13086 - Golden Ratio Overheat

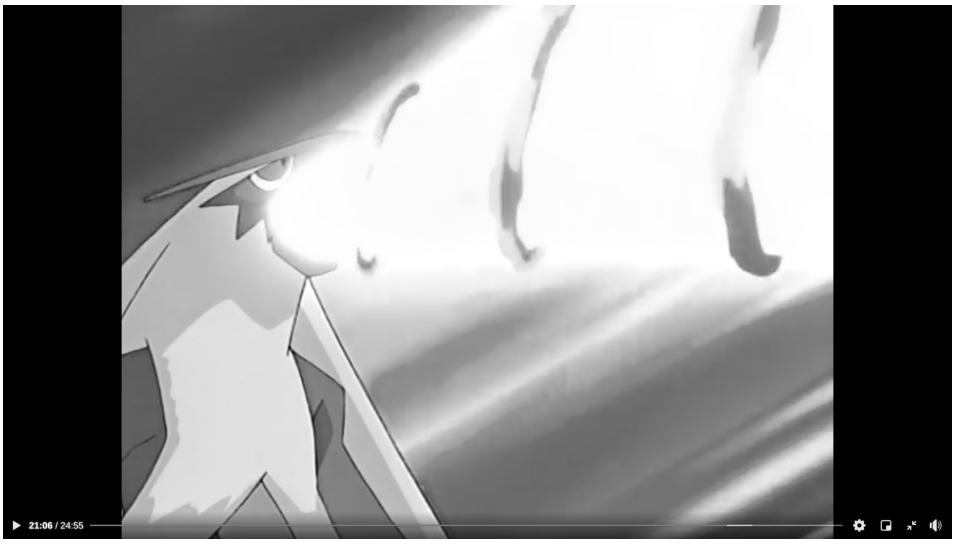
Status (/status/?pid=13086) | Limits Submit (/users/submit/13086)

Description

Last Pokemon Contest (神奇寶貝華麗大賽 (https://wiki.52poke.com/wiki/%E5%AE%9D%E5%8F%AF%E6%A2%A6%E5%8D%8E%E4%B8%BD%E5%A4%A7%E8%B5%9B)) between Satoshi (小智 (https://wiki.52poke.com/wiki/%E5%B0%8F%E6%99%BA%EF%BC%88%E5%8A%A8%E7%94%BB%EF%BC%89)) and Haruka (小遙 (https://wiki.52poke.com/wiki/%E5%B0%8F%E9%81%A5)) ! After Satoshi won the victory in Battle Pyramid, Satoshi and Haruka signed up for an unofficial Pokemon Contest in Terracotta Town.

Pokemon Contest is a contest in which trainers and Pokemons coordinate strength and beauty. It is divided into two stages. In the first stage, the Performance Stage, trainers and their Pokemons will perform moves to showcase their styles and skills. In the second stage, the Battle Stage, trainers battles against each other as well as demonstrating charm of their Pokemons.

Satoshi and Haruka met in the Battle stage, where Satoshi's Sceptile (蜥蜴王 (https://wiki.52poke.com/wiki/%E8%9C%A5%E8%9C%B4%E7%8E%8B)) fought against Haruka's Blaziken (火焰雞 (https://wiki.52poke.com/wiki/%E7%81%AB%E7%84%B0%E9%B8%A1)). In the final ten seconds of the battle, Haruka's Blaziken activates Blaze (猛火 (https://wiki.52poke.com/wiki/%E7%8C%9B%E7%81%AB%EF%BC%88%E7%89%B9%E6%80%A7%EF%BC%89)) and fires Overheat (過熱 (https://wiki.52poke.com/wiki/%E8%BF%87%E7%83%AD%EF%BC%88%E6%8B%9B%E5%BC%8F%EF%BC%89)). To battle in a dazzling style, the pattern of Overheat is cleverly designed with golden ratio:



The pattern of Overheat can be expressed with a string. First, Haruka and Blaziken design two short patterns F_0 and F_1. Then they generate longer patterns using the following recurrence formula: $F_{n-2} + F_{n-1} = F_n$ (**+** means string concatenation). Finally, Haruka tells Blaziken a number **n**, and Blaziken fires the Overheat with pattern F_n.

Given F_0, F_1, **n**, and **k**, please find out the **k**-th character in F_n.

> For full episode, please google "AG191 Satoshi vs. Haruka! Last Battle!!" or refer to this page (https://wiki.52poke.com/wiki/%E5%AE%9D%E5%8F%AF%E6%A2%A6_%E8%B6%85%E4%B8%96%E4%BB%A3_%E7%AC%A(

Explanation on Sample IO

All test data provide same F_0 = "abc", F_1 = "def". We can generate the following patterns:

- F_2 = "abcdef"
- F_3 = "defabcdef"
- F_4 = "abcdefdefabcdef"

The first test data asks **0**-th character of F_0, therefore output 'a'. Similarly, the outputs for the second to fifth test data are 'd', 'f', 'f', 'e', respectively.

Input

The input contains multiple test data. The first line of input contains an integer **T**, being number of test data. After that are **T** test data.

The first line of test data contains two non-empty strings F_0 and F_1.

The second line of test data contains two integers **n** and **k**.

- **1 <= T <= 100**
- **F0** and **F1** contain only lower case alphabets (`a-z`) and have length no greater than 2000.
- **0 <= n < 60** and **0 <= k < |Fn|**, where **|Fn|** is length of **Fn**

Output

For each test data, output the answer in a single line. Remember to add new line in the end.

Sample Input

Download (data:text/plain;charset=utf-8,5%0D%0Aabc%20def%0D%0A0%200%0D%0Aabc%20def%0D%0A1%200%0D%0Aabc%20def%0D%0A2%205%0D%0Aab

```
5
abc def
0 0
abc def
1 0
abc def
2 5
abc def
3 8
abc def
4 7
```

Sample Output

Download (data:text/plain;charset=utf-8,a%0D%0Ad%0D%0Af%0D%0Af%0D%0Ae)

```
a
d
f
f
e
```

Discuss