Dungeon Crawler 2.0

By Theo Joyce

Mr. R. Kedwell

ICS3U

15/06/2016

USB

[Table of Contents](#ToC)

Table of Contents

1. [Step 1 – Defining the problem](#Step1)
2. [Step 2 – Analysis and Design](#Step2)
3. [References](#Reference)
4. [Work Log](#WorkLog)

[Table of Contents](#ToC)

ICS3U – FINAL PROJECT  
STEP 1 – DEFINING THE PROBLEM

***Type of Application*** (i.e. game, text editor, trivia, music app, other)

The type of this application is a game.

# ***Detailed Application Description***

This game is in the Dungeon RPG genre. You, the player, create a simple character, and start off immediately in your first dungeon at level 1. You progress through the game by completing dungeons; a bunch of rooms with monsters you must fight. Upon defeating a monster, you get some gold and experience that goes towards your level. The higher your level the stronger you are, which is important for when you progress to more difficult dungeons, where the monsters may be too much for you if you are a low level. The goal of the game is to complete all the dungeons, but it can still be played after completing this goal.

***Programming Knowledge Required*** (i.e. loops, procedures, etc…)

Explanation: List the programming concept and explain how you plan to use it in your application (i.e. Sorts - to sort the high scores for my application)

Sorts – Will sort the player’s save files alphabetically

Procedures – Will handle things such as collision detection and attacking

Functions – Used for matching an item index to its item, and the reverse.

Structures – Used for enemy profiles, player information, etc.

Variables – Used for storing all of the information needed for the game, such as integers for health, images for the graphics, Boolean for determining if an enemy has been defeated, strings for the player’s name along with some other names such as an enemy’s name.

If Statements – Used for determining things such as what item the player has selected to equip.

Select Case Statements – Used for the movement system and item index system

File Read/Writing – Used for saving and loading player information

Module – Used for storing all cross-form information such as graphics and player information

Buttons – Used for opening menus and selecting attack moves

Radio Buttons – Used for selecting your class in character creation

Counters – Used for determining if an animation should still be playing, and how many runs in a procedure has been done.

Check Boxes – Used for selecting what to buy at a shop

For Loop – Used for inputting inventory information from save file to the game

Array – Used for handling save information

Dynamic Array – Used for handling inventory information

Timer – Used for animation and timed events (such as progressing to the next dungeon room)

Sounds – Used for background music and player sounds (such as attacking)

\*\*Submit this completed form to the drop box on eLearning “Step 1 – Problem Definition” \*\*

[Table of Contents](#ToC)

ICS3U – FINAL PROJECT  
STEP 2 – ANALYSIS AND DESIGN

**APPLICATION SUB PROGRAMS**

1. List the various subprograms that you will create in your project.
2. Outline the purpose of each of the functions & procedures you mentioned from the previous point.
3. Remember that you are NOT writing computer code during this phase.

**DEVELOPMENT PLAN**

Outline the order that the functions and procedures will be created and a time-line for completion. List when you would like particular portions of your application complete.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SUN** | **MON** | **TUE** | **WED** | **THUR** | **FRI** | **SAT** |
| 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| 22 | 23 | 24 | 25 | 26 | 27 | 28 |
| 29 | 30 | 31  Finish inventory system | 1  **Step 1 Due** | 2 | 3  **Step 2 Due**  **Work Log 1**  Finish battle system | 4 |
| 5 | 6  **P.D. Day** | 7  Finish player classes and dungeon creation | 8 | 9 | 10  **Work Log 2** | 11  Finish adding all the dungeons, rooms, and enemy profiles |
| 12  Create a simple dungeon selection screen | 13 | 14  Finish adding all items into the game | 15  **Final Project Due** | 16 |  |  |

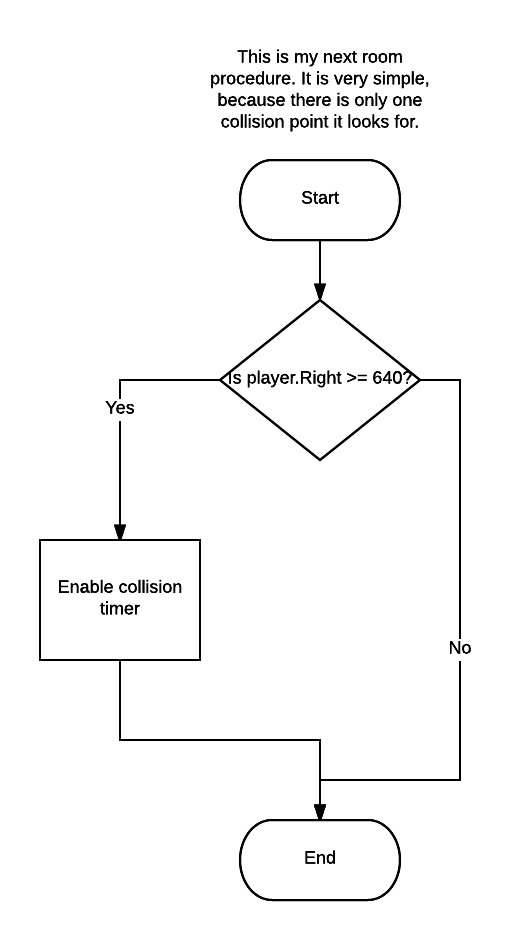
**PROBLEM SOLVING TOOLS**

1. Complete at least 2 of the following 3 problem solving tools.
   1. Create an IPO chart for one of your function or general sub procedures.
   2. Create a Flowchart for one of your function or general sub procedures.
   3. Create a solution for a function or procedure via an algorithm layout.

IPO - itemIndex

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
| itemID (The number representing an item) | Using a case statement, go down the list of case is = 1 , 2 etc, until it reaches the item ID that matches, and then return a string value with the matching name. | String with the name of the item |

Flowchart – nxtRoom



Submit this completed MS Word document on the D2L website to the drop box labelled “**Step 2 - Analysis and Design**” in the Final Project folder. This needs to be submitted, prior to moving on to Step 3.

Sites Used as Reference

**Copy and paste the URL for all websites outside of the course website that you used in the creation of your final project. Submitting work that you do not give credit to the resource can result in loss of marks or a zero on your final project.**

1. <bravefrontierai.imgur.com> - Art
2. <https://msdn.microsoft.com/> - Concepts , CType and WMP integration
3. <https://www.freesound.org/> - Sounds

[Table of Contents](#ToC)

Work Log

<https://github.com/coolomancp/Final-Project/commits/master>