Operations on Array

Following are the basic operations supported by an Array :-

Traverse Print all the array elements one by one.

Insertion Add one element at given index.

Deletion Delete an element at given index.

~~Search~~ **Search** → Search an element using given index or by values.

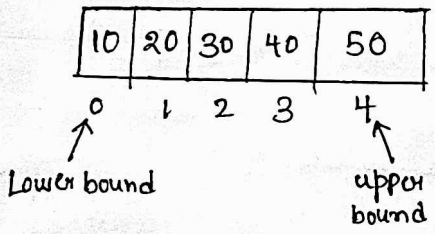
- Linear Search
- Binary Search

Sorting → Sorting the data in Ascending or Descending order.

Traversal on Array

If we want to print or count the element of linear array. That can be achieved by traversing of array.

Logical explaination



Algorithm

(1)
1. Set K = LB
2. Repeat Step 3 and 4 while K ≤ UB
3. Apply Process to LA[K]
4. Set K = K+1 or K++
5. exit.

(2)
1. Start a Loop from 0 to N-1.
   for(i=0; i<N; i++)
2. Access every element of array
   arr[index]
3. Print the elements
4. exit.

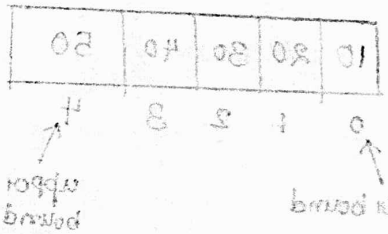
C program / C++ program to traverse the array

ALGORITHM - ①

```
#include <iostream.h>
using namespace std;

int main()
 {
   int arr[5]={10,20,30,40,50},k;

   k=0;
   while (k<=4)
     {
       cout<<arr[k];
       k=k+1;
     }

 }
```

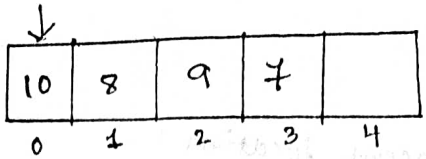
ALGORITHM ②

```
#include <iostream>
using namespace std;

int main()
 {
   int arr[5]={10,20,30,40,50};
   int k;
   for (k=0;k<=4;k++)
     {
       cout<<arr[k]<<endl;
     }

 }
```

