Birthday Attack

Songyue Wang (sywang@cs.hku.hk)

Question: There are n people (n is given) in a room, what is the probability that at least two people share the same birthday?

$$p(n) = 1 - \overline{p}(n)$$

$\overline{p}(n)$ is the probability that all n people have different birthdays

$$\overline{p}(n) = 1 \times (1 - \frac{1}{365}) \times (1 - \frac{2}{365}) \times \cdots \times (1 - \frac{n-1}{365})$$

$$p(n) = 1 - \frac{365!}{365^n (365-n)!}$$

$$\because e^x = 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \cdots (\text{Taylor's Formula})$$

$$\therefore e^x \approx 1 + x (\text{when x is very small})$$

$$\therefore \overline{p}(n) \approx 1 \cdot e^{-\frac{1}{365}} \cdot e^{-\frac{2}{365}} \cdot \ldots \cdot e^{-\frac{n-1}{365}}$$

$$= e^{-\frac{1+2+\cdots+(n-1)}{365}}$$

$$= e^{-\frac{n(n-1)/2}{365}}$$

$$\therefore p(n) = 1 - e^{-\frac{n(n-1)}{730}}$$

Python Code:

```
import math

def calculateProb(d, n):
  exponent = (-n * (n - 1)) / (2 * d)
  return 1 - math.e ** exponent;

# calculate the probability that at least 2 people share the same birthday,
given there is a total of 23 people
print(calculateProb(365, 23))
```

Therefore, in general:

$$p(n,d) \approx 1 - e^{\frac{-n(n-1)}{2d}}$$

d is the space of hash

Normally, a hash contains lower and upper case of letters and digits (26+26+10=62 different available characters). In SHA256, the hash string has 64 characters in hex (0-9 and a-f, 16 characters).

The total space of SHA256 hash (d) is 16^{64}.

Finding how many hashing operations will result in a collision with Python:

Suppose the hash has 8 hex characters, 1000 different messages are hashed

```
#calculate the probability that there is a collision if the hash has 8 hex
characters, and there are 1000 hashing operations
print(calculateProb(16**8, 1000))
```