Element Comparison of Recursive Binary Search Algorithm:

int BSearch(int A[], int low, int high, int key){

if (low>high)

return -1

mid = $\left\lfloor \frac{low+high}{2} \right\rfloor$

if (A[mid]==key)

return mid;

else if (A[mid] > key)

$E(\frac{n}{2}) \longrightarrow$ return BSearch (A, low, mid-1, key)

else

$E(\frac{n}{2}) \longrightarrow$ return BSearch (A, mid+1, high, key)

}

$E(n)$

Recurrence for element comparison of Binary search:

$$E(n) = \begin{cases} 1 & n=1 \\ E(\frac{n}{2})+c & n>1 \text{ and } n=2^k \end{cases}$$

Solution

$$E(n) = E\left(\frac{n}{2}\right)+c$$
$$= E\left(\frac{n}{2^2}\right)+c+c$$
$$= E\left(\frac{n}{2^2}\right)+2c$$
$$\vdots$$
$$= E\left(\frac{n}{2^k}\right)+kc$$
$$= E(1)+c\log_2 n$$

$n=2^k$

$\Rightarrow \log_2 n = k$

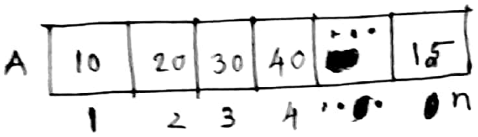
$= c \log_2 n + 1$

$= O(\log_2 n)$

$$\left[\begin{array}{l} E(n) = 1 \quad \text{when} \quad n = 1 \\ \Rightarrow E(1) = 1 \end{array} \right]$$

Element Comparison of Linear Search:

Best case:



key = 10

Element Comparison $= O(1) = 1$

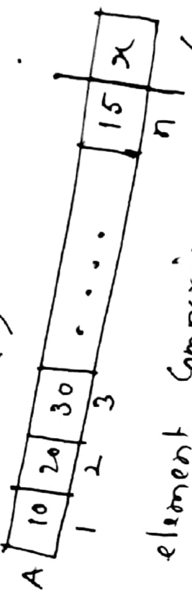
Worst case:

key = 15

Element comparison $= O(n) = n$

OR

element is not present in array $O(n)$



key = x

element Comparison = n+1 (Not found)

Average Case:

$$\text{Average case} = \frac{\text{Best} + \text{Worst}}{2}$$

$$= \frac{1+n}{2}$$

Space complexity of an algorithm:

```
int A[100], i, sum, n;
scanf("%d", &n);
sum=0;
for(i=0; i<n; i++) {
    scanf("%d", &A[i]);
    sum = sum+A[i];
}
printf("%d", sum);
```

space for i = 4 bytes
space for sum = 4 bytes
space for n = 4 bytes
Since n elements are stored in array A and A is integer array.
So space for A array = 4n

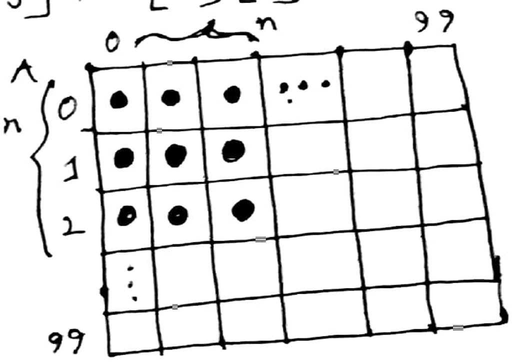
Total space = 4n+4+4+4
= 4n+12 = $O(n)$

Space complexity of an algorithm


```
int A[100][100], B[100][100], C[100][100];
int i, j, k;

for(i=0; i<n; i++){
    for(j=0; j<n; j++){
        c[i][j] = 0;
        for(k=0; k<n; k++){
            c[i][j] = c[i][j] + A[i][k] * B[k][j];
        }
    }
}
```

n=3



Variable	Total elements	Required Space
A	$n*n$	$4n^2$
B	$n*n$	$4n^2$
C	$n*n$	$4n^2$
i	1	$4*1$
j	1	$4*1$
k	1	$4*1$

$$\text{Total Space} = 4n^2 + 4n^2 + 4n^2 + 4 + 4 + 4$$
$$= 12n^2 + 12$$
$$= O(n^2)$$