Assignment 6

Due March 30

March 23, 2022

Instructions

For this assignment, you should write a Python script that solves the problem described below. The first line of your file should contain a comment stating:

> The code for this project represents my own, independent work. I have neither given nor received help on this assignment from other students.

Add another comment with your name after this comment.

Description

In this assignment, you will write a script that reads in a file describing a game of Uno that is in progress and print out information about the state of the game.

Input

Your program should prompt the user to enter the name of the saved game file to load at the beginning of the program.

File format

The input file will store information about the game in the following format. The first 3 lines of the file will contain the number of players (a positive integer), which player is about to player (0 up to players $-$ 1), and the direction of play (1 if moving 1 → 2 → 3, etc., -1 if moving 3 → 2 → 1, etc.), in that order. The file then includes the name of all players (in order), one per line, followed by the number of cards each player has, one per line, in the same order as the player names. The next two lines in the file are the number of cards in the discard pile and the number of cards in the deck (in that order). The sum of all players' hand sizes, the discard pile, and the deck will equal 108, the total number of cards in the game.

The file then lists all cards in all player's hands (player 0 cards, then player 1 cards, etc.), all of the cards in the discard pile (bottom to top), and all of the cards in the deck (bottom to top), one card per line. Each card is described by a two-character string in the game file. “Wild” and “Wild Draw Four” cards are represented as “WI” and “W*”, respectively. Every other card is represented by one character that represents a color (R, G, B, or Y: Red, Green, Blue, or Yellow), followed by one character representing its number or effect. Uno cards are numbered 0-9, with 3 kinds of special cards: Skip (represented as S), Reverse (represented as R), and Draw Two (represented as D). For example, G4 represents a Green 4, while YR represents a Yellow Reverse.

Output

After reading in the input file, your program should print out top card of the discard pile, all of the cards in the first player’s hand (“your hand”), and the number of cards held by all other players in the game, in the format below:

```
Top card:  Green 4

Your hand:
Red 4
Red 7
Wild
Wild
Green Skip
Wild Draw Four
Red 1

Barry's hand:  7 card(s)
Cara's hand:  7 card(s)
Dave's hand:  7 card(s)
```

Your output should use the names listed in the file for players 1–3 when giving the number of cards in their hands. Note that the top card of the discard pile will be the last one listed in the file (bottom-to-top order), not the first. Your code will not be tested on files that are not formatted properly.

Required functions

You must implement two functions:

- `load_game(filename)`: opens the file with the given name and prints out the required output (as described above)
- `full_name(card)`: this function accepts the two character abbreviation of a card and returns a string describing the card. For example, “G4” should return “Green 4”, “YR” should return “Yellow Reverse”, and “W*” should return “Wild Draw Four”.

Most of the logic for your script should be in the `load_game` function.

Sample output

You have been provided with 3 sample Uno game files, `game1.txt`, `game2.txt`, and `game3.txt`. All three files represent different points in the same game, though, so make sure that your script can handle games that have different numbers of players and different player names.

Correct `game1.txt` output:

```
Top card:  Green 4

Your hand:
Red 4
Red 7
Wild
Wild
Green Skip
Wild Draw Four
Red 1

Barry's hand:  7 card(s)
Cara's hand:  7 card(s)
Dave's hand:  7 card(s)
```

Correct `game2.txt` output:

```
Top card:  Yellow 4

Your hand:
Green Skip
Wild Draw Four
Red 1

Barry's hand:  7 card(s)
Cara's hand:  5 card(s)
Dave's hand:  9 card(s)
```

Correct `game3.txt` output:

```
Top card:  Green Reverse

Your hand:
Red 8

Barry's hand:  8 card(s)
Cara's hand:  2 card(s)
Dave's hand:  6 card(s)
```

Hints

The problem is easier if you ignore the parts of the game file that aren't required to be printed. In particular, the current player, the direction of play, all players' hands other than player 0, all discarded cards other than the last (top) one, and all cards in the deck are all irrelevant for producing the required output. You'll need to store everything else: the number of players, all players' names (except maybe player 0), the size of each player's hand, the cards in player 0's hand, and the top card of the discard pile. The players' names, the hand sizes, and the cards in the first player's hand can all be stored in lists.

If you wish to read in all of the data, the trickiest part is storing all of the cards for all of the players. You can use a list of lists to store the players' cards. Basically, each entry is a list containing all of the cards for one player, and the "list of lists" is a list containing all of these lists. For example, in `game1.txt`, the list of lists would contain 4 lists of 7 cards (strings) each.