Assignment 4

Due Feb. 22

February 17, 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:

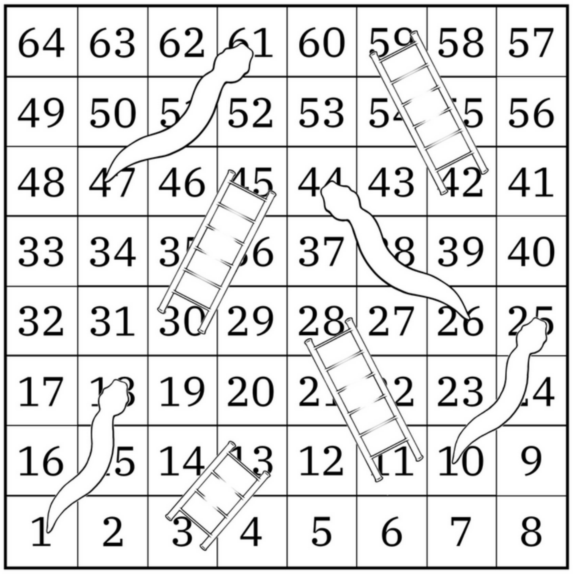
> The code for this project represents my own, independent work. I have neither given nor received help on this assignment from other students.

Add another comment with your name after this comment.

Description

In this assignment, you will write a script to simulate a version of Snakes and Ladders (sometimes called Chutes and Ladders). Snakes and Ladders is a board game for 2 or more people where each player takes a turn rolling a die, moving that many spaces forward on the board. Then, if that player lands on a space with a snake head, they are sent back to the square with the snake's tail, whereas if they land on a square with a ladder, they climb to a higher square. Players will continue taking turns rolling and moving until one of them reaches the final square on the board. Players must reach the goal by an exact roll of the die—any roll that would cause them to move past the final square is ignored, and they lose their turn.

Your game of Snakes and Ladders should match the board below:



The board has a total of 64 squares (numbered 1–64). Players start in square 1, and the goal is square 64. There are four snakes on the board (61 to 47, 44 to 26, 25 to 10, and 18 to 1) and four ladders (3 to 13, 11 to 28, 30 to 45, and 42 to 59).

Input

Your program should not accept any input from the user. However, you must define and implement two functions as part of your solution.

- `snake_or_ladder(pos)`: this function should accept a position on the board (1–64) and return the final space that a player who lands there would end up
  - If the position is the head of a snake, return the tail.
  - If it is the bottom of a ladder, return the top.
  - All other positions should return themselves.
- `take_turn(pos)`: this function should accept the current player's start position, roll the die, then update their position appropriately
  - To roll a die, import the `random` package, and call the function `random.randint(1, 6)`, which will return a random integer between 1 and 6 (inclusive)
  - The player will move forward that number of spaces (unless the roll would put them beyond square 64), then check for snakes and ladders (see `snake_or_ladder`)

Output

At the start of each turn, the game should print the position of both players. At the beginning of the game, this will be:

```
Player 1 is on space 1.
Player 2 is on space 1.
```

Before calling `take_turn`, print the message “Player `X` rolls the die: ”, where you should replace `X` with 1 or 2 (player 1 takes the first turn). Immediately after rolling the die, the `take_turn` function should print the die roll *on the same line* as the previous print statement. If the `snake_or_ladder` function encounters a snake or ladder, it should print the message “Snake!” or “Ladder!”, respectively. Finally, you should print the message “Player `X` wins!!!” when that player reaches the goal (space 64).

Sample output

```
Player 1 is on space 1
Player 2 is on space 1
Player 1 rolls the die:  2
Ladder!
Player 1 is on space 13
Player 2 is on space 1
Player 2 rolls the die:  3
Player 1 is on space 13
Player 2 is on space 4
Player 1 rolls the die:  1
Player 1 is on space 14
Player 2 is on space 4
Player 2 rolls the die:  5
Player 1 is on space 14
Player 2 is on space 9
Player 1 rolls the die:  2
Player 1 is on space 16
Player 2 is on space 9
Player 2 rolls the die:  4
Player 1 is on space 16
Player 2 is on space 13
Player 1 rolls the die:  6
Player 1 is on space 22
Player 2 is on space 13
Player 2 rolls the die:  6
Player 1 is on space 22
Player 2 is on space 19
Player 1 rolls the die:  6
Player 1 is on space 28
Player 2 is on space 19
Player 2 rolls the die:  3
Player 1 is on space 28
Player 2 is on space 22
Player 1 rolls the die:  6
Player 1 is on space 34
Player 2 is on space 22
Player 2 rolls the die:  5
Player 1 is on space 34
Player 2 is on space 27
Player 1 rolls the die:  5
Player 1 is on space 39
Player 2 is on space 27
Player 2 rolls the die:  5
Player 1 is on space 39
Player 2 is on space 32
Player 1 rolls the die:  2
Player 1 is on space 41
Player 2 is on space 32
Player 2 rolls the die:  4
Player 1 is on space 41
Player 2 is on space 36
Player 1 rolls the die:  5
Player 1 is on space 46
Player 2 is on space 36
Player 2 rolls the die:  1
Player 1 is on space 46
Player 2 is on space 37
Player 1 rolls the die:  1
Player 1 is on space 47
Player 2 is on space 37
Player 2 rolls the die:  4
Player 1 is on space 47
Player 2 is on space 41
Player 1 rolls the die:  5
Player 1 is on space 52
Player 2 is on space 41
Player 2 rolls the die:  1
Ladder!
Player 1 is on space 52
Player 2 is on space 59
Player 1 rolls the die:  2
Player 1 is on space 54
Player 2 is on space 59
Player 2 rolls the die:  1
Player 1 is on space 54
Player 2 is on space 60
Player 1 rolls the die:  5
Player 1 is on space 59
Player 2 is on space 60
Player 2 rolls the die:  5
Player 1 is on space 59
Player 2 is on space 60
Player 1 rolls the die:  3
Player 1 is on space 62
Player 2 is on space 60
Player 2 rolls the die:  2
Player 1 is on space 62
Player 2 is on space 62
Player 1 rolls the die:  6
Player 1 is on space 62
Player 2 is on space 62
Player 2 rolls the die:  4
Player 1 is on space 62
Player 2 is on space 62
Player 1 rolls the die:  2
Player 1 wins!!!
```

Hints

The only things to watch for on this lab are making sure that:

- your `snake_or_ladder` function finds all of the snakes and ladders (and no snakes or ladders it shouldn't),
- players lose turns if they roll past the goal, and
- the output format is correct.

If all of these are working, your script is probably correct.