Q ⇒ Given a perfect square value S (ex → $16129 = 127^2$) between 1 to 10^9, find it's square root.

Ex $S = 16129$

ans → 127

Discrete Binary Search or Binary Search on Ans

The search space that was demonstrated using a list is A sequence of numbers that represent some function

A lot of times those graphical representation need not to come from an actual sequence of data.

If we have a problem which can depict the search space into a monotonic function → then we can directly apply B.S on that search space without actual data.

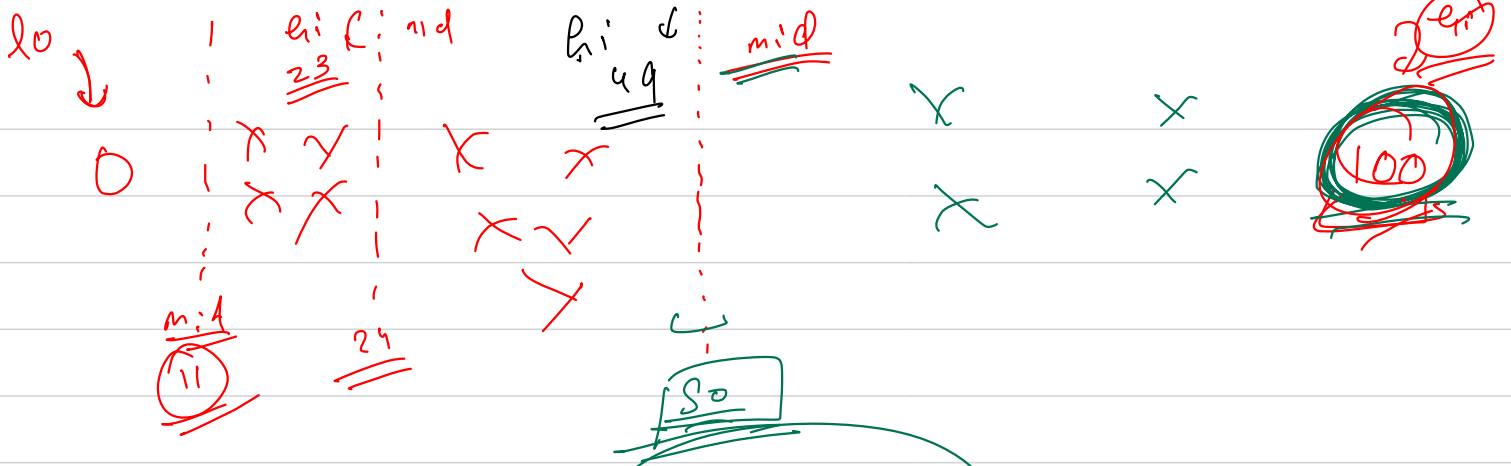
monotonic function → increasing / decreasing

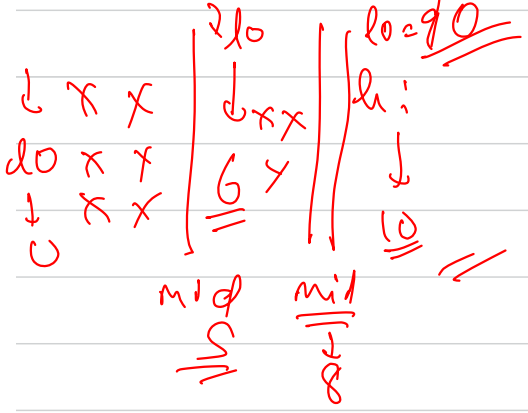
$f(x) = \sqrt{x}$ → Yes

$f'(x) = \frac{1}{2\sqrt{x}} > 0$ → Monotonic function

$\frac{d(f(x))}{d(x)}$

$\int f(x)$ → x^{th} prime number → No





So

mid * mid == n ??

if mid^2 is greater than n, then for every key to the right square will be greater than n

we wanted to find the best suitable ans in the function $f(x) = \sqrt{x}$ (monotonic func^n)

Binary Search

the solution we just implement guess floor($\sqrt{x}$)

TC $\rightarrow$ O(log n)

SC $\rightarrow$ O(1)

