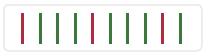
Description

You've played a couple of games with your friend. After a while, you both found out that if there are 2, 3, or 4 pencils left on the table, you automatically win. It happens because a player can take 1, 2, or 3 pencils and leave the other player with only one. The other player has nothing left but to take the last pencil and lose the game.

On the other hand, if you have 5 pencils on the table, you lose. You can take 1, 2, or 3 pencils, your friend will then take 1, 2, or 3 pencils as well leaving you with the last pencil. So, it will again lead to the situation described above but vice-versa.

The same thing happens when there are 6, 7, or 8 pencils left on the table. It will eventually repeat all over again.

It's easier to get a grasp of it with a line of 10 red-green pencils. In this example, we can be sure that if both players know the winning strategy, **the first one will be the winner**. Here is a game process:



The first player has an advantage and takes 1 pencil:

`|||||||||`

The second player has a disadvantage, so if the second player takes any number of pencils from 1 to 3, the first player is left with a winning strategy:

- 1: `||||||||`
- 2: `|||||||`
- 3: `||||||`

The first player stands in a winning position and takes that number. It will lead to a losing position for the second player:

`|||||`

The second player stands in a losing position—if the second player takes any number of pencils from 1 to 3, the first player will be left in a winning position:

- 1: `||||`
- 2: `|||`
- 3: `||`

The first player stands in a winning position and takes the right number of pencils. It leaves the second player with one pencil:

`|`

So as you can see, the outcome of the game is known before the game even starts and depends on the starting position of the players. Your friend came up with the idea of making the game a bit more replayable. Instead of taking input from two players, you need to program a bot that follows a winning strategy. If the bot's position isn't the winning one, you can program it to take any number of pencils (1, 2, or 3) at random. You can also come up with any pattern of your own for the losing position.

To make it easier to understand, **the logic of the bot** could look like this:

If the bot is in a **losing** position, the outcome of the game does not depend on his action, so if the number of pencils on the table in his turn is:

- 5,9,13,17... – bot takes a random number of pencils from 1 to 3
- 1 – bot takes the last pencil and loses

If the bot is in a **winning** position, his goal is to take as many pencils to put his opponent in a losing position, so if the number of pencils on the table in his turn:

- 4,8,12,16... – bot takes 3 pencils
- 3,7,11,15... – bot takes 2 pencils
- 2,6,10,14... – bot takes 1 pencil

Objectives

Implement the bot for the second of the two possible players. For example, in `Who will be the first (John, Jack)` `John` is the user and `Jack` **is the bot (and will always be the bot).**

So, if the player chooses `Jack` as the first player, after that input, the bot's move should be printed.

Your final objective is to expand your program. Write a solution, that can be executed for any initial number of pencils. Check each iteration whose turn is now. If it is the bot, instead of requiring input from the second player, output one line that contains the bot's move (`1`, `2` or `3`) that follows the winning strategy. If the bot is not in the winning position, make it follow any pattern of your liking, as the tests check only the bot's winning position.

Examples

Example 1:

```
How many pencils would you like to use:
> 10
Who will be the first (John, Jack):
> Jack
||||||||||
Jack's turn:
1
|||||||||
John's turn!
> 2
|||||||
Jack's turn:
2
|||||
John's turn!
> 1
||||
Jack's turn:
3
|
John's turn!
> 1
Jack won!
```

Example 2:

```
How many pencils would you like to use:
> 6
Who will be the first (John, Jack):
> John
||||||
John's turn!
```

```
> 1
|||||
Jack's turn:
2
|||
John's turn!
> 2
|
Jack's turn:
1
John won!
```

See hint

Write a program

Report a typo

Code Editor IDE +100

```
import random


def number_of_pencils():
    text = input('How many pencils wou
    while True:
        if not text.isnumeric():
            text = input('The number o
            continue
        if (number := int(text)) <= 0
            text = input('The number o
            continue
        return number


def first_player(names):
    player_name = input(f'Who will be
    while not player_name in names:
        player_name = input(f"Choose b
    return names.index(player_name)
```

Continue

Solve again

Solutions (244)

🎉 Correct

Your practice is really paying off. Well done!

65 learners liked this problem. 18 didn't like it. **What about you?**

😍 🙂 😐 🙁 😡

Topics in stage

✓ Random module

Comments (109) Hints (17) Useful links (3) Solut

Help us develop a better product for you and other learners!

I agree to receive invites to research activities (interviews, surveys)

All courses
Top courses
Beginner-friendly
Career paths
Python
Java
JavaScript
Kotlin

Full catalog

Go
Android
C++
Generative AI
Math
Frontend
SQL and Databases
Data Science
Bioinformatics
Backend

DevOps
Data Analysis
Machine Learning
Drafts

Resources
Blog
University

Subscription
For Business
Pricing

Hyperskill
About
Careers
For Content Creators

Support
Help Center
Terms