Ted Korby
Pseudo Code for Wheel of Fortune Assessment
12/23/2021

Non-OOP Wheel of Fortune Python Game

Variables, lists and dictionaries:

- Player list, 3 users with associated values
    - Player dictionary
        - Name
        - Bank
        - has_million_token=false
- Active_player
    - Integer that is used to determine active player and next player
- Wheel
    - 24 segments * 3 spaces a piece (except for MDS)
        - Type, Label, Value
    - Million Dollar Segment
        - 1 space wins, 2 spaces bankrupt
- Active wheel value
    - Random 'spin' of the wheel, equal to the value of randomly obtained item from wheel
- Puzzle
    - Reference Word Guessing game
- Rounds (Int)
    - 1, 2 and final
    - 1 & 2 are 3 player
    - Final is single player /w largest bank
- Letters (list)
    - Contains all letters or user string.asciichar()
- Final Round letters (list)
    - Contains R, S, T, L, N and E
- Vowel, Vowel Cost (list) (int)
    - Contains vowels
    - Cost: 250
- Guessed_letters
    - List containing letters already guessed
- Final guesses
    - List containing puzzle guesses for final round
- Win checker
    - Bool to check if round was won

Game:

Set up:

Initialize variables, lists and dictionaries

Initialize 3 players with empty banks and preset names: Bob E. Flay, Sue E. Chop, Tone E. Stark

Print welcome statement with game title and player names

Main loop:

Round 1&2 (if round in [1,2]):

Print round number and statement

Generate random puzzle for players to solve

Display blank puzzle like Word Guessing Game

Start with player1(Bob)

Player Turn():

Turn start…

Print player name and bank value

Wait 1 second

Spin the wheel()

Store value to active value

Print active value + statement

If active value is 'lose a turn' or 'bankrupt',

Apply effect

Skip to next player

Else

Prompt user for consonant

If on board and not in already guessed letters and not in vowels

Add to guessed letters

Award player bank with wheel amount

Show board

Go to turn continued

Else

Print bad guess and skip to next player

*Turn continued():

Prompt user input for action

Do one of 3 things (if get_number_input == …):

1. Spin wheel and pick consonant

    Random wheel value

    If…

    else

        Get user consonant choice

    If…

    Else…

2. Buy vowel()

    If player has 250 or more in bank

        Remove 250 from bank

        Prompt for user input

        Recycle if not in vowels

        If on board and not in guessed words

            Reveal and go to turn continued

        Else

            Print statement and skip turn

3. Solve_puzzle()

    Guess = solve_puzzle()

    If guess == puzzle

        Win = True

    Else

        Print statement

        Go to next player

If win = True

    Print statement that player x won

    Win = false

```
            Round += 1
            Empty guessed letters
            Stuff?
            Break
        If active player not in [0,1]:
            Active player = 0
        else
            Active player += 1
Round 3:
        Print final round statement
        Find and select player with highest bank total
        Display user and bank
        Generate puzzle
        Put base words in guessed letters
        Prompt user to guess 3 consonants and a vowel one by one
        After all have been recorded to guesses
            Display puzzle
        Print statement; player has 3 guesses to answer the puzzle OR 15 seconds
        While True:
            If len(final_guesses) < 3:
                Guess = solve_puzzle()
                Append guess to final guesses
                If guess == puzzle:
                    User wins and receives additional 10,000 to bank
                    Print win statement
                    Display name, amount won
                Else:
                    Print incorrect
                    Loop back
```

```
            Else:
                Print lost statement
                Display name, amount won
        Print exit statement
        Wait 5 seconds
        Exit()


If has_bugs == True:
        Squash_bugs()
        Print("problem solved")
```