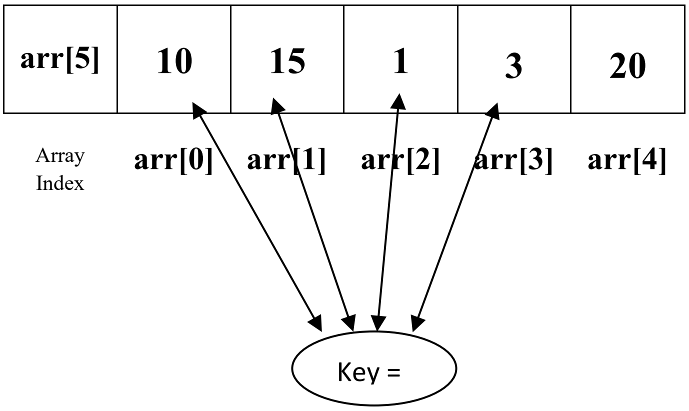
ARRAYS

Searching an element in an array

- One of the applications of the single dimensional array is searching.
- Searching is the process of finding the given key element and its position in the array of elements.
- Simple methods of searching are:
  - Linear Search.
  - Binary Search.

Linear Search / Sequential Search

- It is one of the easiest ways of searching the given key element in an array in linear order, i.e., by comparing the key element with every element of an array one after another.
- On comparison, if the key element is found then the search is successful else unsuccessful.
- Linear search is applicable for both sorted array and unsorted array.
- **Example for Linear search:-**



- From the above example we can notify that key element is compared with every elements of the array and the comparison is stopped only when the key matches the element.
- In the above example comparison is stopped at arr[3] because key is same as element at array.
- Once the element is matched print assuccessful search with its position.

Logic - Input: a[10] –(array), Key (Element to be searched), flag = -1;

```
for(i=0;i<n;i++)
{
    if(key==a[i])
     {
        flag=i;
        break;
    }
}
if(flag>=0)
printf("Successful Search\n %d element found at %d position",key,flag+1);
else
printf("Unsuccessful Search- Element not found\n");
```

Program:

```
#include<stdio.h>
#include<conio.h>

void main()
{
int n, key, a[20], i, flag=-1;

printf(“Enter the number of elements\n”);
scanf(“%d”, &n);

prinf(“Enter the %d elements to an array\n”, n);
for(i=0; i<n;i++)
scanf(“%d”, &a[i]);

printf(“Enter the key element to be searched\n”);
scanf(“%d”, &key);

for(i=0; i<n;i++)
   {
if(key==a[i])
       {
          flag=i;
          break;
        }
     }
if(flag>=0)
     printf("Successful Search \n %d element found at %d
     position\n",key,flag+1);
     else
     printf("Unsuccessful Search- Element not found\n");
getch();
}
```

Advantages of Linear search

- Very simple approach
- Works well for small array.
- Used to search when the elements are not sorted.

Disadvantages of Linear search

- Less efficient if the array size is large.
- If the elements are already sorted, linear search is not efficient.

Binary Search

- ✓ The main disadvantage of the liner search is time consuming in deciding the presence of the element in the array; it can be over taken by using binary search.
- ✓ Binary search can be applied only on the ***Sorted array.***

Steps for binary search are:

1. Note the position of first and last element of the array.
2. Find the position of the middle element.
3. Compare the element in the middle position with the key element to be searched.
4. After comparison following any one case will be resulting-
   a. If the key element is same as the middle element the note the position and stop the search.
   b. If the key element is greater than the middle element then continue the search to right portion of the middle element.
   c. If the key element is less than the middle element then continue the search to left portion of the middle element.

Example:

	low		mid		high
arr[5]	**10**	**15**	**20**	**30**	**45**
Array Index	**arr[0]**	**arr[1]**	**arr[2]**	**arr[3]**	**arr[4]**

Logic:

```
flag=-1,low=0;
high=n-1;
     while(low<=high)
     {
      mid=(low+high)/2;
       if(key==a[mid])
       {
          flag=mid;
          break;
       }
       if(key<a[mid])
          high=mid-1;
       else
          low=mid+1;
     }
if(flag>=0)
printf("Successful Search\n %d element found at %d position",key,flag+1);
else
printf("Unsuccessful Search- Element not found\n");
```

Program:

```
          #include<stdio.h>
          #include<conio.h>
          void main()
          {
          int n, key, a[20], i, low, high, mid, flag=-1;

          printf(“Enter the number of elements\n”);
          scanf(“%d”, &n);

          prinf(“Enter the %d elements to an array\n”, n);
          for(i=0; i<n;i++)
          scanf(“%d”, &a[i]);

          printf(“Enter the key element to be searched\n”);
          scanf(“%d”, &key);

          low=0;
          high=n-1;
               while(low<=high)
                      {
                         mid=(low+high)/2;
                         if(key==a[mid])
                           {
                              flag=mid;
                              break;
                           }
                         if(key<a[mid])
                           high=mid-1;
                         else
                           low=mid+1;
                      }
          if(flag>=0)
               printf("Successful Search \n %d element found at %d
               position\n",key,flag+1);
          else
               printf("Unsuccessful Search- Element not found\n");
getch();
}
```

Advantages of binary search:

- Simple Technique
- Very efficient searching technique

Disadvantages of binary search:

- The list of elements to be searched should be sorted.
- It is necessary to find the middle elements to search any key in the given list.