# The Birthday Problem

## Introduction

Today is my 42nd birthday. In the spirit of the Coding Dojo, we will celebrate my birthday with some light coding as the icing on the cupcake.

## Problem

How many people do we need to have in this room such that the probability of having at least two people with the same birthday is larger than 50%? If you read a book on probability theory and work through the math, you will get a closed form formula for the probability of having duplications. The right number is actually pretty low compared our intuition, thus this problem is sometimes called the *Birthday Paradox*. You can find the close form formula on Wikipedia if you don't want to read a book.

Instead of doing the math ourselves, we will ask the computer to do the hard work. We will simulate people in the room by giving them random birthdays and figure out the probability of having the same birthday.

## Solution

1. Find out how to generate random integers in the range of 1 to 365 in the programming language you are using. This is the birthday of our simulated people in the room. *Some languages can only generate random floating point numbers between 0 to 1 or random integers between 0 to RAND_MAX. You will have to find a way to map them into the right range.*
2. For a specific number of *N* people in the room, generate *N* of such random numbers.
3. Find out if there is any duplications in this set of integers. If there is a duplication, we call it a **hit** or **collision,** otherwise it is a **miss**. *Does your programming language provide a way to do this without sorting first? How would you implement this? What's the complexity of your solution?*
4. Repeat Step 2 and 3 for a number of trials, calculate the ratio of hit v.s. total number of trials. This should be close to the probability of duplicated birthday given by the formula. *Are they actually close to each other? How many trials do you need to get a reasonable approximation?*
5. Repeat Step 2 - 4 for a range of *N* from 0 to 100. Plot the probabilities v.s. *N* using the plotting functions in your language. Can you reproduce the probability distribution as shown in the Wikipedia page?
6. Step 1 and 2 are actually random sampling from integers in the range of 1 to 365. Find out if your programming language provides random sampling from a set of data elements (not necessary numbers). If it does, refactor your code to use such feature.

## Extended Problem

My birthday is one week away from Nicholas Constantine Metropolis. Can you extend our solution to find out the probability of people having birthdays within a week to each other instead of exactly the same day?