Nathan Balfour 2018 (N0784536)

Roguelike progression system write up

By Nathan Balfour

Contents

[Aims of the program 2](#_Toc532065849)

[Analysis 2](#_Toc532065850)

[Path of Exile 2](#_Toc532065851)

[Binding of Isaac 3](#_Toc532065852)

[Summary of findings 3](#_Toc532065853)

[Requirements of the program 4](#_Toc532065854)

[Design 5](#_Toc532065855)

[Input-Output Table 5](#_Toc532065856)

[Attributes, Skills, Attribute milestone bonuses and gifts from the gods 6](#_Toc532065857)

[User Interface 7](#_Toc532065858)

[Flow Chart 14](#_Toc532065859)

[Testing 15](#_Toc532065860)

[Summary 17](#_Toc532065861)

[Appendix 18](#_Toc532065862)

[Bibliography 26](#_Toc532065863)

# Aims of the program

I will be creating a progression system for the genre of roguelike games. The progression system in these types of games are quintessential. The program will aim to be the framework of a popular game genre roguelikes.

-It will include a way for users to make decisions on how to make their character stronger. These choices should be strategic in nature, the choice should not always be easy for the user.

-The system needs to be refreshing and interesting for users even after several times going through the same system. Needs to have a lot of variety.

-The user will also be able to save their unique characters for use later.

# Analysis

It is useful for us to look back at some existing and well received games that include roguelike aspects. From this we can see what the best formats and strategies are for progression in such games. There are many different styles of progression systems and some suit different game play styles better. As we are designing this system it will be important to keep in mind what kind of overall game we are creating for.

## Path of Exile



Figure 1 - Path of Exile Skill Tree Sourced from (Kain, 2012)

In figure 1 we can see the passive skill tree for the game Path of Exile (PoE). This game has rogue like elements in it such as procedurally generated levels as well as a popular mode known as “hardcore”. The aim of the mode is to progress through the same normal game except when your character dies it is deleted and all progression is lost (Path of Exile Wiki, 2018). This skill tree places the user at a point dependent on what character they choose. As they level up they gain points which allows them to unlock nodes along the lines. Each of them makes the character stronger and some have unique effects.

The progression tree stays the same each time so there is an element of strategy in which people will figure out certain paths to go that are the most efficient and powerful (Tahreyn, 2017). This level of branching and decision making for users is important to replay ability.

## Binding of Isaac

Binding of Isaac is a much closer to a traditional roguelike game. The aim of the game is to progress through procedurally generated levels, creating a strong character by picking up random items. This game has no level-based system in place and character strength is based purely on base attributes and the items gained through playing.



Figure 2 - Binding of Isaac's items. Sourced from (Nintendo Today, 2017)

The strategic aspect of the game that separates new players to long time ones is learning which items work well together. Although all the items are completely random in appearance, you still need to know if it’s a good idea to pick up an item when you see it. Items have positive and negative effects, so the choice is not always easy

