**Algorithm to depict Binary Search**

0 1 2 3 4 5 6

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 8 | 14 | 24 | 39 | 54 | 72 | 89 |

Lower bound mid-element Higher bound

* Let’s say we have an array like this and now we want to search a desired element from the list.

Say the desired element is **54**.

* Now, find the mid element likewise: (low + high) // 2 = (0 + 6)//2 = 3
* And at index 3 we have our element “39”
* So now we’ve to shift our bounds.

But the question is which one lower or upper?

* Here we’ll check whether the element we are searching for is greater than the mid element or smaller?!
* So we are searching for **54** and our mid element is **39 viz 39<54.**
* If the value we are searching for is smaller than the mid-element then change the upper bound.
* And if the value we are searching for is greater than the mid-element then change the lower bound.

So here we’ll change the lower bound to the mid-element.

0 1 2 3 4 5 6

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 8 | 14 | 24 | 39 | 54 | 72 | 89 |

Lower bound Higher bound

* Again we’ll find the mid\_value in this list between lower bound and upper bound.

So, (3 + 6) // 2 = 4

So, our new mid\_value is at index 4 viz = **54**.

* Again is it matching with the search value? Yes!

Hence, we found our desired element.