**Lovestuck Academy Documentation**

By Nicholas Chua Jay Ngan year 13 BCZR

Alpha 2 footage https://www.youtube.com/watch?v=0D1orIK3Clw

**3DT Programming - Assessment Schedule**

[**3.7 Use complex techniques to develop a computer program**](https://www.nzqa.govt.nz/nqfdocs/ncea-resource/achievements/2019/as91906.pdf)

|  |  |  |  |
| --- | --- | --- | --- |
| **Achievement Standard** | **Achievement** | **Achievement with Merit** | **Achievement with Excellence** |
| **3.7**  **(6 credits)** | Use complex programming techniques to develop a computer program. | Use complex programming techniques to develop an informed computer program. | Use complex programming techniques to develop a refined computer program. |

|  |  |  |
| --- | --- | --- |
| **Achieved** | ✓ | **Comments** |
| Program performs specified task and is fit-for-purpose |  |  |
| Two or more data types used |  |  |
| Sequence, selection, and iteration control structures used |  |  |
| Takes input from a user, sensors, or other external source, and produces output |  |  |
| Two or more [complex techniques](https://docs.google.com/document/d/1XOWWl3oNlLpHKfWV8Ju11-OAFPSamQcyYxS6kLeCjXc/preview#bookmark=id.q6o8s314dgbm) used |  |  |
| Code is set out clearly and documented with comments |  |  |
| Evidence of testing and debugging on a sample of expected cases |  |  |
| **Merit** |  |  |
| Program is documented with appropriate variable/module names and comments that describe code function and behaviour |  |  |
| [Common conventions for chosen language](https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/inside-a-program/coding-conventions) followed |  |  |
| Evidence of effective testing and debugging the program on a sample of both expected cases and relevant boundary cases |  |  |
| **Excellence** |  |  |
| Program is a well-structured, logical response to task |  |  |
| Program is [flexible and robust](https://docs.google.com/document/d/1hehqsYjgKPZq_ww4dssYpxNuUU2zgn9ZgZTdjcFv8ME/preview#bookmark=id.jrlz5jzhrw40) |  |  |
| Evidence of comprehensive testing and debugging |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Overall Grade:** | **Achieved** |  | **Merit** |  | **Excellence** |  |

Use complex programming techniques to develop a computer program involves:

● writing code for a program that performs a specified task

● using complex techniques in a suitable programming language

● setting out the program code clearly and documenting the program with comments

● testing and debugging the program to ensure that it works on a sample of

expected cases.

Use complex programming techniques to develop an informed computer program involves:

● documenting the program with appropriate variable/module names and organised comments that describe code function and behaviour

● following conventions for the chosen programming language

● testing and debugging the program in an organised way to ensure that it works on a sample of both expected cases and relevant boundary cases.

Use complex programming techniques to develop a refined computer program

involves:

● ensuring that the program is a well-structured, logical response to the task

● making the program flexible and robust

● comprehensively testing and debugging the program.

The programming language chosen must support the required data types, control structures, complex programming techniques, and have good commenting facilities.

A complex computer program:

● uses variables storing at least two types of data (e.g. numeric, text, Boolean, object)

● uses sequence, selection and iteration control structures

● takes input from a user, file, sensors, or other external source

● produces output

● uses two or more complex programming techniques.

Examples of complex programming techniques include:

● programming or writing code for a graphical user interface (GUI)

● reading from, or writing to, files or other persistent storage

● object-oriented programming using class(es) and objects defined by the student

● using types defined by the student

● using third party or non-core API, library or framework

● using complex data structures (e.g. stacks, queues, trees).

Example of ways of making a program flexible and robust include:

● using actions, conditions, control structures and, methods, functions or procedures effectively

● checking input data for validity

● correctly handling expected, boundary and invalid cases

● using constants, variables and derived values in place of literals

**Initial Aim**

Create a game in the genre of visual novel and dating simulator.

3 Heroines with linear and separate stories that don’t impact each other.

Simple and limited controls using number keys.

**Target Audience**

Lonely gamers and anime enthusiasts who have reasonable knowledge in games/otaku culture.

