We will do binary search. Let the variable l store the level such that level l, we are searching a range of $n/2^l$ (so it means at level 1, we are searching half the range, at level 2, we are searching $1/4$ of the range). The variable $b(j)$ stands for the beginning of the range at level l, and the variable $e(j)$ stands for the end of the range at level l. So, When $l = 0$, $b(0) = 1$, and $e(0) = n$.

Our algorithm is basically we will trust the Liar, so we will still keep searching during the YES region when we ask the for the 2 ranges (begin, mid) and (mid, end). However, if he say "NO" for both regions, we will go back 1 level as he has lied somewhere previously.

Repeat the following procedure Q times, or until you guess the answer:

```
if (l < log(n))
   m = (e(l) - b(l))/2; //middle of the range
   Ask q1 = [b(l), m]; //first question is on the first half
   Ask q2 = [m+1, e(l)]; //second question is on the second half
   if q1 == YES
      l++; //go next level
      b(level) = b(level-1) and e(level) = m; //Set up range for the next level
   else if q2 == YES
      l++; //go next level
      b(level) = m+1 and e(level) =e (level-1); //Set up range for the next level
   else //both q1 and q2 == NO, we will go back 1 level and count the lies
      add 1 lie here;
      l--; //go back 1 level
else if (l == log(n)) //The range == 1 here
   Ask q = is it your elements;
   if q == YES
      then guess++;
   if (guess == log(n))
      GUESS and exit;
   if q == NO
      add 1 lie here.
      guess = 0;
      level-- //go back 1 level
```

I can lie $< \log n$ times, so you decrease your level at most $\log n$ times. Whenever you ask 2 questions, you increase your level by 1, unless the range $= 1$. Thus, by the end of $6 \log n$ questions, it must be the case that you have asked at least $2 \log n$ queries where the range $= 1$ (think about it, $2 \log n$ will move you forward $\log nlevel$, you go back at most $\log n$, then you move down $\log n$ again, left with $2 \log n$ where the range $= 1$).

Moreover, at most $\log n$ of those queries can have been lies where range $= 1$ (and we have already counted the non-lies where range $= 1$ that cause you to decrease your level). So there must be **at least** $\log n$ queries where range $= 1$ that are not lies and that do not cause you to decrease your level— i.e., those are correct queries that lead you to guess the correct answer. THIS IS YOUR ANSWER. (seems like can do in $4 \log n$ if this is done more carefully, but not sure how.