## Summary of findings

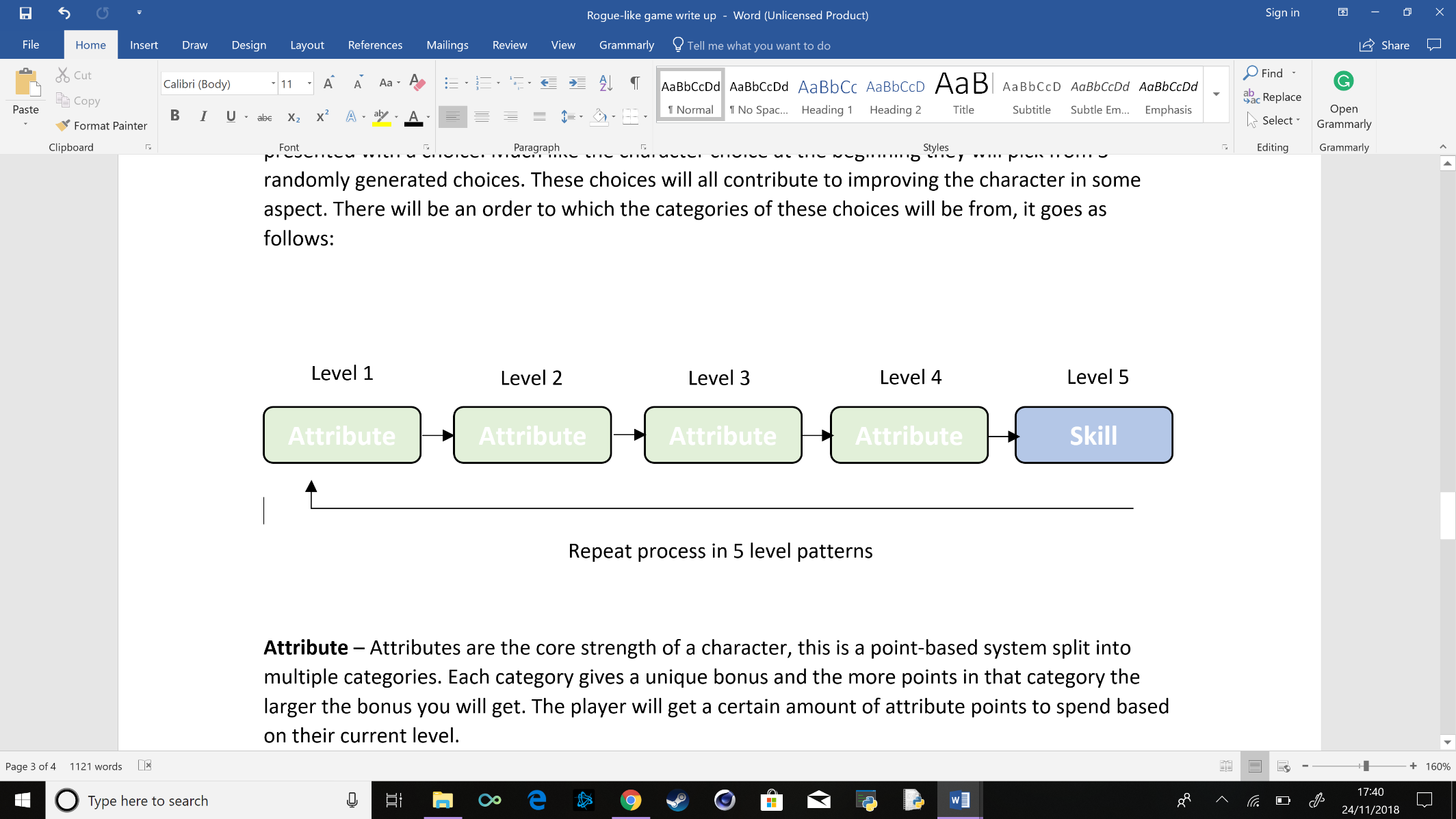
The most important aspects of a roguelike system are randomisation and permadeath (Mchugh, 2018). These games rely on replay ability, playing through from the start repeatedly. Randomisation is the key to keeping the game fresh and enjoyable over multiple restarts of the game. However, the game must be fair and balanced. To keep it fun for the user there must be a level of control to the progression so that the player can take advantage of tactics and skill rather than purely waiting on good luck. To do this we must make the system pseudorandom. The program must be created in such a way that certain aspects can be predicted and learnt such as an overlying pattern.

Progression in roguelike games is typically tied to a level system and items. As the user plays, they will earn something known as experience points (exp). This is earned through a variety of means, such as beating enemies and completing quests. When they get a certain amount of exp they will go up a level. Items will be randomly acquired during their time playing. These items come in a variety of types but they all do one thing, making the character stronger.