**Tests**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Expected Tests** | | | | |
| Area | Testing | Expectation | Problem/Bug | Solution |
| Menu | Confirm Newgame(save exists) | Opens the confirmation screen when there isn’t a save file |  |  |
| Text Display | Going to the next page by pressing 1 | The next page of text displays |  |  |
| Choice Event | Choosing given choices | The selected choice loading its corresponding event. |  |  |
| “Introduction” Scene | Navigation | By pressing 1 or 2 the user can progress through the scene until the end where it loads the “FirstDay” scene. | Does not load the “FirstDay” scene at the end. | Delete an unnecessary method that caused an issue which stops the scene from loading. |
| “First Day” Scene | Playing through a scene | Scene progresses to a choice event and then ends after the selected event runs its course | Text is skipped | Set the choice event to trigger at the right page. |
|  | Playing through a scene | Scene progresses to a choice event and then ends after the selected event runs its course |  |  |
| “First Day” Scene | Playing through a scene. | Changes scene when the player reaches the end of the scene. | "index was outside of bounds of array" error appears | Write code preventing the "index was outside of bounds of array" error |
| Code preventing "index was outside of bounds of array" error | Getting to the end of the text array and going one step further. | No error message occurs and scene changes. | Error message still appears. | Add -1 to the condition codes to account for the 0 elements. |
| “First Day” Scene | Choosing the 4th option. | Goes into the 4th event. | Goes into the 1st event | Set the element where the event trigger happens to element 10 and set a delay to prevent a key press from triggering twice. |
| “First Day” Scene | Choosing the 4th option. | Goes into the 4th event. | Goes into the 1st event | Rework the system: eliminating bools and changing the method to work with array length instead. |
| “First Day” Scene | Choosing the 4th option. | Goes into the 4th event. |  |  |
| Homeroom | Choice text display. | Text changes based on bools in “Save” | NullReferenceException object not set to an instance of an object.  Nothing displayed. | Fix a reference pathway error. |
| Homeroom | Choice text display. | Text changes based on bools in “Save” |  |  |
| Homeroom | 4) Keep to yourself | Goes into the classes event | Nothing happens | Write code for the option. |
| Homeroom | 4) Keep to yourself | Goes into the classes event |  |  |
| Image System Text | Progress through a scene. | Images change as the page number reaches the set points. | NullReferenceException object not set to an instance of an object.  All images are displayed at once | Fixed and added references to the correct places. |
| Image System Text | Progress through a scene. | Images change as the page number reaches the set points. |  |  |
| Image System Text | Progress through a scene | Images are displayed and change when they are set to. | Images cover the text box. | Make the image system a child of the text system so it no longer covers it. |
| Classes | Enter the “Classes” scene. | A random remark is chosen from the array and displayed. |  |  |
| Classes | Go to the next scene | Goes the next scene which is determined by the “lunchCount” int from the save file. |  |  |
| Image System Choice | Choose events | Loads the correct images for each event. | Starting event images do not disappear | Make it so that starting events disappear when the first chosen event image loads |
| Image System Choice | Choose events | Loads the correct images for each event. |  |  |
| Elevator | Press 2 | Loads the “Club1st” scene | Loads “ShiroMeet” scene instead. | Change the name of the scene loaded from “ShiroMeet” to “Club1st” |
| Elevator | Press 1 | Loads the “Home” scene | Doesn’t load it. | Write the code that wasn’t previously there to load the “Home” scene. |
| “EloraMeet” | Progress through the scene, choosing the 2nd choice. | Loads the correct text from the 2nd array and displays the images corresponding to it. | Can’t end the scene as the text is stuck on the 2nd last page. | Fix an if condition that works to stop errors for the choice event but stops the scene from ending. |
| Classes(updated version) | Press space | Chooses a random page from the array. | Nothing happens | Added a missing “page =” in the code. |
| Save | Exit to the menu and use continue. | Goes to the beginning of the last scene you were on. | Goes to the “Introduction” scene. | Have a different script execute the method to load the last scene. Save the name of the scene constantly to avoid and Unity mishaps. |
| Save | Exit to the menu and use continue. | Goes to the beginning of the last scene you were on. |  |  |
