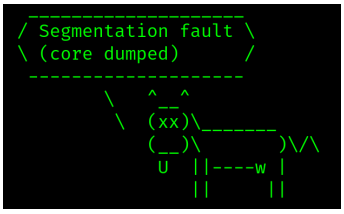
Problem K
Karl Coder



Karl is an aspiring C programmer, and is excited by the risks and rewards of low-level manual memory management. In the program he currently develops, he stores a string containing N non-zero bytes into a buffer named "`buf`". By mistake he accidentally made the buffer $2N$ bytes in size. The last N bytes of the buffer consists of only zero-bytes.

Now Karl needs to know the value N, the size of the string, in a separate part of the program. Traditionally you would recover the length of a string using the `strlen`-function, which reports the position of the first zero-byte in the provided buffer using a linear scan. However, Karl finds that this is much too slow, and that it defeats the advantage of using C in the first place. Can you help Karl efficiently recover N without crashing his program?

index	···	-1	0	1	2	3	4	5	6	7	8	···
buf	···	×	78	67	80	67	0	0	0	0	×	···

The contents of the buffer in sample interaction 3 are shown here.

Interaction

This is an interactive problem where interaction proceeds in rounds. In each round your program can attempt to read a byte from the buffer or report the answer (the value of N):

read: Write a line containing "`buf[`i`]`" if you want to try to read byte i of the buffer. If $0 \leq i < 2N$, then you will get the value of the byte stored in the buffer at index i. If $i < N$ then this will be an integer between 1 and 255, otherwise it will be 0. If $i \geq 2N$ or $i < 0$, you will get the response "`Segmentation fault (core dumped)`" and your program should exit.

answer: Write a line containing "`strlen(buf) =` M" if you want to report that you think that $N = M$. Afterwards your program should exit. Your submission will be accepted if you answered correctly, if you did not trigger a segmentation fault, and if you did not attempt to read entries from the buffer too many times.

A maximum of $2\lceil \log_2 N \rceil$ reads can be made. If your program attempts to read from the buffer after reaching the limit, it will get the response "`Too many reads`" and your program should then exit.

It is guaranteed that $2 \leq N \leq 10^{18}$.

Read	Sample Interaction 1	Write
		`buf[1]`
`65`		
		`buf[2]`
`0`		
		`strlen(buf) = 2`

Read **Sample Interaction 2** **Write**

```
                                buf[1]
50
                                buf[5]
0
                                buf[7]
Segmentation fault (core dumped)
```

Read **Sample Interaction 3** **Write**

```
                                buf[0]
78
                                buf[1]
67
                                buf[2]
80
                                buf[3]
67
                                buf[4]
Too many reads
```