b. you don't really die in the forest, 1hp
- 6. Town
 - a. Rest
 - i. +5hp, -5 for town
 - b. shop
- 7. Dungeon
 - a. Light torch
 - i. Battle
 - ii. if not - 5hp (disadvantage)
 - b. 50% chance to spawn another battle if is lit
 - c. fight the wizard
 - i. wizard has 30hp
- 8. items
 - a. potion +5
 - b. hi potion +10
 - c. revive (use when dead) +20
 - d. key - must have to enter dungeon. can only be looted from creatures in the forest

Testing Plan

Test Case	Function	Expected	Observed
Linked map structure	travel(direction)	travel in a circular linked manner	segmentation fault at one particular space cause: did not link this specific space to the other spaces. copy and paste error
Linked map structure	travel(direction)	travel in a circular linked manner	as expected! character was able to travel across the map
Print map	printMap()	print the map if chosen from menu	as expected
Player status	*playerLoc->printInfo	any functions associated with the space would reflect the correct space	as expected
town Health goes down per "round"	do while in playGame townHealth--	The townhealth would continue to go down for each round	as expected
town menu	rest +5HP, -5 to town	player HP does not go over max but town hp can go to zero	game did not end after town hp dropped to zero. continued to go into negatives. cause: did not create a if statement for this case
town menu	rest +5HP, -5 to town	player HP does not go over max but town hp can go to zero	as expected. included statements to account for town health going down due to resting
town menu	go to shop	display items	as expected
town menu	shop - add item to backpack, subtract gold	item is added to backpack, gold is subtracted	as expected

Test Case	Function	Expected	Observed
display backpack contents	openBP	shows the contents of the backpack with index starting at zero	it was missing an item. cause: index started at 1 so it did not display index 0
display backpack contents	openBP	shows the contents of the backpack with index starting at zero	as expected. fixed index to say 1 but start with the 0 index
battle simulated in forests	battle(character, character)	battle will display each round of damage done and display winner	as expected
Create new enemy with random health	Enemy()	new enemy created for battle	as expected but enemies were often too strong making player die early in the game. adjusted health settings
battle with enemy	battle() enemy()	battle with enemy	variable healths led to player dying a lot. changed forest setting to if you die, you can still leave with 1 health to revive in a town or buy potions
shopping in towns	subtract gold	cannot go to shop if you have no money	as expected
do not add to backpack if backpack is full	addToBP()	stops user and says backpack is full	added 1 extra item. cause: index started at 1
enter moria without a key	isValidDir()	shows the option of going to moria but does not allow the user to travel there without a key	as expected
travel only in a valid direction	isValidDir()	validates the direction before traveling that it is not a null area	as expected

Test Case	Function	Expected	Observed
shows revives in the shop if it has the goods	hasGoods	displays revives as 0 in the menu for rivendell but not kirkwall	as expected
uses the revive if the character "dies"	hasRevive useRevive	will bring the character back to life	game ended instead before that happened
uses the revive if the character "dies"	hasRevive useRevive	will bring the character back to life	as expected after moving code around
use potion and remove it from the backpack	useItem	it will add HP and remove from backpack	as expected
Dungeon - if it is lit start a battle	isLit	starts battle and then continues into dungeon if user is still alive	as expected
50% to start another battle if dungeon is lit	isLit rand	sometimes battle starts, sometimes no	as expected
battle final boss	battle	finalboss has 30HP but does regular battle	final boss always had a random number HP cause: battle was creating a new enemy instead of taking an already created one
battle final boss	battle	finalboss has 30HP but does regular battle	as expected after editing battle function
if player dies end game	isAlive	ends the game once user dies	sometimes menu displayed instead. added more if states with isAlive and worked as expected
end game with congratulations if finalboss is defeated	isBossDefeated	ends the game with congrats	as expected

Reflection

One of the biggest challenges I faced while coding this in particular was how to do the backpack. I completely over-complicated things by creating a backpack class - similar to the linked lists class we had done in a previous lab. so I did that, but I found trouble removing 1 particular item from the middle of

the list and relinking everything. In general, inside a backpack it wouldn't be all linked anyway. I had also created an Object class with derived object classes for a bunch of things but then realized it was not necessary and created way too many files and became difficult to use to pass objects back and forth. I also attempted to use a queue<object> type of thing but that was basically the same as linked lists. I then tried to do an array of objects. At the end of it all, I scrapped all of this and just went with a string to carry all the "objects". It was simple to add items and remove certain ones in the center of it while also maintaining a size limit.

Besides this, I had difficult time deciding on story. In the beginning, it was difficult to conceptualize spaces and derived spaces. I saw the example of the bedroom and kitchen on piazza and I got stuck into thinking about just rooms. I couldn't come up with a story inside a house. I had to think bigger about how spaces could encompass something more to have more interactions to it. I spent quite a bit of time with pen and paper drawing out how to map out the spaces and how to travel across them, and what each space should do. I started out with coding just the spaces and linking them together and then testing that the player could travel across all of them. At this point, I did not include any of the limitations yet.

After adding the town health and space interactions like buying potions and stuff, I decided to add +5 gold for defeating enemies. I didn't have it first, but when I was testing I found that I usually spent all the gold in the first town before getting to the next one that sold more expensive items I wanted to show. With the +5 gold, you can grind for more money to buy more items. I say grind loosely because the town health goes down pretty quickly with traveling. I messed around a lot with the drop rate for the key. It was originally 30% but the game just kept ending with the key never dropping. After changing it to 50% it seemed like it always dropped on the first encounter. So I ended up lowering, but made it 100% drop if town health was almost up so that the game could at least continue onto the dungeon portion.

Ultimately, I was able to reuse a lot of code I had done for previous labs and projects and tweak them to my liking, as I'm sure was intended anyway. I took the code from the last project with the tournament for battles, but I found it less exciting when I couldn't see how the battle was going with just who won and lost. This project was a great way to tie up a lot of what we've done for this quarter and I really enjoyed working on it.