36

32

(6.----), 6

lo 4 — 5 — hi 7

0

ans ← 5

mid 3 — mid 8

Candidate ← 3

$3*3 < 37$

hi 6

12

mid 6

37

mid

lo 6 — hi 7

ans ← 6

first number just less than $\sqrt{37}$

$6*6 > 37$

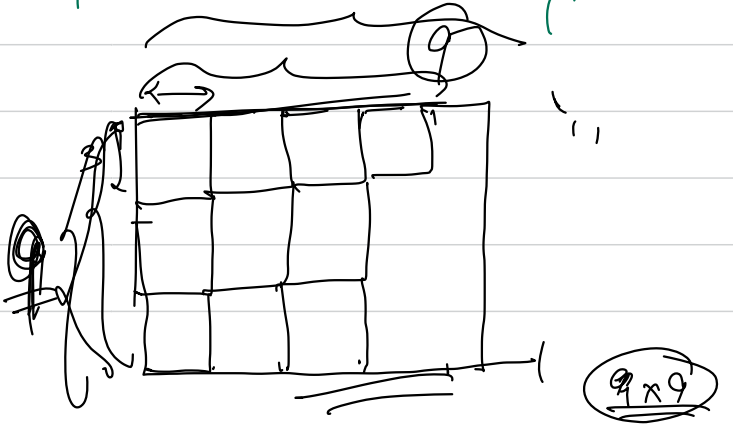
$6*6 < 37$

Q => There are n rectangles of same size (w x h)
w → width, h → height. We is required to find a square of the smallest size into which these rectangles can be packed. We can't rotate rectangles. Find the side of the smallest square.