| Affection System | Go through a scene that adds affection points at the end. | Affection points are added and the value carries over to the next scene. | Affection points and other save data does not carry over. | Set the correct path for a script when accessing the save file. |
| Affection System | Go through a scene that adds affection points at the end. |  |  |  |
| Continue | Exit a scene to the menu, go to either “Lore” or “Affection”, return to the menu and return to the scene by going to continue. | Loads the menu then the desired scene, then the menu and back to the scene exited in the first place. | Does not load the scene exited in the first place. | Put a condition where the scene name is not recorded when the scene is named “Menu”, “Lore” or “Affection” |
| Continue | Exit a scene to the menu, go to either “Lore” or “Affection”, return to the menu and return to the scene by going to continue. | Loads the menu then the desired scene, then the menu and back to the scene exited in the first place. |  |  |
| Save | Go to a one-off scene | Save file records the visit to the one-off scene | Save file does not record the visit to the one-off scene. | Write the code that was missing to actually record the visit. |
| Save | Go to a one-off scene | Save file records the visit to the one-off scene |  |  |
| Classes | Progress through the scene. | Loads a lunch scene based on the “lunchCount” int in save at the end of the scene. | Does not load any scene. | Fixed mistakes with the dictionary values used to load the scene. |
| Classes | Progress through the scene. | Loads a lunch scene based on the “lunchCount” int in save at the end of the scene. |  |  |
| Lunch | Go through the “Classes” scene multiple times. | A new lunch scene is loaded each time. |  |  |
| Elevator | Visit the “ShiroMeet” scene and reach the “Elevator” scene. | “Talk to the girl with cat ears” becomes “2)Club” |  |  |
| Elevator | Press two after unlocking “2)Club” | Goes to the Club1st scene. |  |  |
| Classes | Press space | A random page is chosen. | Nothing happens | Fix the max and min values the random integer could take. |
| Classes | Press space | A random page is chosen. |  |  |
| **Boundary Tests** | | | | |
| Choice Event | Selecting the highest number choice | It loads the corresponding event. |  |  |
| Image System Text | Set certain images to trigger at a page number is less than 0 so that they don’t appear. | The images not supposed to appear don’t and the scene progress normally. |  |  |
| Image System Text | Set certain images to trigger at a page bigger than the maximum page number. | The images not supposed to appear don’t and the scene progress normally |  |  |
| Pressing Space | Hold the space bar | Goes to the next page once |  |  |
| “lunchCount” reset | End the “Classes” scene 8 times. | The next lunch scene will be the second lunch scene again as there are no more new lunch scenes. |  |  |
| **Unexpected Tests** | | | | |
| Introduction | Press one and two and the same time | Scene progresses normally |  |  |
| Choice Event | Trying to select choices not given by pressing 3 and 4 when there are only 2 choices | Nothing happens |  |  |
| Image System Text | Set all images to trigger at the same scene | The only images that appear are “Back3” and “Girl6” |  |  |
| Image System Text | Set images designed to appear later on first and the first images to appear later | Later images do not disappear when the first images appear. |  |  |
| Image System Text | Go through a scene normally | The correct images load. | Text boxes and other UI are covered by the image system canvas. | Make the image system a child of the main UI canvas. |
| Affection Point System | Repeatedly exiting and entering a scene which starts it all over from the beginning. | This allows the player to repeatedly farm affection points as affection points are given at the start of the scene for text only events and when a choice is made for choice events. |  | Set it so that affection points are only awarded just as the scene changes |
| Pressing space | Press the space bar rapidly | Goes to the next page multiple times |  |  |
| Pressing space + choice | Press the space bar during a choice event | Nothing happens |  |  |
| Lore | Press one and two at the same time | The single key press registered by the computer first carries out the corresponding method or nothing happens as both key presses are registered |  |  |
| Elevator | Choose unavailable options.  (Options are available after talking to heroines.) | Nothing happens. |  |  |
| Affection Viewing Scene | Press random keys | Nothing happens |  |  |
| Affection Viewing Scene | Click random things | Nothing happens |  |  |

**Credits**

Programming and Sprites by Nicholas Chua

Referenced code from Josh Browne and Aidan Diprose.

The free for commercial use font was ‘**7:12 Serif Font’** by Christian Munk <https://christianmunk.1001fonts.com/>