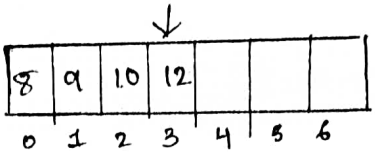


Insertion operation of an Array

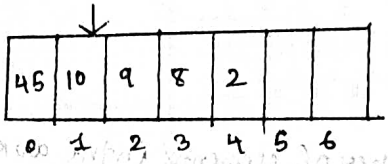
(a) Insertion at beginning



(b) Insertion at end

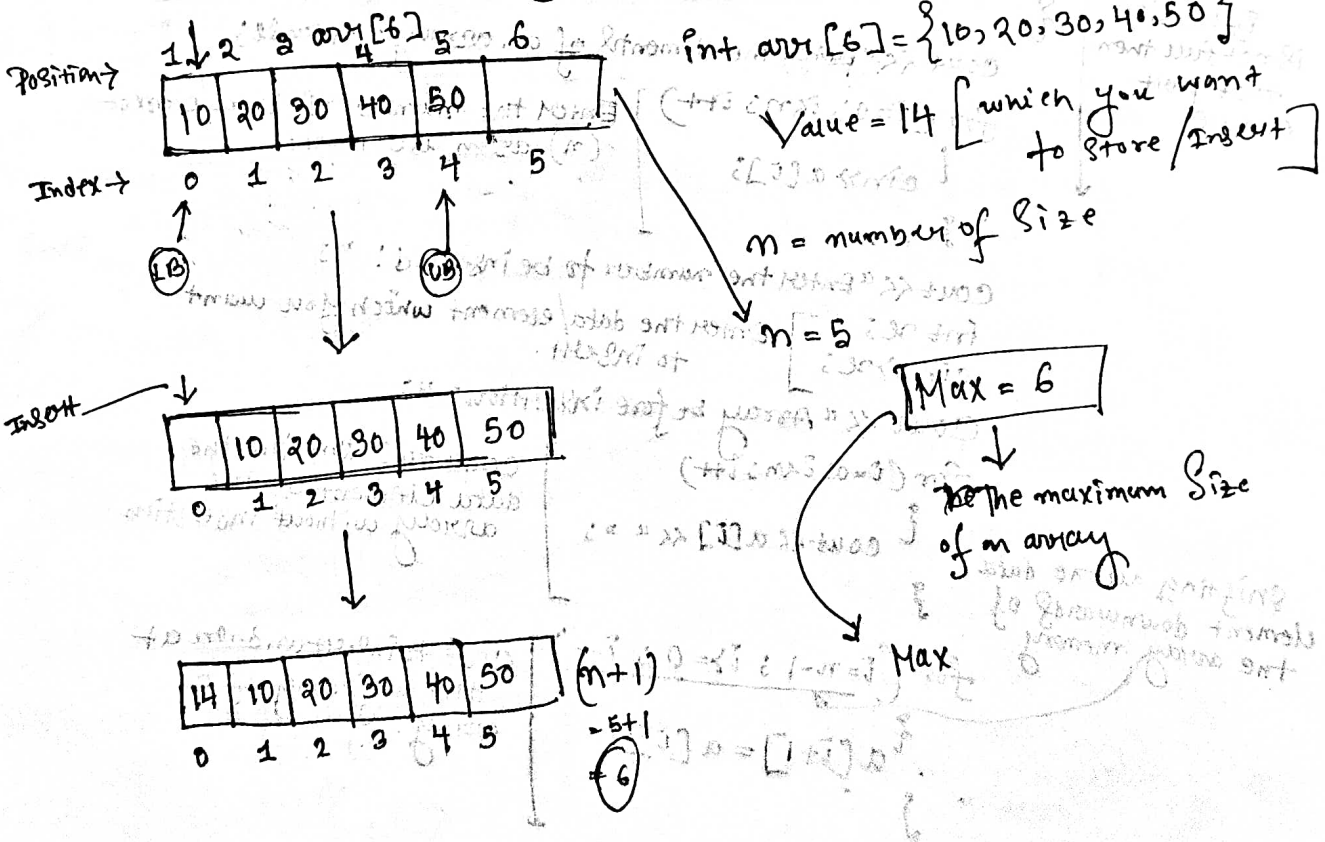


(c) Insertion at a Specific Location



Insertion at the beginning of an Array

When we want to store any data element or value at the beginning or any starting index of an array that time the operation is called as Insertion at the beginning.



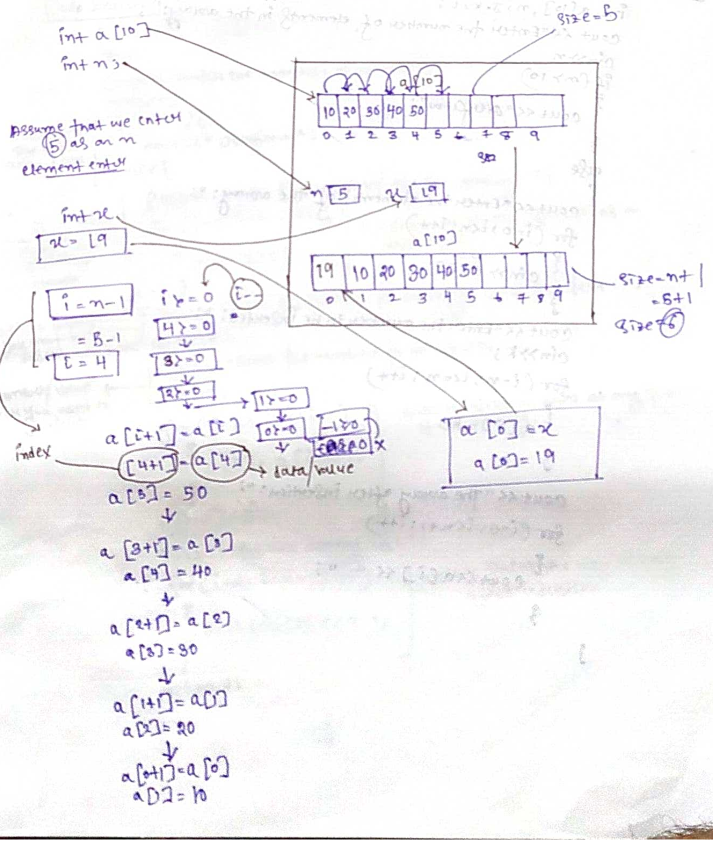
ALGORITHM

begin
IF N = MAX, return
ELSE
    N = N+1
    For All Elements in A
        More to next adjacent location
    A[FIRST] = New_Element
end

