Birthday Paradox

Monday, November 23, 2020 11:33 AM

Weak Collision Resistance - Given a specific hash value output, y, it should be computationally infeasible to come up with a value x such that f(x) = y

Strong Collision Resistance - It should be computationally infeasible to come up with any two different values x1 and x2, such that f(x1) = f(x2).

Here are two different questions:

1) What is the probability that someone in the class has the birthday September 14th?
2) What is the probability that some two people have the same birthday?

Think of the birthdays as the output of the hash function. The first question is saying…find someone with a particular birthday.

The second one is saying, find any two people with the same birthday…

There are 47 people on the call, except for me. (My bday is Sept 14.)

The probability one of you has a Sept 14 birthday is at most 47/365.

But, the probability that some two of you have the same birthday is much higher!!!

Calculate the opposite: the probability that all n people in a room have a different birthday.

Probability the first person is different than anyone I asked yet is 365/365

Probability the second person is different than anyone I asked yet is 364/365.

Probability the third person is different than anyone I asked yet is

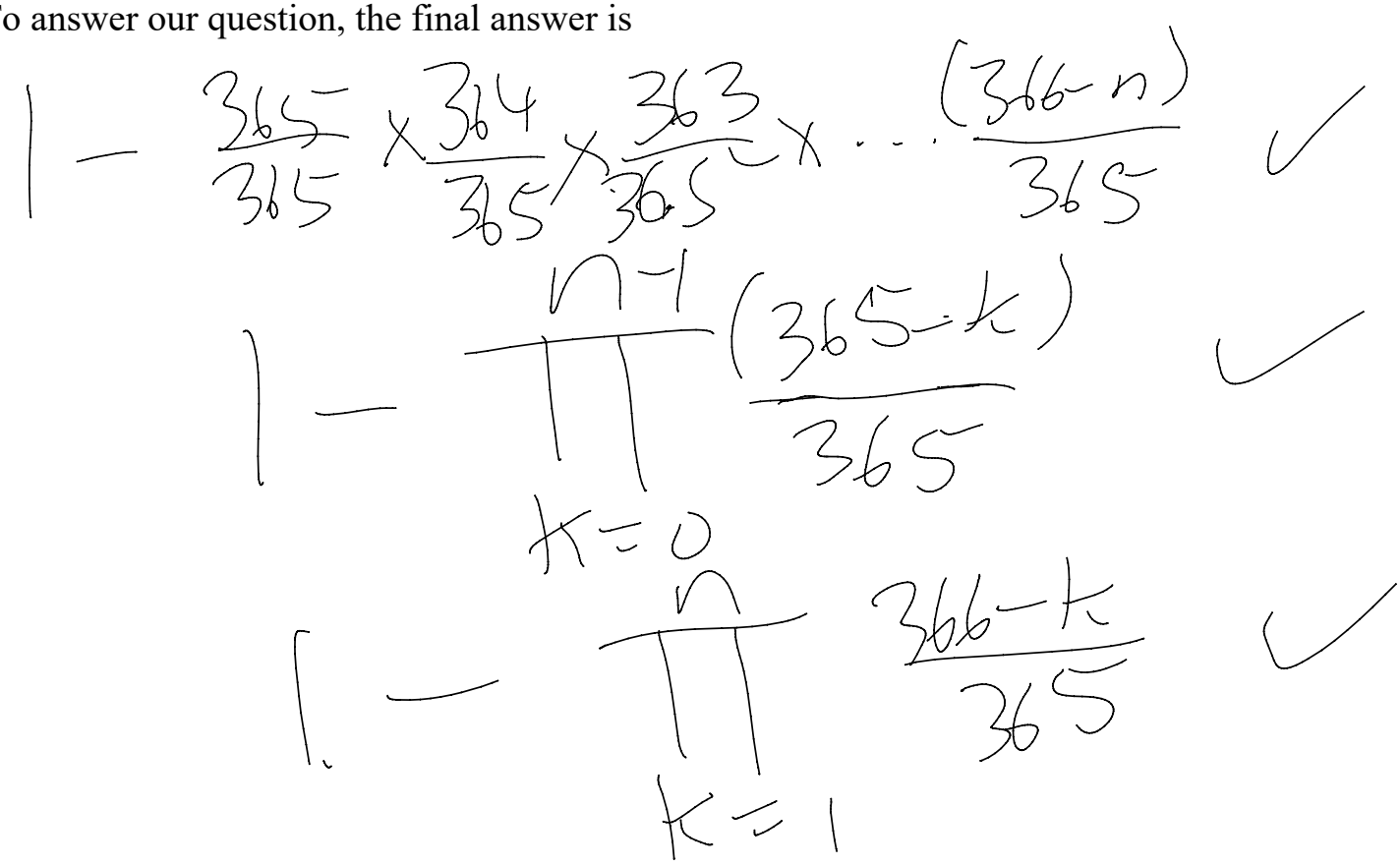
363/365.

…

Probability the n^{th} person is different than anyone I asked yet is

(365-n+1)/365 = (366-n)/365

To answer our question, the final answer is



In general, if there are n total hash values, you only have to randomly generate sqrt(n) possible inputs before you get some collision (ie. two different inputs that hash to the same value.)

Sample Assignment (Fall 2016) to illustrate Birthday Attack

Given a Hash Function.
Goal: Find two input strings that map to the same output.

Technique: HashMap in Java.

Key --> Value
hash function output --> String which created it.

HashMap will be an inverse map of the hash function.

When I find a new string that hashes to the same value as an old string, my table will NOT be empty at that slot!!!

19 --> cat
27 --> dog

later we find that elephant --> 19

look in the table at slot 19, we see cat is already there!
So I have my match!