Ex
w → 2
h → 3
n → 10

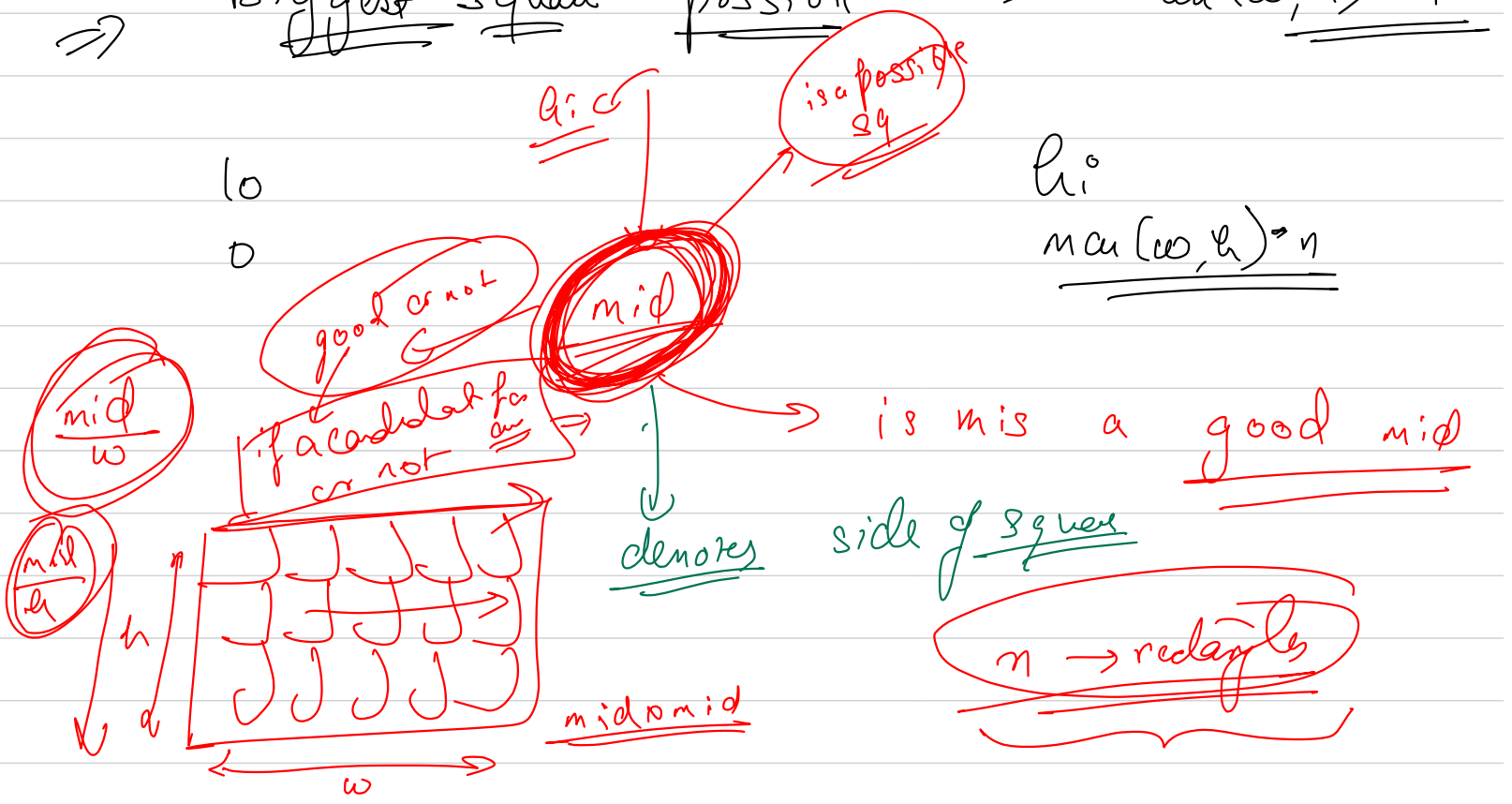
Ans → 9

$w, h, n \leq 10^9$



mi → not possible → 0

⇒ Biggest square possible → max(w, h) * n



To count the no of rectangles that fits into mid × mid Square.

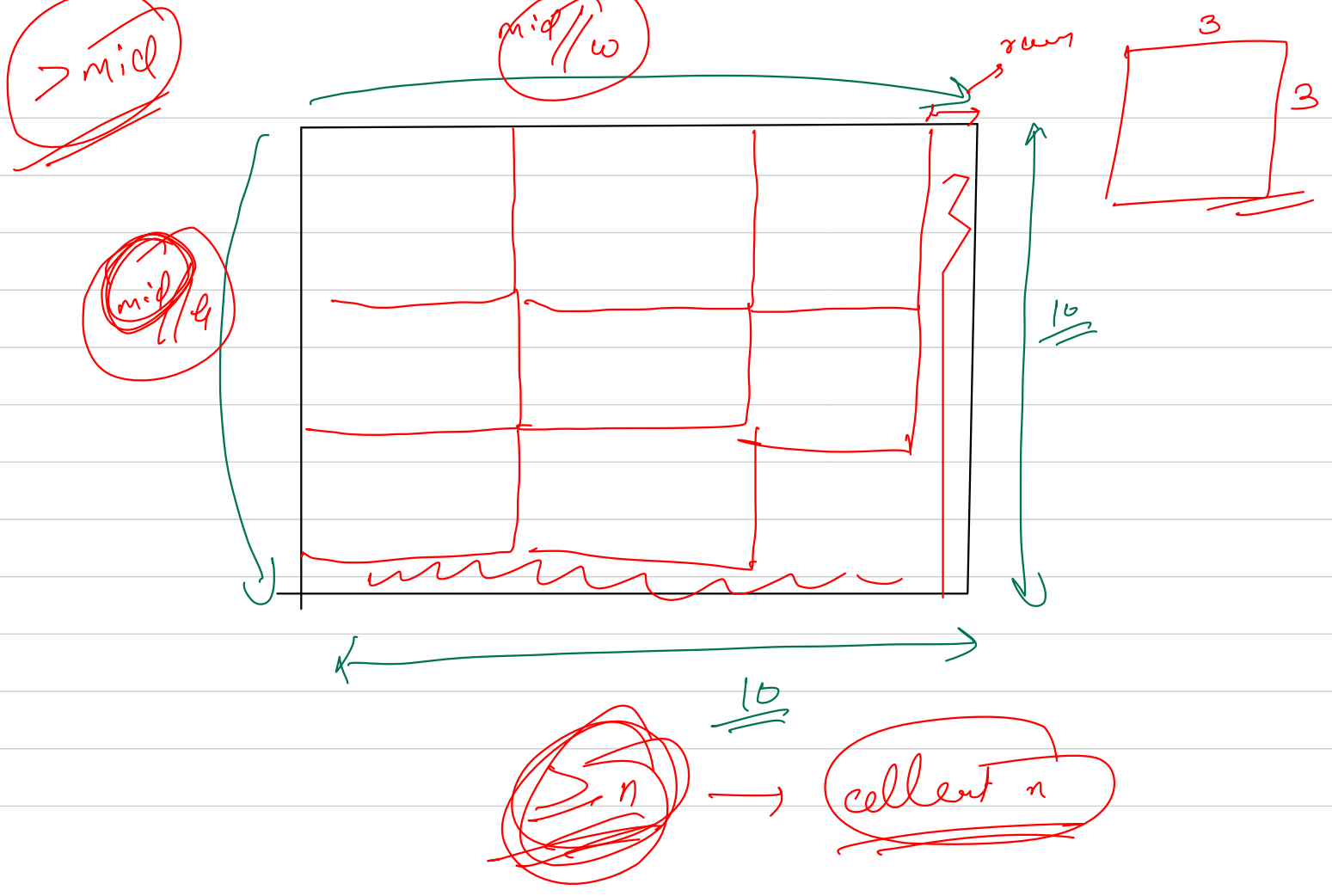
= product of no of rectangles that fit on the first side & no of rectangles that fit on the other side

