

28/12/2020.

Q. Write logic for Ternary search?

⇒ Ternary Search.

$$\text{mid}_1 = \underline{1} + (r-1)/3$$

$$\text{mid}_2 = r - (r-1)/3$$

	0	1	2	3	4	5	6	7	8	9
Search	1	2	3	4	5	6	7	8	9	10

6

	0	1	2	3	mid ₁	5	6	mid ₂	7	8	9	
	1	2	3	4	5	6	7	8	9	10		key > ar[mid ₁] & key < ar[mid ₂] r = 10

l = 0

	0	1	2	3	4	mid ₁	5	mid ₂	6	7	8	9	
	1	2	3	4	5	6	7	8	9	10			key > ar[mid ₁] & key < ar[mid ₂] l = 4 r = 6

	0	1	2	3	4	mid ₁	5	6	7	8	9	
	1	2	3	4	5	6	7	8	9	10		key = ar[mid ₁]

key found
l = 5
r = 5

⇒ Time Complexity : $O(\log_3 n)$

Program

```
#include <stdio.h>

int ternarySearch(int l, int r, int key, ar[])
{
    if (r >= 1) {
        int mid1 = 1 + (r - 1) / 3;
        int mid2 = r - (r - 1) / 3;
        if (ar[mid1] == key) {
            return mid1;
        }
        if (ar[mid2] == key) {
            return mid2;
        }

        if (key < ar[mid1]) {
            return ternarySearch(1, mid1 - 1,
                                key, ar);
        }
        else if (key > ar[mid2]) {
            return ternarySearch(mid2 + 1, r,
                                key, ar);
        }
        else {
            return ternarySearch
        }
    }
    return -1;
}
```

