

Final Project Proposal

yoRPG remade with items, and grid movement

Grid movement:

Protagonists will move on a rectangular grid, which will be displayed in the terminal. Randomly selected tiles on the grid will have monsters and shops. The protagonist's tile will be randomly selected and represented with the letter p, and the protagonist will have coordinates, stored in a 2d array, that correspond to their location on the grid. The protagonist can move by changing these coordinates, and this change will be reflected on the grid. If the party moves onto a tile with a monster on it, a battle begins.

Battle

Turn order is decided by sorting the speeds of the protagonist and the monster. After the protagonist makes a move, their speed becomes the difference between their speed and the monster's speed. If the protagonist is still faster, they get a second attack. This can also happen if the monster is faster. Speeds are reset to their original value after the opposing party attacks. Protagonists can use items before an attack to boost stats until the monster attacks

Items

After winning a battle, protagonists will receive gold, which they can use to buy items that can boost their stats. Items will be separated into consumables and weapons. Consumables

have a turn limit, and after a protagonist makes a certain number of turns, the effects of the consumable will end. Weapons provide a constant stat boost, so long as it is equipped.

NPCS

All monsters, protagonists, and npcs will descend from a Character class, but there will be an interface only for fighters (monsters and protagonists). Characters without this interface may sell items or provide effects when interacted with on the grid.

Boss Battle

When the game starts, all monsters and npcs will be shown. The boss monster will have a tile with a unique letter, and moving onto the tile will start a battle. Defeating this monster ends the game.

Driver

All events will take place in a driver class that will keep track of turns for item duration, and will display the grid when out of battle. A scanner will be used to take player inputs to move the party around the grid and to select options. Everything that happens will be stored in a loop that ends only when the boss is defeated

Tools / topics we are hoping to implement with this project

- Inheritance and Constructors: Character class and all its descendents
- Interface: Protagonists and Monsters
- Arrays and 2d arrays: Item inventory, grid coordinates
- Search algorithms and Math.random(): Randomly searching for tiles to become monster / player / npc tiles
- Sorting algorithms: Speed based turn system in battles
- Iteration and recursion: Driver / running the game, grid creation
- Scanner: For all cases where option select is needed

Notes and additional ideas

A level system would be interesting, but we don't know if it will fit with any of the major tools or topics we've learned, or if it would be integral to the project