

458. Poor Pigs

👍 232

👎 539

🏠

⌵

Description (/problems/poor-pigs/description/)

💡 Hints (/problems/poor-pigs/hints/)

📄 Submissions (/problems/poor-pigs/submissions/)

💬

(/problems/poor-pigs/discuss)

> Solution with detailed explanation

🔗 Share

🔔 Subscribe

🚩 Report

📄

Solution with detailed explanation

17.5K  
VIEWS

▲  
75  
▼

Last Edit: October 6, 2018 8:09 PM

👤 (/ifyouseewendy)

ifyouseewendy (/ifyouseewendy)

★ 147

To be honest, I've spent several days thinking about this question. It's not only fun, but also confusing. it drives me up the wall, especially for its **easy** difficulty. 😞

Thinking Process

1. What if you only have one shot?

Eg. 4 buckets, 15 mins to die, and 15 mins to test.

The answer is 2. Suppose we use A and B to represent pigs, we could have

Buckets	0	1	2	3
Drink by		A	B	AB

Obviously we could use the binary form to represent the solution.

Buckets	0	1	2	3
Drink by	--	A_	_B	AB
Binary	00	10	01	11

**Conclusion:** If we have  $x$  pigs, we could use them to represent (encode)  $2^x$  buckets.

2. What if we could have more than one attempts?

Eg. 4 buckets, 15 mins to die, and 30 mins to test.

At the moment, I consider the problem as an encoding problem: *With more attempts, how to use fewer pigs to represent all the buckets?*

I got lost at this step by keep thinking the binary way. After hanging around the forum, I got the idea to change my views. Let's go back to the one shot situation. What does the binary form mean? It's much easier if we regard it as:

- 0 means the pig does not drink and die.
- 1 means the pig drinks in the first (and only) round.

We could generalise with:

- 0 means the pig does not drink and die.
- 1 means the pig drinks in the first round and die.
- 2 means the pig drinks in the second round and die.
- ...
- t means the pig drinks in the t-th round and die.

**Conclusion:** If we have  $t$  attempts, we could use  $t+1$ -based number to represent (encode) the buckets. (That's also why the first conclusion uses the 2-based number)

### Example

Eg. **8** buckets, **15** mins to die, and **40** mins to test.

We have  $2 (= (40/15).floor)$  attempts, as a result we'll use 3-based number to encode the buckets.

How many pigs do we need? Answer is  $2 (= \text{Math.log}(8, 3).ceil)$

Buckets	0	1	2	3	4	5	6	7
3-based	00	01	02	10	11	12	20	21
First Rd	--	_B	--	A_	A B	A_	--	_B
Second Rd	--	--	_B	--	--	_B	A_	A_

For example 3-based number 02 means: the pig **A** does not drink and die, and the pig **B** drinks in the second round and die.

### Ruby Code

```
class FooTest < Minitest::Test
  def poor_pigs(buckets, minutes_to_die, minutes_to_test)
    states = minutes_to_test / minutes_to_die + 1

    Math.log(buckets, states).ceil
  end

  def test_run
    assert_equal 1, poor_pigs(2, 10, 10)
    assert_equal 2, poor_pigs(4, 10, 10)
    assert_equal 3, poor_pigs(8, 10, 10)
    assert_equal 4, poor_pigs(16, 10, 10)

    assert_equal 1, poor_pigs(2, 10, 20)
    assert_equal 2, poor_pigs(4, 10, 20)
    assert_equal 2, poor_pigs(8, 10, 20)
    assert_equal 3, poor_pigs(16, 10, 20)

    assert_equal 5, poor_pigs(1000, 15, 60)
  end
end
```

Comments: **15**

Sort By ▼



Type comment here... (Markdown is supported)

Preview

Post



pyeprog (/pyeprog) ★2 April 14, 2018 8:08 AM

It's really a beautiful conclusion!

(/pyeprog)

0 ^ v | Share | Reply



zhangyan985211 (/zhangyan985211) ★43 March 28, 2018 5:37 AM

I think this question is related to Entropy

(/zhangyan985211)

0 ^ v | Share | Reply