

* Longest Increasing Subsequence : Const
 $1 \leq n \leq 10^5$

app :- find all the subsequence ending at i th ind.

Brute (generate all the increasing subsequences)

Step by
Step
Solution :-

[i]

now to find the len of longest increasing subseq. ending at this ind (i).

we need to go from $0 \rightarrow i-1$

① to see which subsequence could elem $[i]$ be a part of

② if not, start a new inc subsequence from elem $[i]$

acc to this app
 $P.C = O(n^2)$

But acc to constraints :-

we need a soln with $O(n \log n)$.

MI \rightarrow if we could figure out for ind i

that which subsequence

could $elem[i]$ be a part of, in $O(\log n)$

could be BS?

How?

Thought process

assume soln we already have till $i-1$ ind

is then see how could we apply BS for the i th ind.

So what we need?

\rightarrow len of LIS ending at i th ind

\rightarrow && we cannot keep a track of all the LIS ending at every ind.

as space ~~is still $O(n^2)$~~

9.7

8.

(i-1)

(i)

Date
Page

7 3 5 3 6 2 9 8

So len of LIS ending at i

6 attached at end

3 5 6 → ans = 3

5 6 → 2

3 6 → 2

or some of r

In which we've a longest LIS of len say 4.

but ending elem = 8.

LIS

len = 4

2

5

~~6~~

here

if 7 & 8 encountered

8

~~UB~~

then scan ptr (7 & 8)

→ & we encountered 6 later

So In above

LIS we could put 6

So ~~LIS~~

LIS will now'd still be above

only 2

But keep the 6 but it could later make a bigger

what to do?

there later

if some

elem encountered

put it

to its

place

NOT
is the
LIS

→ But we're concerned about the Ten. So it'd give correct ans.

new 6 belongs here
↓
nothing
↓
2 3 5 6 0
↑
take this
from other
Subs.

for every elem, do a
BS on vector \rightarrow to find
~~upper~~ bound ind of
elem $T(i)$
& put it there.

→ do new
US len
 ≥ 5 .

~~vector $\{x_i\}$~~
↓
~~At this position~~

~~dp~~ ↓
this way an elem i
↓
can use previous parts
of LIS → i.e. elems smaller
than it ↓
for its own LIS