Hackers Labyrinth

Requirements Document

By: Jasper Charlinski
Sebastien Van Den Bremt
Ethan Posner
Cassius Galdames

Date due: Oct 19th, 2022

Instructor: Dr. David Wessels

Table of Contents

1. Introduction

1.1 Description:

The project will be a labyrinth type puzzle game where the player progresses through a series of computer science related challenges. They will be doing this in the context of a story about the main character having to escape from a deranged captor. The player will have 3 attempts to complete each challenge, if successful the player will be rewarded with a clue that will help them escape. At the end of each challenge the player will have a choice on what room they would like to go to next (left or right door) which will affect the forthcoming challenges. On the final level of the game, the player will use the clues they have collected to complete the final challenge and escape the Hackers Labyrinth.

1.2 Scope

Hackers Labyrinth will be a multi-leveled game that will incorporate Computer Science principles into an interactive design.
Deliverables: The project will be a directory with a main executable. This directory will have subdirectories with files and other executables that will be called to, from the main.
Constraints: The system constraints for Hackers Labyrinth is that the game needs to be able to run on a VIU/CSCI Linux operating system.
Assumptions: We are assuming that the user can perform basic Unix commands as well as understands basic Computer Science principles.
Time Requirements: The project needs to be finalized and tested by December 9th.

1.3 Case Scenario

Steve is a new graduate from the Computer Science Diploma at Vancouver Island University and is on the hunt for his first software developer position. He is having some trouble reviewing all the material he has learned during his time at VIU, to prepare for the technical interviews he has lined up. He is searching for a fun interactive way to sharpen his problem solving skills, and review basic computer science principles. His friend has suggested he tries out Hacker's Labyrinth to help with the interview preparation.
He opens the application by running the executable for the game on his Linux machine, and this immediately resizes his terminal screen and the title "Hacker's Labyrinth" appears on the screen in ASCII Art. Then the opening menu pops up where Steve chooses the difficulty of the game, which he sets to easy and reads the game instructions. He then selects the start game option where he is transported into a dungeon where his character must solve a multitude of puzzles within a given time parameter or face sudden death. The first puzzle touches on encryption

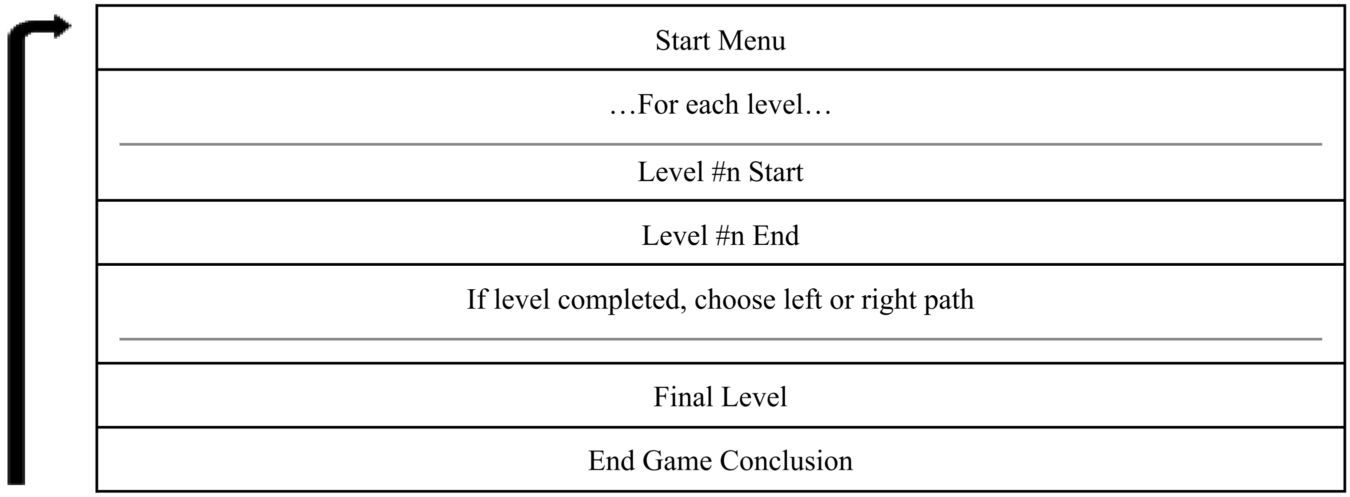
which he solves quickly with the use of the notepad and enters the code into the prompt. He gets a success message and is given the option to open one of two doors. He chooses the right door and is prompted with a “bonus 1 minute in next game” prize, then a new puzzle and the time remaining. The new level is a rising block puzzle where Steve must use his problem solving skills to solve it. He does so in 4 minutes after getting 1 answer wrong and using most of his allocated time. He gets another success message and is once again given the option to open one of two doors. He choses left this time and receives a bonus to the final level. He then has a remaining 3 minutes and 23 seconds to complete the final level. After entering two wrong answers, he runs out of time and fails the level. A message takes over the screen that reads “GAME OVER, YOU DIED” and the game closes itself. Steve realizes that he still has a lot of reviewing to do before his technical interview.

