CODECHEF Discuss
A *Directi* Educational Initiative

questions   tags   users   badges   unanswered | **ask a question**   ab

CodeChef Discussion

Search Here...   ● questions   ○ tags   ○ users

BANROB - Editorial

1

PROBLEM LINK:

Practice
Contest

Author: Kirill
Tester: Kevin Atienza
Editorialists: Pushkar Mishra and Suhash Venkatesh

DIFFICULTY:

Easy

PREREQUISITES:

Math, Binary Exponentiation, Sum of Geometric Progression

PROBLEM:

Given that two thieves have 1 billion to divide amongst them according to a particular method, report the amounts both the thieves will receive upon division. The particular method here is that after a certain amount of time t, only (1 Billion * p^t) amount of money can be taken, where $0 \le p \le 1$. Also, there are only M minutes to divide the stolen money and escape.

EXPLANATION:

Subtask 1
Let us start by taking an example. Let $p = 0.5$ and $t = 4$. We not proceed minute by minute.

- First minute
  Money left to take = 10^9.
  Thief 1 to propose a plan.

- Second minute
  Money left to take = $10^9 * 0.5$.
  Thief 2 to propose a plan.

- Third minute
  Money left to take = $10^9 * 0.5^2$.
  Thief 1 to propose a plan.

- Fourth minute
  Money left to take = $10^9 * 0.5^3$.
  Thief 2 to propose a plan.

After the fourth minute, both the thieves will lose all the money. So, let's go back in time minute by minute and see what the optimal division would be.

- If in the fourth minute, Thief 2 proposes a division of 0 and $10^9 * 0.5^3$ respectively, Thief 1 will accept it.
- If in the third minute, Thief 1 proposes a division of $10^9 * 0.5^2 - (10^9 * 0.5^3)$ and $10^9 * 0.5^3$ respectively, Thief 2 will accept it.
- If in the second minute, Thief 2 proposes a division of $10^9 * 0.5 - (10^9 * 0.5^2) + (10^9 * 0.5^3)$ and $(10^9 * 0.5^2) - (10^9 * 0.5^3)$ respectively, Thief 1 will accept it.
- Finally, if in the second minute, Thief 1 proposes a division of $10^9 - (10^9 * 0.5) + (10^9 * 0.5^2) - (10^9 * 0.5^3)$ and $(10^9 * 0.5) - (10^9 * 0.5^2) + (10^9 * 0.5^3)$ respectively, Thief 2 will accept it.

The acceptances will be because the theif accepting the plan can in no way gain more money in the future than being offered at that point in time. Please note this point in the question:
"Each thief wants to maximize his earnings, but if there are two plans with the same amounts for him, he would choose the one which leads to a larger total amount of stolen dollars."

We can make a very important observation here. The amount that Thief 1 will gain is given by:
$val = 10^9(p^0 - p^1 + p^2 - p^3 \ldots p^{M-1})$. Thus, the amount that Thief 2 gets is $10^9 - val$.

Subtask 2
If we observe the terms inside the bracket in the expression we got for val in subtask 1, we can see that it is basically the summation of terms of a Geometric Progression (GP).
The GP is: $1, -p, p^2, -p^3, \ldots$

We common ratio $r = -p$ and the first term of GP $a = 1$. The formula to calculate the sum of first n terms of a GP is well known:
$S = \frac{a(r^n - 1)}{r - 1}$

Thus, in this case, $S = \frac{((-p)^M - 1)}{(-p) - 1}$.
This will work in $\mathcal{O}(M)$ with linear exponentiation.

Tags:

editorial ×**3,984**

easy ×**1,220**

math ×**150**

sept15 ×**145**

binary ×**73**

Asked: **05 Sep, 17:34**

Seen: **2,176 times**

Last updated: **03 Oct, 11:03**

Related questions

DIVIDE - Editorial

EQUALITY - Editorial

FLOORI4 - Editorial

ACMICL2 - Editorial

DONUTS - Editorial

CODECRCK - Editorial

LMA1 - Editorial

STFM - Editorial

NDLVOTE - Editorial

CC2 - Editorial

Subtask 3

The solution of subtask 2 can be optimised by using binary exponentiation to calculate $(-p)^M$. This will allow us to compute the sum of the GP in $\mathcal{O}(\log M)$.

COMPLEXITY:

$\mathcal{O}(\log M)$ per test case.