## Requirements of the program

The program will give the option for the user to pick a starting character. The starting character will determine the user’s base attributes. The user will have a choice of 3 characters, which have been selected from a larger pool of possible characters.

Next, the user will level up. As this is a system to be implemented into a larger program the levelling up is to simulate what would happen as the user plays. When the user levels up they will be presented with a choice. Much like the character choice at the beginning they will pick from 3 randomly generated choices. These choices will all contribute to improving the character in some aspect. There will be an order to which the categories of these choices will be from, it goes as follows:



**Attribute** – Attributes are the core strength of a character, this is a point-based system split into multiple categories. Each category gives a unique bonus and the more points in that category the larger the bonus you will get. The player will get a certain amount of attribute points to spend based on their current level.

**Skill** – skills are game changing abilities that the character gains, for example “you can now dodge an incoming attack”. In this program, skills will be purely a proof of concept andwon’t change anything.

When the player has acquired a certain number of attribute points in a single attribute they will unlock a bonus based on that attribute.

Each time the player levels up they will also have a small chance to gain a special bonus called “a gift from the god’s”. The chance will progressively get higher as they level up, they will also be limited to a maximum of 2 gifts per character. This will provide a bonus to the character making them stronger somehow. They will also have 3 choices for this as well.

The user will be able to do this up to a certain level. The higher they go the smaller the benefit they will gain from their choices. For example, attributes will scale with the level. As the player level increases, the less attribute points they will be able to assign to an attribute. This design choice is common in games as it makes early player experience more interesting.

When the player stops playing the progression needs to be saved for the next time the user wants to continue. The program will give the user the option to save the character’s name and progress, including all attributes and skills acquired.

# Design

## Input-Output Table

|  |  |  |  |
| --- | --- | --- | --- |
| Inputs | Processes | Storage | Outputs |
| Menu choices | Saving choice to a variable and limiting to only valid choices | Store in temporary variable for testing which choice they chose | Move forward to choice they made |
| Attribute/skill/gift from the god choice | Get 3 random attribute/skill/gift from the lists of values  Store choice to a variable  Test for attribute milestone reached | Store in temporary variable for testing which choice they chose  Store choice in list of acquired attribute/skill/gift | Confirm choice and tell them if they have reached an attribute milestone and what bonus they gained from it |
| Load character of choice | Test if there are any characters available to load  Read all character’s names from file into a list and give them a number for the user to enter as a choice  Read all stats of chosen character into respective lists | List of all character names in file  Storage of name and level of chosen character  List of all attributes and points  List of all skills | Confirm load of chosen character and move to character menu  If there are no available characters to load. Inform user and return to main menu |
| Save character | Write name, level and stats of character to a file | Pickle file containing all information of character | Confirm saved character and return to main menu |

## 

## Attributes, Skills, Attribute milestone bonuses and gifts from the gods

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Skills** | **Attribute milestones** | **Gifts from the gods** |
| Strength | Extended dodge | Strength 10 – gain 5 constitution points but lose 3 intelligence | Gain a 40% bonus to strength & constitution |
| Constitution | Heavy weight | Strength 25 – gain the berserker skill | Gain a 40% bonus to dexterity & luck |
| Dexterity | Light weight | Strength 50 – gain the duel wield skill | Gain a 40% bonus to intelligence & luck |
| intelligence | shorty | Constitution 10 – gain passive healing skill | Gain 10 strength |
| luck | Sure guard | Constitution 25 – gain 30% constitution but lose 5 luck points | Gain 10 constitution |
|  | charge | Constitution 50 – gain stable skill | Gain 10 dexterity |
|  | focus | Dexterity 10 – gain 2 points of constitution and luck | Gain 10 intelligence |
|  | Lucky shot | Dexterity 25 – gain Double dodge skill | Gain 10 luck |
|  | Hot stuff | Dexterity 50 – gain multishot skill | Gain 3 random skills but lose 2 you currently have |
|  | Ice cold | Intelligence 10 – gain 10 luck but lose 40% dexterity and constitution |  |
|  | confusion | Intelligence 25 – gain blast zone skill |  |
|  | sprint | Intelligence 50 – gain reflect skill |  |
|  | thorny | Luck 10 – gain By a hair skill |  |
|  | sharpshooter | Luck 25 – gain volatile skill |  |
|  | Scream, aim and fire | Luck 50 – gain now you see me skill |  |
|  | dirty |  |  |
|  | Squeaky clean |  |  |
|  | Dazzling smile |  |  |
|  | Dark sight |  |  |
|  | spectral |  |  |
|  | Natures call |  |  |
|  | lash back |  |  |
|  | Money grabber |  |  |