2. Requirements

2.1 Visual

2.1.1: Basic Layout

Hackers Labyrinth will be an immersive terminal window game, the basic layout of how the game will progress is as follows.



Start Menu

…For each level…

Level #n Start

Level #n End

If level completed, choose left or right path

Final Level

End Game Conclusion

Terminal setup

Upon the game being executed, a bash script will set up the initial terminal environment. Making the window full screen and changing the size of the text. Then it will display the start menu.

The terminal window will have a black background and green text for all normal scenarios. In certain cases the background color and text may change to better suit the scenario.

Start Menu

The start menu will be written in bash script, it will have options of

- Start game
- Difficulty
- Help (Controls)
- Exit game

When the user selects the start game, the bash script will call to the main cpp file.

2.1.2 Menu

Start Menu

The start menu will be the main bash function that will call the main cpp executable. It will be the first thing the player will see, hence it must communicate the theme of the game with as little detail as possible. The start menu will have the following options

- Start game (S)
  This selection will call the main cpp executable and which will then begin the first level.

- Difficulty (D)
  This option will allow the user to change the difficulty, the difficulty affects the size of the puzzles and, amount of time given to solve them, and the amount of attempts allowed per puzzle.

- Help (Controls) (H)
  This option will provide the player with a brief summary of the controls for the various actions.

- Exit game (Q)
  This option will deconstruct the game and exit the terminal window.

In-game Menu

Accessed by the player entering the command M
Options are navigated to by entering commands

The player will have access to an in-game menu at any time during the game, except during cutscenes. The in-game menu will consist of the following options

- Close menu (Q)
  This will close the menu and return to the game

- Open Notepad (N)
  This will open the in-game notepad feature, see 2.3.3 for more detail.

- Help (H)
  This option will provide the player with a brief summary of the controls for the various actions.

- Exit game (K)
  This option will deconstruct the game and return to the start menu.

The in-game menu will disrupt the current game and take over the screen, although the game timer will not pause while the player is in the menu.

2.1.3 Level Environment:

Each main level will have the same basic environment

Player will be displayed a question
Player provides answer (some questions may have different ways of providing answers)
System verifies answer
If correct, the player will have the choice to take the right path or left path. If incorrect, the player will be displayed and end game animation and return to the start menu.

The terminal setup described in 2.1.1 applies to each level.

2.2.3 Art

Level completion

Upon the completion of each level the player will be shown a relevant ASCII image or animation.
Each level will have unique art that relates to that particular level.

Path choice

After each successfully completed level, the player will have a choice of taking a right or left path. During this, the player will be shown an ASCII image of two doors.

End game

If the player fails a level, an ASCII animation will be shown to the player. This animation will be the same for every time the player fails.

Conclusion

When the player successfully completes the game, they will be shown an ASCII animation of the player escaping the Hackers Labyrinth and a short paragraph about the decisions they made along the way. The idea is that this conclusion paragraph may be slightly different for each playthrough of the game.

Audio

Levels may be accompanied by sound effects and music, but this decision is **not final.**