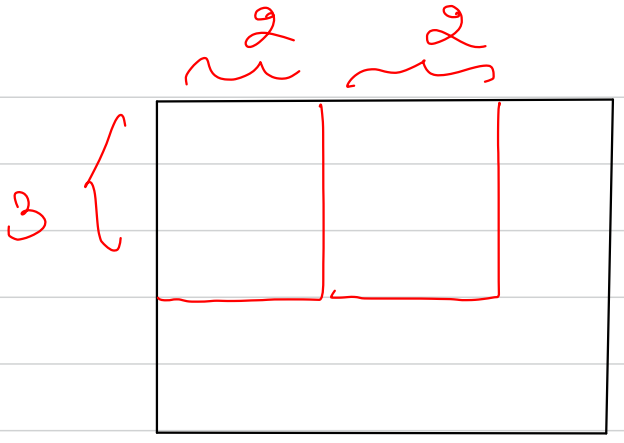
if this

$(mid // w) * (mid // h) \Rightarrow n$

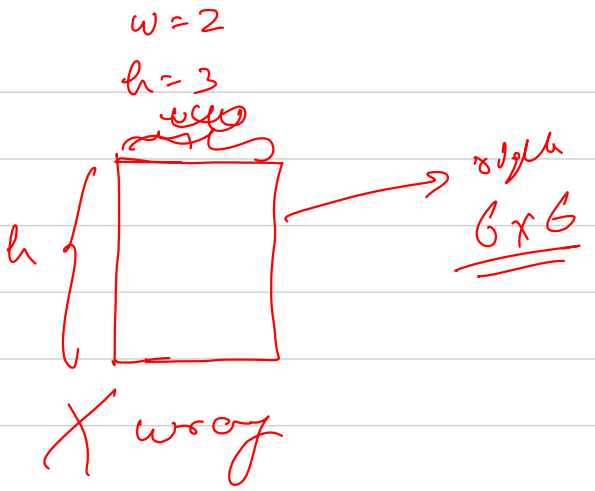
then this is good mid

atleast includes n


> mid
mid // w
row
3
3
mid // h
16
16
n
collect n



5x5



TC → $\log(\max(w,h)*n)$

SC → O(1)

$w, h \le 10^9$

$n \le 10^9$

Q. You have a practical, and you went to the photocopy shop to buy 'n' copies of a page.

There are 2 machines at the shop, first one copies a sheet in x second, & the second one is copying 1 page in y second.

You can run both machines parallelly & can create photocopy of both original & copied page.

Find the min time reqd to make n copies.

$n \leq 10^8$, $(x, y) \leq 10$ ⟶ $n = 4$ $x = 1$ $y = 1$

ans → 3

$n = 5$, $x = 1$ $y = 2$ → 4

min time

↳ lo

⇒

to create n copies

1 di

max (x, y) * n

↳ the machine that takes more time we make all copies by it

→ we will check if it is a good mid? → if it is possible to prin n copies in mid seconds

1st copy will be done separately

1st machine → mid / x copies

2nd machine → mid / y copies

mid/x + mid/y + 1 ≥ n