

# Assignment - 06

Submitted by- Vineet Giri

Submitted to- Mr Naman  
Chauhan

BCA year: 1st

Section: C

## Section 1: Basic Array and Operations

### Program 1: Input and Display 1D Array

```
#include <iostream>
using namespace std;
int main() {
    int arr[5];
    cout << "Enter 5 elements: ";
    for (int i = 0; i < 5; i++)
        cin >> arr[i];
    cout << "Array elements are: ";
    for (int i = 0; i < 5; i++)
```

```
    cout << arr[i] << " ";
return 0;
}
```

Output:

```
Enter 5 elements: 1 2 3 4 5
Array elements are: 1 2 3 4 5
```

## Program 2: Sum of All Elements in 1D Array

```
#include <iostream>
using namespace std;
int main() {
    int arr[5], sum = 0;
    cout << "Enter 5 elements: ";
    for (int i = 0; i < 5; i++) {
        cin >> arr[i];
        sum += arr[i];
    }
    cout << "Sum = " << sum;
    return 0;
}
```

Output:

```
Enter 5 elements: 1 2 3 4 5
Sum = 15
```

## Program 3: Find Maximum and Minimum in 1D Array

```
#include <iostream>
using namespace std;
int main() {
    int arr[5];
    cout << "Enter 5 elements: ";
    for (int i = 0; i < 5; i++) cin >> arr[i];
    int max = arr[0], min = arr[0];
    for (int i = 1; i < 5; i++) {
        if (arr[i] > max) max = arr[i];
        if (arr[i] < min) min = arr[i];
    }
    cout << "Maximum = " << max << "\nMinimum = " <<
min;
    return 0;
}
```

Output:

Enter 5 elements: 5 2 9 1 7

Maximum = 9

Minimum = 1

## Program 4: Reverse a 1D Array

```
#include <iostream>
using namespace std;
int main() {
    int arr[5];
    cout << "Enter 5 elements: ";
    for (int i = 0; i < 5; i++) cin >> arr[i];
    cout << "Reversed array: ";
    for (int i = 4; i >= 0; i--) cout << arr[i] << " ";
    return 0;
}
```

Output:

Enter 5 elements: 1 2 3 4 5

Reversed array: 5 4 3 2 1

## Program 5: Count Even and Odd Numbers

```
#include <iostream>
using namespace std;
int main() {
    int arr[10], even = 0, odd = 0;
    cout << "Enter 10 numbers: ";
    for (int i = 0; i < 10; i++) {
        cin >> arr[i];
        if (arr[i] % 2 == 0)
            even++;
    }
}
```

```
        else
            odd++;
    }
    cout << "Even numbers: " << even << "\nOdd
numbers: " << odd;
    return 0;
}
```

Output:

```
Enter 10 numbers: 1 2 3 4 5 6 7 8 9 10
Even numbers: 5
Odd numbers: 5
```

## Program 6: Search Element in 1D Array

```
#include <iostream>
using namespace std;
int main() {
    int arr[5], num, found = 0;
    cout << "Enter 5 elements: ";
    for (int i = 0; i < 5; i++) cin >> arr[i];
    cout << "Enter number to search: ";
    cin >> num;
    for (int i = 0; i < 5; i++) {
        if (arr[i] == num) {
            found = 1;
            break;
    }
```

```
    }
    if (found)
        cout << "Number found!";
    else
        cout << "Number not found!";
    return 0;
}
```

Output:

```
Enter 5 elements: 4 7 2 8 1
```

```
Enter number to search: 8
```

```
Number found!
```

## Program 7: Input and Display 2D Array(Matrix)

```
#include <iostream>
using namespace std;
int main() {
    int arr[2][2];
    cout << "Enter 4 elements: ";
    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            cin >> arr[i][j];
    cout << "Matrix is:\n";
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++)
```

```
        cout << arr[i][j] << " ";
        cout << endl;
    }
    return 0;
}
```

Output:

Enter 4 elements: 1 2 3 4

Matrix is:

```
1 2
3 4
```

## Program 8: Add two Matrices

```
#include <iostream>
using namespace std;
int main() {
    int a[2][2], b[2][2], c[2][2];
    cout << "Enter first matrix: ";
    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            cin >> a[i][j];
    cout << "Enter second matrix: ";
    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2; j++)
            cin >> b[i][j];
    cout << "Sum matrix:\n";
    for (int i = 0; i < 2; i++) {
```

```
    for (int j = 0; j < 2; j++) {  
        c[i][j] = a[i][j] + b[i][j];  
        cout << c[i][j] << " ";  
    }  
    cout << endl;  
}  
return 0;  
}
```