2.3 Levels of Play

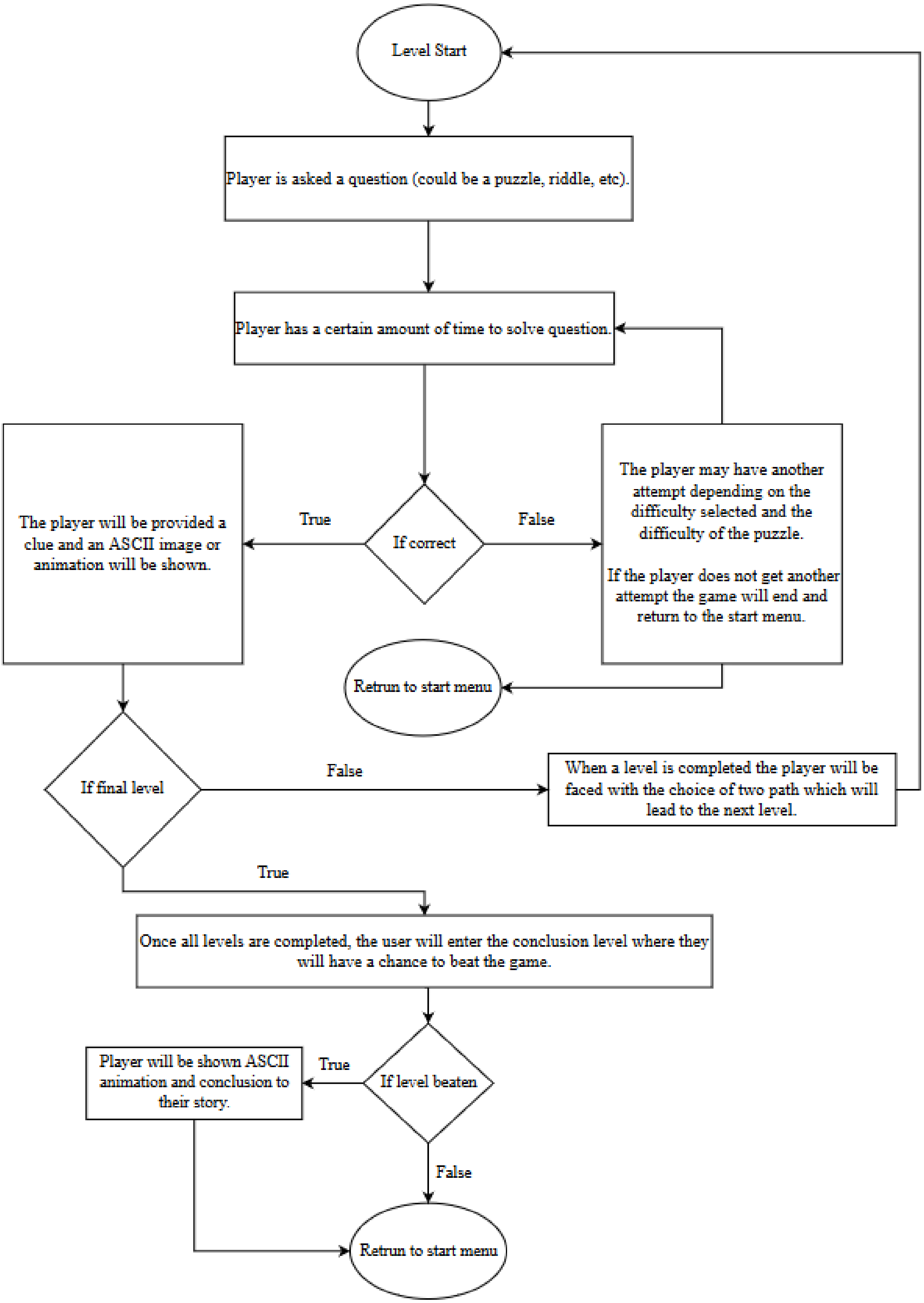
2.3.1 Level Conditions

Within each level, the player is first presented with a scenario for what they need to solve. They are required to input an answer within a given time limit.

At any time, the player can enter M to go to the menu screen. While in the menu screen, the game timer won't pause. The in-game menu is described in section 2.1.2.

The player will be given a certain number of retries and will be sent to the main menu if they run out of retries.

The process of completing levels is shown in the figure below.


Level Start
Player is asked a question (could be a puzzle, riddle, etc).
Player has a certain amount of time to solve question.
If correct
True
False
The player will be provided a clue and an ASCII image or animation will be shown.
The player may have another attempt depending on the difficulty selected and the difficulty of the puzzle.
If the player does not get another attempt the game will end and return to the start menu.
Retrun to start menu
If final level
False
When a level is completed the player will be faced with the choice of two path which will lead to the next level.
True
Once all levels are completed, the user will enter the conclusion level where they will have a chance to beat the game.
If level beaten
True
Player will be shown ASCII animation and conclusion to their story.
False
Retrun to start menu

2.3.2 Puzzles

Puzzles are not final and are subject to change.