## User Interface

It is important to get the formatting of text correct when using a text based UI. The formatting/spacing needs to be clear between steps and the choices that the user must make need to be clear and easy to understand. For example, using numbered choices and stating what to input at each choice. Each menu will also contain a help option so that you understand what each choice does.

Main menu and tooltips

Hello and welcome to the roguelike progression system. This program is designed to demonstrate a functioning progression system housed in a roguelike game.

The aim of the program is to create a character which will progress and get stronger by levelling up. The level up mechanic in this is to simulate playing through a game and gaining experience points for completing various tasks.

Each time you choose to level up you will be given a selection of either attributes or skills that will benefit your character. These choices will stick with your character permanently but do not fret if you make a bad choice! You can always make a new character!

Below is information on each of the options you will be able to pick in this program

New Character

This option will allow you to create a character and progress through the rest of the program.

Load Character

If you have already made a character and saved it you can load that character with this option and progress where you left off

Hello and welcome to the roguelike progression system. Below is the main menu, please make a choice to continue.

1. New Character

2. Load Character

3. Help

4. Quit

Help

This will be available at almost all menus throughout the program. If you ever get stuck just pick that and you will be given help specific to the part of the program you’re in. If you ever want an overview of everything in the program visit here.

Quit

Stops the program and loses any progress made

Save and Quit

Gives you the option to save your character and then quit to the main menu

Level up

Simulates the character getting enough experience points (exp) to go to the next level and get stronger

View character

Gives an overview of your character including its name, level, attributes and skills

Key terms

Level

This is a number given based on how many times you have surpassed the amount of experience points needed. This is a gauge of how much you have played the game and how strong you are. In this program there is no exp system as no gameplay is available. However, the point is that this can be easily implemented into a full game and use experience points

Attributes

Attributes are the core strength of a character, this is a point-based system split into multiple categories. Each category gives a unique bonus and the more points in that category the larger the bonus you will get. You will get a certain amount of attribute points to spend based on their current level. When you get a certain amount of points in an attribute you will reach an attribute milestone and get a special bonus. These points will not affect any gameplay in this program, they are purely a demonstration of how the system works

Skills

Skills are game changing abilities that the character gains, for example “you can now dodge an incoming attack”. In this program, skills will be purely a proof of concept and won’t change anything.

Gift from the gods

Gift’s from the gods are special bonuses granted rarely when levelling up. If you gain a gift from the gods you will be presented 3 choices of bonuses that will strengthen your character in various ways, like picking skills and attributes.

Character menu

What would you like to do?

1. Level up

2. View Character

3. Help

4. Save and Quit

Character menu tooltips

This is the character menu

This is where you manage your character by making them stronger, looking at their stats or saving them!

