Python Final Project Ideas

Racetrack Game

3 sprites (characters) start at the left side. User picks a winner. For each of the 3 sprites, a random number 1-5 is generated for each sprite. After the three numbers are generated, the sprite are moved to the right (blanks insert before sprite). First sprite to cross 100 spaces from left wins. If more than one, the one farthest to left wins. If still tied, the winner is picked by a random number.

Adventure game

3 floors, 3 rooms on each floor. Rooms can have dragons, stairs up, treasure, etc. Have to kill dragon and get treasure to win. Use enters a room, can get inventory of things in room.

Battle

Battle different monster with different super powers (pokemon).

Card game war

where a deck of 52 cards (1-13) are shuffled and dealt out to two players. The cards are played one from each deck and the higher card wins and is added to the winners deck. If a tie, flip over cards until one player wins all the cards in the war. The game continues until one player has all cards.

Card games - Old Maid, Fish

Convert numbers to/from Roman Numerals

Rock/paper/scissors

Simple baseball/soccer game

Use random numbers and batting averages for two teams, probabilities of hits, outs, double play, errors, etc., play for 9 innings or until a team wins.

Additive persistence

Additive Persistence is a property of the sum of the digits of an integer. The sum of the digits is found, and then the summation of digits is performed on the sum, repeating until a single integer digit is reached. The number of such cycles is that integer's additive persistence. Consider the following example: 1. The beginning integer is 1234 2. Sum its digits is 1+2+3+4 = 10 3. The integer is now 10 4. The sum of its digits is 1 + 0 = 1 5. The integer is 1. When the value reaches a single digit, we are finished. This final integer is the additive root The number of cycles is the additive persistence. The integer 1234 has an additive persistence of 2 (first sum was 10, then the second sum was 1). The final digit reached is called the integer's additive digital root. The additive digital root of 1234 is 1. The **multiplicative persistence** and resulting multiplicative root are determined the same way, only multiplying the digits of an integer instead of adding. For example 1. The beginning integer is 1234 2. The product of 1*2*3*4 = 24 3. The integer

Introduction to computer science

*is now 24 4. The product of 2*4 = 8 5. The integer is now 8. When the value reaches a single digit, we are finished. This final integer is the multiplicative root. As before, the number of cycles is the multiplicative persistence. For 1234, the multiplicative persistence is 2, and its multiplicative root is 8.

Some Other Ideas

1. A 2-player game.
2. A single-player game (player vs computer).
3. A calculator with buttons.
4. A choose your own adventure novel (use buttons to make a choice).
5. A jigsaw puzzle (or other type of puzzle).
6. Tic-tac-toe [we have since made this Project 3]
7. A shopping list (With choice to add items and to mark them as purchased)
8. A program that randomly picks a person from the class, using bouncing balls to represent each person (does this ring a bell?)
9. A car racing simulation where you can put a bet on one of the cars.
10. A physics simulation showing the effect of gravity on different (real) planets.
11. A Jeopardy-style game show quiz.
12. A virtual pet
13. A "spin-the-wheel" app that does something depending on where the wheel randomly spins.
14. A recipe book which lets you click on a recipe to show the steps, with multiple possible recipes (and images)
15. A command-line calculator program using Python.
16. A quiz game with multiple wacky and nonsensical questions (basically a guessing game) which also had a bunch of random mini-games like "catch the ball" to cleverly incorporate some of the block requirements we expected.
17. A science trivia game.