

Syrma: Mid-Point Check

Syrma Team Members

Britany Hughes
William Taylor Mack
Andrew Sturtevant

Submission Contents

Included in Mid-Point Check:

- Syrma: Mid-Point Check (PDF)
- SyrmaTextAdventure Zip
 - README Detailing how to compile and use
 - C++ Source Code
 - survivingSpace binary
 - RoomFiles
 - The current room data files
 - Parser
 - Functionality text files
 - Parser data files

Project Status

The Syrma team is about halfway in our implementation of our proposed game. While we have about $\frac{1}{3}$ of the target number of rooms completed, it is our belief that once the game engine, I/O and parser are hammered out, adding additional rooms will be a fairly straightforward process. Right now, we have our starting screen, with simple ASCII word-art, and the three basic options. Right now the only options that currently works in the menu is to “Start a New Adventure” & “Exit”. The basics of loading a game are complete, however currently our save function is not functioning so there is nothing new to load. Once inside of the game, the user starts in a “room”, can look around and can attempt to interact with a couple of objects. The user can also

go to specific rooms, 'take' certain objects. Finally, the user can also use the 'help' command to get some helpful hints.

User Instructions

Starting Game

There are a couple of ways you can play "Surviving Space!"

1. Recommended (not Required): Adjust console size to at least 152 columns.
2. Input the command `./survivingSpace` while within the the directory that the survivingSpace binary resides. (Note: The binary should not leave the directory that has all of the source code)
3. Follow the instructions in the provided README.

Navigating the Game

Using just the console, the user can input commands to their character. For this report, those commands that can be issued are:

- Look
- Go [Insert Room Name]
 - The room names currently are:
 - Starting Room
 - Top of Tree
 - Field with Grazing Animals
 - Pool of Water
- Take
- Help
- Exit

Aliases for several of these commands are functional; however, traveling via directions (up, down, left, right, north, south, west and east) as well as the command "walk" have not been properly implemented yet and should be avoided. A full listing of what verbs and nouns are recognized by the parser can be found in the XX_functionality.txt files in the Parser folder. However, it should be noted that aside from the above listed commands and their aliases, the rest of these commands are not currently implemented in the game engine.

Game Objective

The objective of our game is to solve a number of puzzles to not only get off of the planet that you are stranded on, but to restart your ship. Successful completion of the game is considered to be once the player navigates away from the planet!

Gameplay

Right now, there is only one puzzle in our current version. By the end of the product, there will be at least 5 different puzzles for the user to solve before being able to successfully complete the game. Some of the interactable objects within the game will be able to be used by the user, but not all of them will be necessary to solve the provided puzzles.

Knife & Bag Puzzle: The first puzzle of the game is that the user cannot pick up the knife unless they have the corresponding bag with magical runes. Once they have the knife in their hands, they can use that knife to cut through the bushes and gain access to the rest of the game.

Death: In this current version, there is no way for the user to 'die'. However, in the final version there will be at least 1 opportunity for failure, in that they can approach the 'Predator' too many times without 'Fish,' and the user could get eaten instead.

References

<http://www.cplusplus.com/doc/tutorial/files/>

https://www.tutorialspoint.com/cplusplus/cpp_files_streams.htm

<http://www.cplusplus.com/reference/string/string/>

<http://www.cplusplus.com/reference/map/map/>