**Halloween Project**

Before you begin, take the time to read and reread the directions for this project. It’s important to understand the rules before you jump in. Any misunderstandings can be addressed early, not on due dates. Following directions is paramount to success in anything you do.

You will design and program, which is a game, that allows the user to enter no less than nine (9) rooms in search of a bag or bags of Halloween candy, or some sort of treasure, or picking up pieces to complete a puzzle. Use your imagination and be creative.

The player must be met with at least one obstacle per room in their search for the candy/treasure/puzzle pieces. The bag of candy can be hidden or out in the open, it’s placement is up to you. There can be more than one bag of candy. When the player collects all the candy, they win the game and the game is over.

The game can be over by the player going into the wrong room and facing an obstacle such as a ghost that terminates the player and the game. In which case, the game can be restarted and played again by the user, this time the player has new information that will help him or her play better. A timer or some kind of score keeping device will be something extra to add pressure and the feel of competition for the player.

You will be working on the project with a partner. The project completion date is November 2nd,2017. Presentations will begin November 3rd. Your presentation of your game is limited to 3 minutes. Refer to the rubric grading chart below for more specific details on how grading will be done.

The project must include everything you have learned up to this point. Forms, Checkbox, Label, PictureBox, ComboBox, TextBox, CheckedListBox, Button, GroupBox, If Then Else, Random Number Generator, Constants, Variables, Data Types, MessageBox, InputBox and anything else not mentioned in this rubric. Projects that include animation and sound will be given extra credit as we haven’t done these yet in class and if you include them it demonstrates a level of knowledge above expectation.

Your mission is to create a multiform game. Each room should be its own form. You can also have sub rooms off those main rooms, for instance a closet in a bedroom, a butler pantry off the kitchen or an upstairs bathroom of the master bedroom. The bag of candy must be challenging to find, and this can include putting a puzzle together, solving a riddle or something along those lines.

**The suggested rooms are:**

1. Foyer
2. Kitchen
3. Den
4. Downstairs Hallway
5. Upstairs Hallway
6. Master Bedroom
7. Childs Bedroom
8. Basement
9. Attic
10. Garage (optional)

You have total control over the design and content. Utilize all that you currently know and feel free to use the sample programs that we have done in class to guide you through the process.

**GRADE PROCESS:**

There will be three major grades for this project; all two will be the project type grades in PowerSchool.

The first grade is based on your programs technical merits. It should include all the objects discussed in the rubric, and how they work. Even if some of the code doesn’t work, I’ll take into consideration what the coders were trying to accomplish and how the coders explained the problem during their presentation and make allowances.

The second grade is based on your progress through the project. I’ll be checking each day for progress. As a rule, you should have the framework for at least two rooms done per day.

The third grade will be based on your presentation of the project to the class. Take time to rehearse what you and your partner are going to present and how you will say it. As you demo your game, I expect you to talk about the logic that you built into the code and why you decided to take that approach. It is also possible to recognize that another approach would have been better.

**RUBRIC**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 |
| Problem Solving  **5%** of Coding Grade | Little or no logic employed | Numerous errors when solving problem | Less than 5 errors | No errors |
| Strings and  ToString  **5%** of coding  Grade | Does not include Strings | Includes some strings but they don’t work | Employs 10 strings | Strings function correctly |
| Random number generator  Values  Integers  **5%** of coding grade | Does not employ the rand/val/int functions | Includes rand/val/int but do not function correctly | Has rnd/val/int but some used incorrectly | Has rnd/val/int and they work perfectly |
| Daily progress  **25%** of your overall grade for this project | Did not produce a storyline and did not make significant progress each day | Has some new code each day but major and many logic errors | Code is produced each day with minimum logic errors | Code is designed well and project reflects the concept |
| Technical Approach  **25%** of your overall Grade for this project | No Story Line  No Code  Nothing significant learned | Followed instructions but the project was not completed many errors | Some rooms are detailed, some are not. | All rooms are laid out with enough detail to support the coding effort |
| Variables  **5%** of coding Grade | Clearly does not understand the use and syntax for variables | Some variables coded correctly some not | Most variables coded correctly | All variables coded correctly |
| Use of  Forms,  Labels,  TextBoxes, MessageBoxes, RadioButtons, CheckBoxes, CleckListBox,  MenuStrips,  Buttons,  PictureBoxes,  If Then Else  **20%** of coding Grade | Inappropriate use of controls. Does not convey the student understands how these controls are used. | Some controls coded correctly | Most but not all coded correctly | Exquisite use of controls |
| 5% Partnership | No synergy generated between partners | Some evidence of better work based on work done together | Clearly both partners learned from the experience of working together | Project includes work that was not covered in class and is evidenced by the partnership working together. |
| Presentation  **5%** of coding Grade | No presentation given | Presentation was not clear did not follow a script. Winged it. | Presentation contained elements of a script but went off script | Presentation was well organized and rehearsed. |