1. Block the Rising Acid

The player is given the following prompt:

> You find yourself in a room with many blocks of different sizes. The blocks are labeled with numbers. It appears that the numbers indicate how many inches tall they are. You hear a voice on a loudspeaker:
>
> "In 2 minutes, acid will begin to rise. It will flood the whole room and dissolve everything in it unless you can find a way to block it from spreading throughout the whole room. Here's a hint: the acid will rise as high as your highest pair of two blocks stacked together."
> You notice that on the floor there seems to be a long slot across the whole room. The player sees a hole on one side of the slot. Perhaps this is where the acid comes from.
>
> The blocks are too skinny and unstable to stand on when stood up, so you won't be able to stand on them.
>
> You arrange the blocks and find that they have the following heights:
> <list of block heights>
>
> How high must you stack the blocks?

The player must then solve the puzzle by finding how high the blocks can be stacked such that a wall is constructed with uniform height. They must put stacks of two blocks together in the slots so that all stacks have the same height.
For example, if the heights are 4, 3, 2, 5, then we can stack blocks together as (4+3=7) and (2+5=7), which gives us a wall of height 7. The acid would then rise to a height of 7, and this would block it from filling the whole room. We could also have combined blocks as (4+2=6) and (3+5=8), but since the acid rises to the height of the highest stack, the room would be filled.

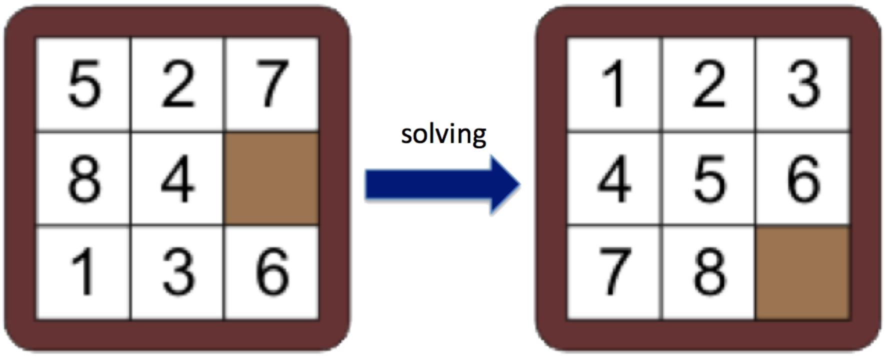
2. 15 Sliding Tiles

The player is given the following prompt:

You find yourself in a room with a large 4-by-4 grid containing 15 tiles labeled 1-15 in random order and one empty space mounted to a wall. On the opposite facing wall are mounted spikes. You hear a voice on the loudspeaker…

"You have 5 minutes to arrange the 15 tiles from low to high, begin"

The ground shakes and the walls begin to close in.
Example of 15 Sliding Tiles:



3. Decrypt Keeper

The player is given the following prompt:
"Welcome user, I hope you remember basic principles of message encryption."
“You will need to decrypt a message that will give you an equation, the result of this is the code you need to make it out of here. Here is the encrypted message:”
TNTEOYOVEYRWTISRTDIDTEEYMFTWEDBHE

“You will get up to three messages to help you crack this code, you will get their plain text version, followed by their encrypted one. For each message (hint) you request, 45 seconds will get trimmed off of your time. The first one is free, good luck player.”

GOODLUCKPLAYER
GDCLEOLKAROUPY

Help

CRACKTHECODE
CCHORKEDATCE

Help

SOLVETHISORDIE
SVHOIOEIRELTSD

Solve
TWENTYTIMESFORTYTWODEVIDEDBYTHREE
20X42/3
The code is: 280
Hint the encryption is a Transposition Cipher (permutation 2,3,4)

4. Cipher Escape

The player is given the following prompt:

You find yourself in a small, dark room. You cannot see anything. You hear a voice on the loudspeaker:

"You must use critical thinking skills and decoders to solve the riddles of this room. You will have 10 minutes to escape before the room fills with a poisonous gas. Enter in commands, such as "touch" or "look", to interact with the room, and decode messages to progress. Good luck player."

