**Python Project: “Wheel of Fortune”**

1. **Introduction**

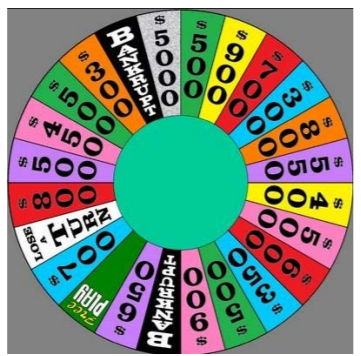
Project contains a puzzle which is a phrase containing several words, which are made up of letters. At the start, all the letters are set to blank. The players are to fill in the blanks by guessing one letter at a time, until a player could complete the phrase correctly. Before guessing, a player needs to spin a wheel, shown in Figure 1 below.

Figure 1: Wheel of Fortune

This YouTube video: <https://www.youtube.com/watch?v=OeX1rJ9Abr0> shows how the game is played. The program should be able to play like shown in the time segment starting from 3:16 up till 5 minutes.

1. **Rules and Actions**

A player spins the wheel which will stop at a sector. Here are the rules and actions upon a spin:

1. If the spin stops at a number, the player is to call out a consonant within ten seconds.
2. Player calls out a consonant:
3. If the consonant exists in the phrase, the player is awarded the dollar value of the number multiplied by the number of occurrences of the consonant in the phrase, and the consonant is filled into the phrase. The player is then allowed one of three action:
   1. Spin the wheel again, which means we return to Step 1.
   2. Buy a vowel by calling out the vowel desired. One of three things can happen:
      1. Each vowel costs $250 and is to be deducted from the player’s total so far. If the player’s total is below $250, the player cannot buy and therefore must spin again.
      2. The vowel exists and is filled into the appropriate blank(s) in the puzzle. $250 is deducted from the player’s total regardless of the number of occurrences of the vowel. The player then can buy another vowel (return to Step 1.a.i.b) or spin the wheel again (return to Step 1).
      3. The vowel doesn’t exist or has already appeared. The player loses the turn. $250 is deducted.
   3. Solve the puzzle.
4. If the letter does not exist, the player loses the turn.
5. If the spin stops at “Bankrupt”, the player loses all the money won so far and loses the turn.
6. If the spin stops on “Lose a turn”, the player loses the turn but keeps the money won so far.
7. If the spin lands on “Free play”, the player is to call out a letter, either consonant or vowel.
   1. If the letter exists, it is filled into the phrase. No money is added. Then the player returns to spinning the wheel, buying a vowel or solving the puzzle.
   2. If the letter doesn’t exist, the player proceeds to spinning the wheel, buying a vowel or solving the puzzle without losing a turn.
8. Upon successfully filling in a letter, the player is offered one chance only to complete the phrase within 10 seconds.
   1. If this is successful, the player keeps the money earned so far, and the game ends. All the other players get nothing.
   2. If this is not successful, then the player loses the turn.
9. Once the current player loses the turn, the next player spins the wheel, and we return to Step 1 for this new player.
10. **What is Required**
11. Write a program to play the game according to the rules above with three players. The program should be easily readable and understandable. Use meaningful names for variables and add comments appropriately. You need to have a mean to identify the players.
12. You will need to provide a display. It should be clear, neat and intuitive for a player to comprehend it and play the game. It should contain the following contents:
    1. A picture of the wheel with the sectors, each marked correctly with a number or phrase.
    2. A means of simulating spinning and stopping the wheel. You need to be able to control how long the wheel spins, with a certain degree of random variation.
    3. A means for indicating which sector a spin stops at.
    4. At the start, display blanks that indicate the letters to be filled in.
    5. As the game progresses, fill in the letters in the right position as a player calls them out.
    6. Display the score of each player as the game progresses.
13. When a player successfully completes the phrase, award the winning amount to the player and terminate the game.
14. **Resources**
    * + 1. The phrases that you can use in the program are stored in a text file called **WofFPhrases.txt.** Each phrase is contained in one line in the file. Your program should automatically and randomly select a phrase from the file and offer that as the puzzle to be solved.

If the phrase selected is “Wheel of Fortune”, then your display should show a pattern of boxes such as this:



As the game progresses, the boxes are filled in with letters until a player guesses the complete phrase correctly.

* + - 1. The displays will need to be done with graphics. For the graphics, you need to use the graphics library provided with Python. One basic requirement is that the display must be clear, so that the user can see exactly the states of play in the game.
      2. A flow chart is also included below.