Menu options

Level up

Simulates the character getting enough experience points (exp) to go to the next level and get stronger

View character

Gives an overview of your character including its name, level, attributes and skills

Save and Quit

Gives you the option to save your character and then quit to the main menu

Key terms

Level

This is a number given based on how many times you have surpassed the amount of experience points needed. This is a gauge of how much you have played the game and how strong you are. In this program there is no exp system as no gameplay is available. However, the point is that this can be easily implemented into a full game and use experience points

Attributes

Attributes are the core strength of a character, this is a point-based system split into multiple categories. Each category gives a unique bonus and the more points in that category the larger the bonus you will get. You will get a certain amount of attribute points to spend based on their current level. When you get a certain amount of points in an attribute you will reach an attribute milestone and get a special bonus. These points will not affect any gameplay in this program, they are purely a demonstration of how the system works

Skills

Skills are game changing abilities that the character gains, for example “you can now dodge an incoming attack”. In this program, skills will be purely a proof of concept and won’t change anything.

Gift from the gods

Gift’s from the gods are special bonuses granted rarely when levelling up. If you gain a gift from the gods you will be presented 3 choices of bonuses that will strengthen your character in various ways, like picking skills and attributes.

Level up choices

Da ding!

You levelled up, please choose one of the following

1. Skill/Attribute 1

2. Skill/Attribute 1

3. Skill/Attribute 1

4. Help

User chooses

You have gained [chosen skill/attribute]!

Character load menu

Please select a character to load

1. Character name 1

2. Character name 2

3. Character name 3

…

or

Oops! Looks like you haven’t got any characters to load yet

Start playing and save a character to load it here next time you play!

New character menu

Your adventure begins here!

Please select a class from the options below. This will determine your base stats and skills

1. Class 1

Attributes

Skills

2. Class 2

Attributes

Skills

3. Class 3

Attributes

Skills

A fine choice indeed! Please choose a name for your character:

Save menu

Would you like to save? (y/n)

Y

Saving [character name]…

Done!

Returning to the main menu

or

Would you like to save? (y/n)

n

Returning to the main menu

Attribute milestone reached

Oh! You’ve reached an attribute milestone for [attribute]!

You have gained [attribute milestone bonus name] gaining [bonus stats]!

Gift from the gods

…

Hm?

You feel something strange

…

You’ve received a gift from the gods!

Please choose the bonus you would like to receive

1. gift 1

2. gift 2

3. gift 3

User chooses

You have gained [gift]!

## Flow Chart

# Testing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Number | Description | Data Type | Expected Result | Pass/Fail | Cross Reference |
| 1 | Test main menu based on input given | valid | Moves forward to selected part of the menu | pass | [See appendix](#_Appendix) |
| 2 | Test main menu based on input given | Valid extreme | Moves forward to selected part of the menu | pass | [See appendix](#_Appendix) |
| 3 | Test main menu based on input given | Invalid | Will say that the input is not valid and ask for input again | pass | [See appendix](#_Appendix) |
| 4 | Test main menu based on input given | Erroneous | Will say that the input is not valid and ask for input again | pass | [See appendix](#_Appendix) |
| 5 | Test character menu based on input given | valid | Moves forward with selected part of the menu | pass | [See appendix](#_Appendix) |
| 6 | Test character menu based on input given | Valid extreme | Moves forward with selected part of the menu | pass | [See appendix](#_Appendix) |
| 7 | Test character menu based on input given | invalid | Will say that the input is not valid and ask for input again | pass | [See appendix](#_Appendix) |
| 8 | Test character menu based on input given | erroneous | Will say that the input is not valid and ask for input again | pass | [See appendix](#_Appendix) |
| 9 | Test load character menu based on input given | valid | Loads character selected or goes back when selecting appropriate option | pass | [See appendix](#_Appendix) |
| 10 | Test load character menu based on input given | invalid | Will say that the input is not valid and ask for input again | pass | [See appendix](#_Appendix) |
| 11 | Test load character menu based on input given | Erroneous | Will say that the input is not valid and ask for input again | fail | [See appendix](#_Appendix) |
| 12 | Test load character menu when there is no character to load | invalid | Will inform user that there is no saves available and continue back to the main menu | fail | [See appendix](#_Appendix) |
| 13 | Test saving and quitting a character and loading another character in the same session | valid | Will assign character stats to newly loaded character | pass | [See appendix](#_Appendix) |
| 14 | Test saving and quitting a character then making a new character | valid | Will assign character stats to the newly selected character | pass | [See appendix](#_Appendix) |
| 15 | Test saving a character with a name that already exists | valid | Will overwrite original file | pass | [See appendix](#_Appendix) |
| 16 | Test attributes being changed when levelling up | valid | Will alter attributes based on choice | pass | [See appendix](#_Appendix) |
| 17 | Test skills being added when levelling up | valid | Will add skills to character stats based on choice | pass | [See appendix](#_Appendix) |
| 18 | Test attribute milestones applying bonuses when reached | valid | Will automatically apply bonuses when each milestone is reached | pass | [See appendix](#_Appendix) |
| 19 | Test gift from the gods triggers properly and applies bonuses | valid | Will trigger a new menu with a selection of bonuses and will apply the bonus based on choice | pass | [See appendix](#_Appendix) |
| 20 | Test trying to level up when at max level | invalid | Will inform player that they are max level and go back to the menu | pass | [See appendix](#_Appendix) |

