

Jordan Hamilton

Professor Luyang Zhao

CS 162

4 December 2018

## Final Project Design and Reflection

### **Design:**

The goal for this project is creating a wilderness survival game that simulates a survival expert searching for a trapped hiker, navigating through inherited classes of linked Space objects, with different properties and virtual functions. The game is based on user input handled by a menu class with input validation to prompt the user to search their current space to reveal surrounding spaces before traveling to them, thus allowing the user to find the hiker while searching along a path of terrain that reduces the player's energy as little as possible.

I hope to link these space objects on a 2-dimensional array with a random starting location for the player and hiker. The player will use a backpack as a container that is filled with items while searching in the player's current space, and these items must be used by the player to restore energy. If the backpack becomes too full, the player can't continue searching in new spaces and must consume those items. If the player's energy is too low, they cannot move and must consume items to continue moving. The player loses the game if their energy is depleted and their backpack has no items they can use to replenish their energy. Finally, searching has a chance to reveal the radio, an item collected randomly by searching which can only be used when the player finds the hiker and is at the hiker's location.

**Test Plan:**

Test Case	Input Value	Expected Result	Actual Result
Energy is depleted and backpack is empty	N/A	Game ends	Game ends
Energy is depleted and backpack has items	N/A	Player has a chance to use items to restore energy	Player has a chance to use items to restore energy
Energy is depleted and only radio remains in backpack before player is at hiker's location	N/A	Game ends	Game ends
Player uses radio while not near hiker	3	Error message is displayed	Error message is displayed
Player searches from peak	1	All unexplored linked spaces are revealed on the map	All unexplored linked spaces are revealed on the map
Player searches with 10 items in pack	1	Error message is displayed	Error message is displayed
Player tries to search in space that was already searched	1	Error message is displayed	Error message is displayed
Player tries to move out of bounds	2	Error message is displayed	Error message is displayed
Player moves to space where hiker is	2	Radio use instructions displayed to win game	Radio use instructions displayed to win game
Player uses radio on space where hiker is	3	Game ends	Game ends

**Reflection:**

When designing this project, I hadn't considered where to gather information about items in the backpack. This was difficult to consider after I had already begun writing the game class, because the gameplay was dependent on information like the number of items in the backpack to end the game if it was empty and the player was out of energy, to not allow additional searches if the item limit had been reached, or to restore the player's health when an item in the backpack was used. Since the backpack was a pointer to an object in the player class, and the player was a pointer to an object in the game's class, this made accessing that information more complex than I had anticipated. My original intent was to have accessor functions in the player class, to get properties of the items in the backpack vector, but this felt like an unnecessary step when the backpack was a class itself that could have accessor and mutator functions. I struggled with altering the player's energy correctly based on what was used from the backpack when having a pointer to a backpack object as part of the game class, which also felt inaccurate since the items were properties of the player object in a sense. I decided to use a `getBackpack` function which returned the pack object of the player, which felt like a good compromise. This allowed me to use the backpacks' public functions from the game class with `player->getBackpack()->hasItems()`, for example.