Consider finding minimum element in an array of n elements.

What's wrong with starting with the following list when n = 3?

```
min is
A[0] = {0, 2, 3
        1, 3, 2
        2, 1, 3
        2, 3, 1
        3, 1, 2
        3, 2, 1}
```

Same min
A[1]

Same min
A[2]
Adv can have a list \( L \) with \( N \) inputs where min element is in a different position.

\[
\implies \text{# comparisons to find min} \geq \lceil \log_2 n \rceil
\]
L

others 2, 3, 4, 5, 6

1- - - - - -
- - - - - -
- - - ( - - -
- - - - -
- - - - - - - 1
Different adversary strategy

Possible positions for min:

0, 1, 2, 3, ... n-1

Any question can be answered to remove only one possible candidate for min.
possible mins

\[ 1, 2, 3, \ldots, n-1 \]

\[ a_5 < a_7 \]

\[ \text{not possible min} \]

pos min < pos min: Answer "yes"

move larger to not possible min

pos min < not pos min: \[ \text{Yes} \]

not pos min < not pos min: be consistent w/ past
there's n elements initially in possible mins

Each comparison moves \( \leq 1 \) to not possible min.

How many times do we subtract 1 from n until only 1 left.

\( n - 1 \)