# Summary

Overall the program performs well. There is a large amount of variety already to each character and that can be increased easily with a longer list of skills, attributes and characters. The small chance to get a bonus that increases over time is fun and rewarding.

There are still occasional bugs with the program that I cannot pinpoint such as some milestones causing errors. The system in place for the milestones is very inefficient and would slow down the program a lot if there was a longer list of them. There was a problem with pressing enter on some choices and it would take it as an empty input. To fix this I have added a test for empty inputs so that It will take it as an invalid input and make the user choose again.

There were some issues with file handling. I bounced between specifying the whole project directory and just the local file I was accessing. The local file directory seems to work the best on different devices, but I still have had some occasional issues. I would say that to remove the problem completely I should use Python 3’s pathlib to automatically build the path for me and deal with the OS specific details. I couldn’t easily switch to using this in my code as I am using os.listdir which creates a list of all the possible characters to load.

A lot of the tests that run throughout the program are just long lists of if statements which works in its current state but as the program would get larger the slow down would be too significant. If this was also implemented into a real time roguelike game, testing for skills being there would be incredibly slow. The only solution to this I can think of in python would be to load them all into memory constantly so they are quick to access. However, this is not a good programming technique. The only other case would be to move over to a different language which encompasses this better.

# 

# Appendix

Tests

|  |  |
| --- | --- |
| Test Number | Screenshot |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 9 |  |
| 10 |  |
| 11 | Prints error when typing a string as the choice |
| 12 |  |
| 13 |  |
| 14 |  |
| 15 |  |
| 16 |  |
| 17 |  |
| 18 |  |
| 19 |  |
| 20 |  |

# Bibliography

Kain, E., 2012. *This Is Path Of Exile's Insanely Awesome Passive Skill Tree.* [Online]   
Available at: https://www.forbes.com/sites/erikkain/2012/08/14/this-is-path-of-exiles-insanely-awesome-skill-tree/  
[Accessed 24 November 2018].

Mchugh, A., 2018. *What is a roguelike?.* [Online]   
Available at: https://www.greenmangaming.com/blog/what-is-a-roguelike/  
[Accessed 24 November 2018].

Nintendo Today, 2017. *7 tips and tricks to help you not suck at the Binding of Isaac: Afterbirth.* [Online]   
Available at: http://nintendotoday.com/binding-of-isaac-afterbirth-tips-switch/  
[Accessed 24 November 2018].

Path of Exile Wiki, 2018. *League.* [Online]   
Available at: https://pathofexile.gamepedia.com/League  
[Accessed 24 Novemeber 2018].

Tahreyn, 2017. *[3.1] SRS LL Auramancer in-depth guide.* [Online]   
Available at: http://www.pathofexile.com/forum/view-thread/1816896  
[Accessed 24 November 2018].