458. Poor Pigs

April 15, 2019 | 3.7K views

There are 1000 buckets, one and only one of them is poisonous, while the rest are filled with water. They all look identical. If a pig drinks the poison it will die within 15 minutes. What is the minimum amount of pigs you need to figure out which bucket is poisonous within one hour?

Answer this question, and write an algorithm for the general case.

General case:

If there are `n` buckets and a pig drinking poison will die within `m` minutes, how many pigs (`x`) you need to figure out the **poisonous** bucket within `p` minutes? There is exactly one bucket with poison.

Note:

1. A pig can be allowed to drink simultaneously on as many buckets as one would like, and the feeding takes no time.
2. After a pig has instantly finished drinking buckets, there has to be a **cool down time** of `m` minutes. During this time, only observation is allowed and no feedings at all.
3. Any given bucket can be sampled an infinite number of times (by an unlimited number of pigs).

Solution

Approach 1: Pig as a qubit

Intuition

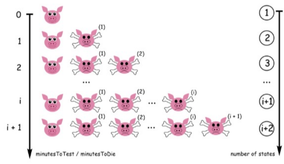
These "strange" questions are now asked by Google, Baidu and IBM because of their interest in quantum computing. Quantum bit (or *qubit*) is the basic unit of quantum information, it's the quantum version of the classical binary bit. Binary bit has only two states : `0` and `1`, and on a very basic level the qubit has more. In such questions we deal with an object (here is a pig) which has more than two states.

How many states does a pig have

If there is no time to test, i.e. `minutesToTest / minutesToDie = 0`, the pig has only one state - `alive`.

If `minutesToTest / minutesToDie = 1` then the pig has a time to die from the poison, that means that now there are two states available for the pig : `alive` or `dead`.

One more step. If `minutesToTest / minutesToDie = 2` then there are three available states for the pig : `alive / dead after the first test / dead after the second test`.

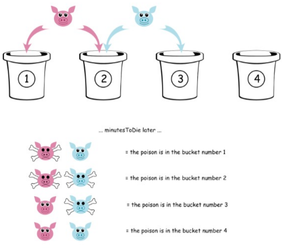


> The number of available states for the pig is `states = minutesToTest / minutesToDie + 1`.

How many buckets could test `x` pigs with 2 available states

One pig could test 2 buckets - let's make him drink from the bucket number `1` and then wait `minutesToDie` time. If he is alive - the poison is in the bucket number `2`. If he is dead - the poison is in the bucket number `1`.

The same way two pigs could test $2^2 = 4$ buckets

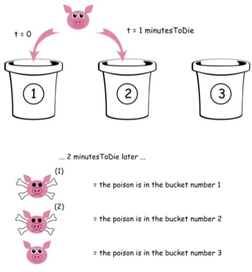


> Hence if one pig has two available states, `x` pigs could test 2^x buckets.

How many buckets could test `x` pigs with `s` available states

> After the discussion above, the answer is quite obvious : s^x buckets.

Let's consider as an example one pig with 3 states, *i.e.* `s = minutesToTest / minutesToDie + 1 = 2 + 1 = 3`, and show that he could test `3` buckets.



Solution

Hence the problem is to find `x` such that $states^x \geq buckets$, where `x` is a number of pigs, `states = minutesToTest / minutesToDie + 1` is a number of states available for each pig, and buckets is number of buckets.

The solution is well known : $x \geq \log_{states} buckets$. To simplify the code let's rewrite the equation with the help of natural logarithms :

$$x \geq \frac{\log buckets}{\log states}$$

Implementation

C++ | Java | Python — Copy

```
class Solution:
    def poorPigs(self, buckets: int, minutesToDie: int, minutesToTest: int) -> int:
        states = minutesToTest // minutesToDie + 1
        return math.ceil(math.log(buckets) / math.log(states))
```

Complexity Analysis

- Time complexity : $\mathcal{O}(1)$ since it's a constant time solution.
- Space complexity : $\mathcal{O}(1)$.

Rate this article:

Previous Next

Comments: 4

Sort By

Type comment here... (Markdown is supported)

Preview Post

pumbaacave ★ 10 April 16, 2019 3:17 PM
pigs are so poor, maybe we can use robots
9 Share Reply
SHOW 3 REPLIES

KuroushiA ★ 15 July 5, 2019 8:08 AM
I feel companies are asking this question to test on Dr Claude Shannon's Information Theory. Each pig can carry log(minutesToTest / minutesToDie + 1) bit of information. To identify the bucket, we need log(buckets) bit of information. Which means, we need Ceil(log(buckets)/log(minutesToTest / minutesToDie + 1)) pigs.
4 Share Reply
SHOW 1 REPLY

osiris1147 ★ 4 April 16, 2019 8:38 PM
I can't see how this is in any way related to qubits, but maybe someone could explain it to me.
A qubit is a state of a two level quantum system. It can be described as a superposition of the state 0 and 1 (or whatever someone calls the eigenstates of the system). If you apply a measurement on the system the wave function collapses to the state 0 or 1, where the probabilities for each state depend on
Read More
4 Share Reply
SHOW 1 REPLY

klimkina ★ 50 April 16, 2019 11:51 PM
So the answer is 5 pigs and we are using them like this:
minute 1: Pig 1: 10,20..90, 100; Pig 2: 210,220..290, 300
minute 2: Pig 1: 110..120..190 ...
minute 3: Pig 1: 1,11,21..91 ...
...
Read More
0 Share Reply

XuYa ★ 7 July 8, 2020 2:04 AM
PLM
0 Share Reply

kamenshchikov ★ 6 February 4, 2020 3:33 AM
Disagree: how do you assign 1000 buckets to 5 pigs in 4 trials so that you are guaranteed to find an answer?
While this solution makes sense in terms of information theory, we have to take into account:
Read More
0 Share Reply

syyh ★ 7 December 7, 2019 7:00 AM
Below is my some interpretation of the idea with traditional computer knowledge.
Let's assume there are 2 states, and x pigs. And number each bucket 0,1,2,....,
Let 0th pig eat buckets 0,2,4,..., and 1th pig eat 0,1,4,5,... In other words, kth pig eats every bucket whose number's kth bit is 1.
Make a number whose kth bit is 1 only if kth pig dies. The number is the the poisonous bucket.
Read More
0 Share Reply

duohuang60835 ★ 9 September 25, 2019 4:34 AM
I'll try to make a conclusion by math.
give each bucket a list with length x (pig number), each element in the list means ith pig drink this bucket in time element. 0 means never drink in this bucket, element should smaller or equal to it's states discussed above (minutesToTest / minutesToDie + 1), call it s.
Read More
0 Share Reply