SAMPLE SOLUTIONS:

Author

Tester

Editorialist

sept15   binary   math   easy   editorial

This question is marked "community wiki".

edited **03 Oct, 11:03**

kevinsogo
1.6k ●4 ●16 ●42

asked **05 Sep, 17:34**

pushkarmishra
1.3k ●15 ●29 ●42
accept rate: 4%

---

9 Answers: oldest   newest   **most voted**

4

For those who couldn't understand the question, I'll explain it with a test case. Suppose p=0.6, M = 5.

t=0 -> money -> 1 billion. Here Chef will decide(make a decision).

t=1-> money -> 0.6 billion. Here other will decide.

t=2 -> money -> 0.36 billion. Here Chef will decide.

t=3 -> money -> 0.216 billion. Here other will decide.

t=4 -> money -> 0.1296 billion. Here Chef will decide.

If the situation comes to t=4, it is obvious that chef will take all the 0.1296 billion, and other thief will agree.

So, before this time (at t=3), the other thief will decide to give Chef 0.1296 billion, so that he will accept because in any case that is the maximum amount chef can get. So other thief will get (0.216-0.1296) = 0.0864 billion.

So, before this time (at t=2), the Chef will decide to give other thief 0.0864 billion, so that he will accept because in any case that is the maximum amount other thief can get. So chef will get (0.36-0.0864) = 0.2736 billion.

So, before this time (at t=1), the other thief will decide to give Chef 0.0.2736 billion, so that he will accept because in any case that is the maximum amount chef can get. So other thief will get (0.6-0.2736) = 0.3264 billion.

So, before this time (at t=0), the Chef will decide to give other thief 0.3264 billion, so that he will accept because in any case that is the maximum amount other thief can get. So chef will get (1-0.3264) = 0.6736 billion.

Finally, you will get a G.P. and then it can be solved easily.

Code https://www.codechef.com/viewsolution/8032865

link | award points

answered **14 Sep, 18:56**

rishivikram
298 ●6
accept rate: 23%

2

Well, for the fourth test case:

t = 0 -> p = $1000 \cdot 10^6$

t = 1 -> p = $500 \cdot 10^6$

t = 2 -> p = $250 \cdot 10^6$

t = 3 -> p = $125 \cdot 10^6$

I started backwards. At the third minute it is the second thief turn, right. Then the gold that is left is 125000000. Which means that if the second thief offers 0 125000000, the Chef should accept it. Otherwise. at the fourth minute, the Chef will take 0 as well. And we know that - **"Each thief wants to maximize his earnings, but if there are two plans with the same amounts for him, he would choose the one which leads to a larger total amount of stolen dollars."**.

So at the third minute the Chef would accept the plan 0 125000000.

Then, at the second minute, if the Chef offered the plan 125000000 125000000, the second thief would accept it, because the total amount of stolen dollars is more. If the Chef tried to offer more to himself and less for the second thief, the second thief would reject the offer, and receive more at the next turn.

Then again, at the first minute, if the second thief offered the plan 125000000 375000000, the Chef would accept it, because of the total-amount-rule.

At last, at the beginning the Chef will propose the plan 625000000 375000000, and the second thief will accept it, because of the total-amount-rule.

link | award points

edited **14 Sep, 18:32**

answered **14 Sep, 18:29**

aragar
70 ●3
accept rate: 12%

Check this video and subscribe to the channel for more video editorials.. https://www.youtube.com/watch?v=KaucPRyejis Hope this

2 helps..

link | award points

answered **14 Sep, 22:23**

ankit15
21•1
accept rate: 0%

This is really a great step. Video editorials are always useful! :)

kay_kay (14 Sep, 23:41)

---

sum of GP :P https://www.codechef.com/viewsolution/8129204

1 link | award points

answered **14 Sep, 17:21**

xariniov9
41•1
accept rate: 0%

Let $a(M)$,$b(M)$ be the proportions of the money the first and second thief obtain given M minutes to decide.

1 **Clearly $a(1) = 1$,$b(1) = 0$ as with no time left the second thief will agree to anything to avoid arrest.**

For general M, the first thief thinks:

If the other guy rejects my offer, I will get $pb(M - 1)$ and he $pa(M - 1)$ since we play the same game with less money, less time, and roles switched.

He will reject especially if I offer him less than $pa(M - 1)$.

He will greedily accept if I offer more than $pa(M - 1)$ but that would not be optimal for me; he will also accept by the tie-breaking rule if I offer exactly $pa(M - 1)$, which will leave $1 - pa(M - 1)$ for me, certainly at least as much as $pb(M - 1)$.