> touch/feel

You feel the wall in front of you. You can make out distinct bumps:

```
.      .      .       .     .    . .         . .     .     .       .     .    .     . .
.     .      .      . .      .     .                 .      .     .       .   .       .
.            .      .            .                   .     .      .           .     . .
```

Answer: listen closely

> listen/listen closely/hear

You focus your hearing. Suddenly, a faint tone can be heard beeping:

.-.. .. --. .... - / ... .-- .. - -.-. .... / .- -... --- ...- .

Answer: light switch above

> touch switch/flip switch/reach switch

You reach for the switch up high and manage to flip it on.
A dimly lit lightbulb emits an orange glow in the room.

> look/look around/see

You look around and see three sets of codes written on the wall to your left,
and a door with a PIN pad on the wall to your right. The codes are:

01100010 01100001 01110011 01100101 00110110 00110100

QVNDSUk=

83 73 88 57 84 87 79 52

Answer of Binary: Base64
Answer of Base64: ASCII
Answer of ASCII: SIX 9 TWO 4

> Use PIN pad/touch PIN pad/touch/feel

“A digital message on the PIN pad reads: "Enter a 4-digit number:"”

> 6924

With the correct 4-digit passcode inputted, the player completes the level.

Any incorrect commands will display the message: “[user input] is not recognized as a command”.
For the PIN pad, the error message “Incorrect passcode” will be displayed and the player will be required to interact with the PIN pad again to enter the passcode.

Conclusion Level: Regular Expressions

There are 3 ways of reaching the final level:

- Complete All Puzzles: If the player completes all puzzles, the final level door will automatically be unlocked.

- Find All Keys: Throughout the game, the player randomly receives keys by selecting one of two doors to open at the end of each game. Once they have collected three keys, they will be able to open the final level door.

- Instant Final Level: It is possible for the player to be instantly sent to the final level upon opening the final door.

The player will be asked to write a regular expression for either a phone number, an email address, an IP address, or a standard street address.

Upon reaching the final level, the player will be given the following prompt:
“You hear a voice on a loudspeaker: ‘Congratulations player! You have reached the final level. Now your final task is to write a regular expression to match the following: <insert type of string here>.’”

The player will then be prompted to enter their regular expression.

If the regular expression is correct, they will be shown ASCII art as a reward. As with other levels, they will be given three tries before receiving a game over.

2.3.3 Notepad

The Notepad is accessed from the in-game menu, see 2.1.2 for more detail, by entering the command **M** and navigated to with command **N**

The player will be able to type notes or clues into their notepad which they can then refer to in preceding levels.

When the player selects the notepad option, a Vim window will open and they can begin to type.

The player can type “:Q” to quit the notepad. Everything the user types is automatically saved when the notepad colses.

The player's notepad will be written to a premade .txt file which will be updated each time the user opens the notepad.

2.3.4 Path Choice Bonuses

Upon completion of each level, the user is presented with the option of opening one of two doors. These are presented as the “left” and “right” door. The choice of door will have unforeseeable short and long-term effects for the playthrough of the game. Only one bonus will be granted at a time.

Certain bonuses, such as time increases, will stack on previous ones. For example, if the player receives a permanent 1 minute time increase, then receives a 1 minute time increase in the following game which applies to the next game, they will now have a total of 2 extra minutes for completing the next level.

Instant Final Level

Rarity: 1/18

The player receives the following prompt:

+1 Minute in Next Game

Rarity: 1/3

The player receives the following prompt:
“A voice begins to play on the loudspeaker. It says ‘Congratulations on making it past that puzzle player. I was sure that one would do you in. No other victim finished that puzzle in quite the same way as you. Guess what else? By selecting the correct door, you have been rewarded with an extra minute of time in your next game. You’ll need it in this next one!’”

+1 Minute Permanent Time Limit Increase

Rarity: 1/6

The player receives the following prompt:

“The door you selected has a small note on the ground behind it. The note reads ‘Greetings player. Through sheer dumb luck, you have been given a permanent increase of 1 minute to each puzzle’s time from here on.’”

Receive One Key to Final Room

Rarity: 3/4

The player receives the following prompt:
“The door you selected has a key on the floor behind it. You pick up the key. It could come in useful later.”

Skip Next Level

Rarity: 1/12

The player receives the following prompts:
“The door you selected has a small coupon shaped paper on the food behind it. A voice begins to play on the loudspeaker. It says ‘Congratulations player! You have found one of the rarest items I have hidden, you may skip the next level and still receive the clue. Your life is spared… for now.’”