Simulating the Birthday Problem in Python

The “birthday problem” asks how many people need to be in a room for it to be more likely than not that at least one person shares a birthday with another person. To simplify the problem:

- Assume there are only 365 days in each year (no leap year).
- Assume that every birthday is equally likely (birthdays are uniformly distributed).
- The simulation should count the number of people up to and including the person that has a repeated birthday.

**Part 1:** Write a simulation to simulate trying the birthday problem a total of 1,000 times. In this part, use a function in your simulation called `findBirthdayCount()` that returns the number of people needed to have a duplicate birthday. *(We’ll program that later!)*

**Part 2:** Write the `findBirthdayCount()` function that always returns 50, allowing us to test run the simulation before we simulate the count.

**Part 3:** Test your simulation. What is in our DataFrame?
Part 4: Complete `findBirthdayCount()` function, counting the number of people needed before two people share a birthday.

### Algorithm:

### Python:

Part 5: Run the simulation again, with the `findBirthdayCount()` function completed!

**Analysis:** Create a histogram of the result. How many people need to be in a room for it to be more likely than not for two people to share a birthday?

...how do we know this?