Implementation in C++

```
#include<iostream>
using namespace std;
int main()
{
    int a[10], n, i, j;
    cout << "Enter the number of elements in the array:";
    cin>>n;
    if (n>10)                              // Check if the size of the array is full or not
    {
        cout<<"overflow";
    }
    else                                   // If the size is not full then that part execute
    {
        cout<<"Enter the elements of an array: "<<endl;
        for (i=0; i<n; i++)                // Enter the number of data/element (n) as an user input.
        {
            cin>>a[i];
        }
        cout<<"Enter the number to be inserted: ";
        int x;                             // Enter the data/element which you want to insert.
        cin>>x;
        cout << "Array before insertion: ";
        for (i=0; i<n; i++)                // code to print the the data element of the array without insertion
        {
            cout<<a[i] << " ";
        }
        for (i=n-1; i>=0; i--)             // code to insert and data at the beginning of an array.
        {                                  // Shifting all the data element downwards of the array memory
            a[i+1] = a[i];
        }
        a[0] = x;                          // insert the Value of x in 1st index of the array
        cout << "The array after insertion: ";
        for (i=0; i<n+1; i++)              // cod to print the data element of array with insertion.
        {
            cout<<a[i]<< " ";
        }
    } //end of else
    return 0;
} //end of main()
```

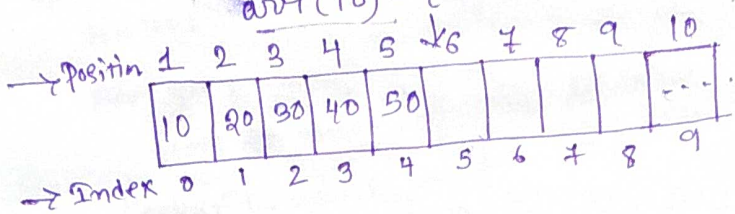
Visualization



Insertion at the end of an Array

when we want to store any data/element at the end of the array that 0 time the operation is called as insertion operation at the end of an array.

arr(10) = {10, 20, 30, 40, 50};

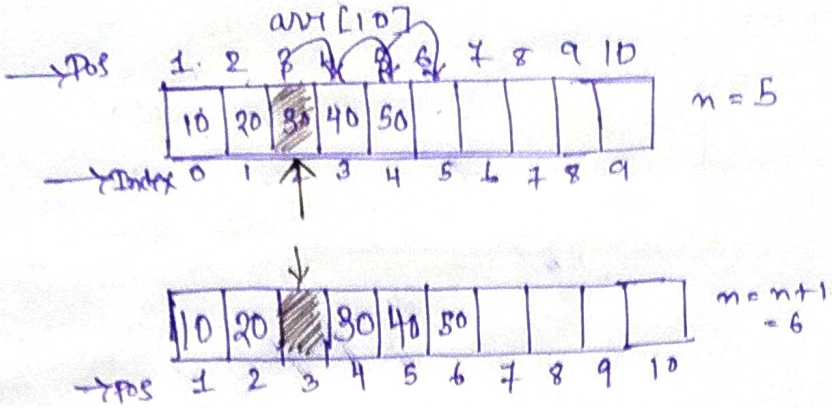


~~Code~~

Code Implementation

```
int a[10], n, j, k, i;
cout<<"Enter the number of elements in the array:";
cin>>n
if (n>10)
 { cout<<"overflow";
 }
else
 {
 cout<<"Enter the elements of the array:";
 for (i=0; i<n; i++)
  { cin>> a[i];
  }
 cout<<"Enter the number to be inserted:";
 cin>>k;
 for (i=n; i<=n; i++)
  {
  a[i]=k;
  }
 cout<<"The array after insertion:";
 for (i=0; i<n+1; i++)
  {
  cout<<a[i]<<" ";
  }
}
```

Insert at element at given Location



Code Example [C++]

```
int a[10], n, i, j, k;
Cout<<"Enter the numbers of elements in the array: ";
cin>>n;
if(n>10){
    cout<<"Overflow";
}else{
    Cout<<"Enter the elements of the array: ";
    for(i=0; i<n; i++)
    {
        cin>>a[i];
    }
    cout<<"Enter the position at which you want to insert: ";
    cin>>k;
    cout<<"Enter the number to be inserted: ";
    cin>>j;
    for(i=n; i>k; i--)
    {
        a[i]=a[i-1];
    }
    a[k-1]=j;
    cout<<"the new array is: ";
    for(i=0; i<n+1; i++)
    {
        cout<<a[i]<<" ";
    }
    return 0;
}
```

Check if the size of the array is full or not.

If the size is not full then that part execute.

Enter the no of data/element as an user input.

Position where you want to insert new value

Number/value which you want to insert

code to insert an data at any position of an array.

code to print the new array after insertion.