

Q.5) Algorithm proof

The two main conditions to analyse are \rightarrow ① if ($a \leq x \rightarrow \text{data}$)
 \rightarrow ② if ($x \rightarrow \text{data} \leq b$)

① As soon as $a > x \rightarrow \text{data}$, the recursion stops.

Till this point, the code has essentially just been searching for a .

\therefore Time to get to this point is $O(\log n)$
{ Searching is $O(\log n)$ }

② Very similar argument to ①, it stops when $b < x \rightarrow \text{data}$.

Again, the similarity with searching for b essentially gives time to get till here as $O(\log n)$

In searching for these two extremes, the algorithm traverses through all the m required elements. It prints these m elements the moment it encounters them. m print operations gives the code $O(m)$ since this much HAs to be done no matter what.

Combining all of these aspects, we can say that

the overall time complexity is of the order

$$\underline{O(m + \log n)}$$