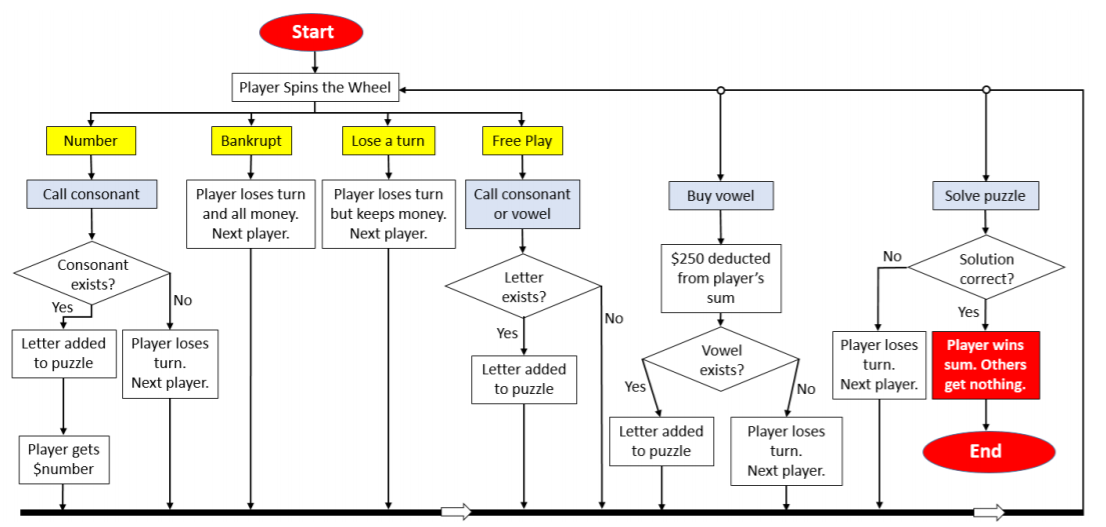
**Flowchart of the Game**

Figure 2: Flowchart of the Game

Notes:

* 1. Program starts at “Start” Box. The flow follows the directions of the arrows.
  2. Red boxes demarcate the start and end. Yellow boxes are outcomes of a spin. Grey boxes are for decisions by the player.
  3. Box which says “Next player” meant that there would be a change in player.

**FAQs**

**Q:** Is there a need to create a GUI?

**A:** You do not have to create an interactive GUI. The display is purely for output, simulating the turning of the wheel and displaying the result of each turn. All inputs (such as calling out a letter) can be made through text interface in the Python shell. But if you can create an interactive GUI, you may do so.

**Q:** Do we have to allow multiplayers where by multiple people can play the game?

**A:** Yes, multiple players – three.

**Q:** Can we use Object-Oriented Programming? Can we have multiple files?

**A:** Yes, you can use OOP (Preferably OO style).

**Q:** Can we use other graphics libraries like matplotlib or Pygame?

**A:** No, Graphics should be done using Turtle only.

**Q:** How do we graphically rotate the wheel which requires the rotation of the text as well?

**A:** It is not easy to rotate the wheel using Turtle, especially when you must rotate the text as well. You can avoid doing that and yet produce the effect of the wheel rotation.

In rotating the wheel, you need to have a marker which marks the sector that the wheel ends at. This marker can be a little circle, square or cross, something graphical for marking the stopping position of the wheel. The marker and the wheel have a relative motion, i.e., the wheel rotates, and the marker is fixed.

Now instead of rotating the wheel, you can rotate the marker instead, which is a lot easier to do graphically.

**Q:** How do I determine how far the wheel should rotate upon a spin?

**A:** The distance the wheel travels upon a spin can be counted by the number of sectors passing the position of the marker (or when you rotate the marker, then the number of sectors the marker travels through). This distance needs to be non-specific, i.e. different spins can produce different distances. This distance needs to be within a certain range, say between 20 and 30 sectors; the actual number is to be determined randomly. If you want to add variety to the game, you can even have players of different physical strengths, with the stronger players spinning through a larger number of sectors.

**Q:** There is a requirement of a time limit of 10 secs for the player to make their decision. However, the time limit is only mentioned at the first spin for each player. Also, it is also only mentioned where the player upon guessing a correct letter, has 10 seconds to decide if he wants to complete the phrase.

* 1. If the player does not want to complete the phrase, is there still a 10 seconds limit?
  2. As there are quite several decision-making process (choosing whether to buy vowel, spin again or guess the phrase) involved. Am I supposed to limit all decision-making to 10 secs?

Once a decision has been made, is it true that the player only has 10 secs to enter his 'guess' (can be one letter or the whole phrase)?

**A:**

a.If the player doesn’t want to complete the phrase, then he/she must choose one of the other available actions.

b. Yes, there’s always a 10 second limit to the player’s action: spin, solve or call a letter. In the game on TV, they have only 3 seconds.

**Q:** Can I use the Tkinter library?

**A:** No. Please stick to using Turtle.

**Q:** Should we make the wheel or use the picture given in the project handout? If we use the picture, then how will it interact with turtle?

**A:** You can use the picture given or make your own wheel. If you use the given wheel, then you can keep it static, essentially as the background to your display, but then rotate the marker that shows where a spin land on.

Basically, you need to simulate the effect of spinning the wheel. This point of “simulating the effect” also means that you can draw your own wheel which may or may not look like a wheel at all, so long as you can produce the effect of a spin and the marker landing on a particular sector.