

```
// Author - Krishna Kumari
```

```
// ARRAY INITIALISATION
```

```
// CASE - 1 int aar[] = {1,2,3,4,5};  
// CASE - 2 int brr[5] = {13,24,23,223,12};  
// CASE - 3 int crr[5] = {1,4}; // other 3 blocks filled with 0  
// CASE - 4 int drr[2] = {1,5,3,6,3}; // ERROR
```

```
// Formula for finding value at any index in an array by calculating its address
```

```
    // (Base address + (data_type size) * index))
```

```
// Program to print double of an array through user input
```

```
// #include <iostream>  
// using namespace std;  
// int main()  
// {  
//     int arr[5];  
//     for (int i = 0; i<5; i++)  
//     {  
//         cout << "Enter the value in array at index " << i << " ";  
//         cin >> arr[i];  
//     }  
//     for(int j = 0; j < 5 ;j++)  
//     {  
//         cout << arr[j] * 2 << " ";  
//     }  
  
//     return 0;  
// }
```

```
//-----
```

```
// Linear Search in an array - is done to check whether given or targeted element  
// is present in the array or not.
```

```
// #include<iostream>  
// using namespace std;  
// int main()  
// {  
//     int arr[5]={1,2,3,4,5};  
//     int target;  
//     int flag = 0;  
//     cout << "Enter your target element: ";  
//     cin >> target;  
  
//     for(int i=0;i<5;i++)  
//     {  
//         if(arr[i] == target)  
//         {  
//             flag++;  
//             break;  
//         }  
//     }
```

```
//      }
//      if(flag == 1)
//      {
//          cout << "Element present in the array...";
//      }
//      else
//      {
//          cout << "Not found...";
//      }
//  }
```

```
//-----
// Array & Functions
```

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

// Whenever pssing an array always send the size/no of elements in the function
// void print(int arr[] , int size)
// {
//     for(int i = 0; i < size ; i++)
//     {
//         cout << arr[i] << ' ';
//     }
// }
// int main()
// {
//     int arr[5]={1,2,3,4,5};
//     int size = 5; // here size is the no of Elements
//     print(arr , size);
//     return 0;
// }
```

```
//-----
```

```
// linear search using function
```

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

// int linearsearch(int arr[] , int n , int target)
// {
//     for(int i=0;i<n;i++)
//     {
//         if(arr[i] == target)
//         {
//             return true;
//         }
//     }
//     return false;
// }
// int main()
// {
//     int arr[5]={1,2,3,4,5};
//     int n = 5;
//     int target = 4;
//     bool ans = linearsearch(arr , n , target);
```

```

//      if(ans == 1)
//      {
//          cout << "Element Found...";
//      }
//      else
//      {
//          cout << "Not Found...";
//      }
//}

//-----

// Count 0's and 1's in an array

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

// void countZeroOne(int arr[],int n,int countZero,int countOne)
// {
//     for(int i = 0; i < n; i++)
//     {
//         if(arr[i] == 0)
//         {
//             countZero++;
//         }
//         else if(arr[i] == 1)
//         {
//             countOne++;
//         }
//     }
//     cout << "Number of 0's are: " << countZero << endl;
//     cout << " Numebr of 1's are: " << countOne << endl;
// }
// int main()
// {
//     int arr[5]={0,1,1,0,0};
//     int n = 5;
//     int countZero = 0;
//     int countOne = 0;
//     countZeroOne(arr , n, countOne,countZero);
//     return 0;
// }

//-----

// Max and Min number in an array

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

// void MaxNumber(int arr[] , int n, int max)
// {
//     for(int i=0;i<n;i++)
//     {
//         if(arr[i] > max)
//         {
//             max = arr[i];
//         }
//     }
// }

```

```
//      cout << "maximum number in array is: " << max << endl;
// }
// void MinNumber(int arr[] , int n, int min)
// {
//     for(int i=0;i<n;i++)
//     {
//         if(arr[i] < min)
//         {
//             min = arr[i];
//         }
//     }
//     cout << "Minimum number in array is: " << min;
// }
// int main()
// {
//     int arr[5]={15,23,12,56,34};
//     int max = arr[0];
//     int min = arr[0];
//     int n = 5;
//     MaxNumber(arr , n ,max);
//     MinNumber(arr,n,min);
//     return 0;
// }
```