We obtain the recursion

$$a(1) = 1, a(M) = 1 - pa(M - 1)$$

$$a(M) = 1 - p(1 - pa(M - 2)) = 1 - p + p^2 a(M - 2)$$

$$a(M) = 1 - p + p^2 - p^3 + \ldots + (-p)^M a(1)$$

But Since $a(1) = 1$,

$$a(M) = 1 - p + p^2 - p^3 + \ldots + (-p)^M$$

$$a(M) = \frac{1 - (-p)^M}{1 - (-p)}$$

$$b(M) = 1 - a(M)$$

$$b(M) = 1 - \frac{1 - (-p)^M}{1 + p} = \frac{p + (-p)^M}{1 + p}.$$

So, After multiply with 10^9 we get our final $a(M)$ and $b(M)$.

Link to the solution:https://www.codechef.com/viewsolution/8037182

link | award points

edited **16 Sep, 17:00**

answered **16 Sep, 16:57**

vidhan13j07
21•3
accept rate: 0%

Can anyone help me out? suppose the test cases are:

0 4

1 0.5

2 0.5

3 0.5

4 0.5

for 1 0.5 output --> 10^9 0.0 ... ok .

2 0.5 output--> 5*10^8* 5*10^8* ... ok

3 0.5 output --> 750000000.0000 250000000.0000 --> why???? .. see chef has 750000000.0 i.e decision is made at t=1 . And chef offer other thief 250000000.0 because at the beginning of 3rd min other thief will be able to stole only 250000000.0 money .BUT why other thief agreed to chef ? I mean if other thief would not have agreed to chef then he will offer 500000000.0 to chef after the beginning of the 2nd minute and able to keep 500000000.0 with himself . And in that case chef also would not able declined to other thief because after 2nd min both of them will have reduce amount of money .

same problem with 4 0.5 --> output--> 625000000.0000 375000000.0000 ??? again decision made at beginning of 1st minute. Why other thief always agreed to chef ?. I doing something wrong . Please correct me

link | award points

edited **14 Sep, 17:57**

answered **14 Sep, 17:53**

mra1709
1•1
accept rate: 0%

1 @aragar .. Thanks a lot . I was thinking after t time both can (10^9 X p^t) amount of money . But actually total money both can take away is (10^9 X p^t) .

mra1709 (14 Sep, 18:44)

0 I am little bit confuse in this problem, in the problem statement it is written that:
"Each thief wants to maximize his earnings, but if there are two plans with the same amounts for him, he would choose the one which leads to a larger total amount of stolen dollars"
Also chef has opportunity to propose plan first that means he must take advantage of his turn
Now lets suppose M is odd:
then last turn is of chef and other has to accept his plan, correct ?
lets say : M=7,p,B,(x,y) where (0<=p<=1),B=>Bllion,x=amt amassed by chef and y= amt amassed by other
at t=0 chef proposes (B,0)
at t=1 other proposes (0,B*p*)
*at t=2 chef proposes (B*p^2,0)
at t=3 other proposes (0,B*p^3*)
*at t=4 chef proposes (B*p^4,0)
at t=5 other proposes (0,B*p^5*)
*at t=6 chef proposes (B*p^6,0)
at t=7 they have (0,0)
if other accept his plan then chef will get (B,0) (Maximum amt. stolen){because other have same amt. in all plans i.e 0} right ? but why he should accept his plan this question arises, because chef rejects all his plans bcoz he knew that at last he is going to propose last plan and other have to accept it then amt amassed by chef should be 1 Billion.
If M is even now chef knew that at last other is going to prposes plan and he(chef) have to accept it so chef will propose him plan (B-B*p*,B*p*).. can any body tell me am i wrong if yes then why ?

link | award points

answered **14 Sep, 19:56**

yugrockz
1
accept rate: 0%

0 Here is a simple solution.

At time t-1, the thief(thief 1) who has the chance would divide the money in his favor. That is the division would be {max*p^(t-1),0}.

Keeping this in mind the other thief(thief2) will divide the money as {max*p^(t-2)-max*p^(t-1), (max*p^(t-1)} in the previous chance as the sum of money will be more and thief 1 will be getting the amount he wants in the last chance.

The same logic goes on every chance and the chef would hence choose the division as (max –max*p*+*max*p^2-max*p^3…… *max*p^(t-1), max*p-max*p^2+max*p^3…… *max*p^(t-1))

This forms a simple geometric progression which can be calculated using G.P.formula.
https://www.codechef.com/viewsolution/8165228

link | award points

edited **14 Sep, 21:15**

answered **14 Sep, 21:09**

nik_iit007
1
accept rate: 0%

0 What is O(logM)

link | award points

answered **15 Sep, 14:15**

h4cktivist
1
accept rate: 0%

Your answer

[hide preview]

☐ community wiki

Type the text

Privacy & Terms

Post Your Answer