Output:

```
Enter first matrix: 1 2 3 4  
Enter second matrix: 5 6 7 8  
Sum matrix:  
6 8  
10 12
```

## Program 9: Subtract two Matrices

```
#include <iostream>  
using namespace std;  
int main() {  
    int a[2][2], b[2][2], c[2][2];  
    cout << "Enter first matrix: ";  
    for (int i = 0; i < 2; i++)  
        for (int j = 0; j < 2; j++)  
            cin >> a[i][j];  
    cout << "Enter second matrix: ";  
    for (int i = 0; i < 2; i++)
```

```

        for (int j = 0; j < 2; j++)
            cin >> b[i][j];
        cout << "Difference matrix:\n";
        for (int i = 0; i < 2; i++) {
            for (int j = 0; j < 2; j++) {
                c[i][j] = a[i][j] - b[i][j];
                cout << c[i][j] << " ";
            }
            cout << endl;
        }
        return 0;
    }
}

```

Output:

```

Enter first matrix: 5 6 7 8
Enter second matrix: 1 2 3 4
Difference matrix:
4 4
4 4

```

## Program 10: Multiply Two Matrices

```

#include <iostream>
using namespace std;
int main() {
    int a[2][2], b[2][2], c[2][2];
    cout << "Enter first matrix: ";
    for (int i = 0; i < 2; i++)

```

```
for (int j = 0; j < 2; j++)
    cin >> a[i][j];
cout << "Enter second matrix: ";
for (int i = 0; i < 2; i++)
    for (int j = 0; j < 2; j++)
        cin >> b[i][j];
cout << "Product matrix:\n";
for (int i = 0; i < 2; i++) {
    for (int j = 0; j < 2; j++) {
        c[i][j] = 0;
        for (int k = 0; k < 2; k++)
            c[i][j] += a[i][k] * b[k][j];
        cout << c[i][j] << " ";
    }
    cout << endl;
}
return 0;
}
```

Output:

Enter first matrix: 1 2 3 4

Enter second matrix: 5 6 7 8

Product matrix:

19 22

43 50

## Program 11: Bubble Sort in Ascending Order

```
#include <iostream>
using namespace std;
int main() {
    int arr[5];
    cout << "Enter 5 elements: ";
    for (int i = 0; i < 5; i++) cin >> arr[i];
    for (int i = 0; i < 4; i++)
        for (int j = 0; j < 4 - i; j++)
            if (arr[j] > arr[j + 1])
                swap(arr[j], arr[j + 1]);
    cout << "Sorted array (Ascending): ";
    for (int i = 0; i < 5; i++) cout << arr[i] << " ";
    return 0;
}
```

Output:

```
Enter 5 elements: 5 3 1 4 2
Sorted array (Ascending): 1 2 3 4 5
```

Program 12: Bubble Sort in descending order

```
#include <iostream>
using namespace std;
int main() {
    int arr[5];
    cout << "Enter 5 elements: ";
    for (int i = 0; i < 5; i++) cin >> arr[i];
```

```

    for (int i = 0; i < 4; i++)
        for (int j = 0; j < 4 - i; j++)
            if (arr[j] < arr[j + 1])
                swap(arr[j], arr[j + 1]);
    cout << "Sorted array (Descending): ";
    for (int i = 0; i < 5; i++) cout << arr[i] << " ";
    return 0;
}

```

Output:

```

Enter 5 elements: 5 3 1 4 2
Sorted array (Descending): 5 4 3 2 1

```

## Program 13: Sort Names of students(Bubble Sort)

```

#include <iostream>
#include <string>
using namespace std;
int main() {
    string names[3];
    cout << "Enter 3 names: ";
    for (int i = 0; i < 3; i++) cin >> names[i];
    for (int i = 0; i < 2; i++)
        for (int j = 0; j < 2 - i; j++)
            if (names[j] > names[j + 1])
                swap(names[j], names[j + 1]);
}

```

```
cout << "Sorted names: ";
for (int i = 0; i < 3; i++) cout << names[i] << " ";
return 0;
}
```

Output:

```
Enter 3 names: Ravi Amit Mohit
Sorted names: Amit Mohit Ravi
```

## Program 14: Linear Search in 1D Array

```
#include <iostream>
using namespace std;
int main() {
    int arr[5], num, flag = 0;
    cout << "Enter 5 elements: ";
    for (int i = 0; i < 5; i++) cin >> arr[i];
    cout << "Enter element to search: ";
    cin >> num;
    for (int i = 0; i < 5; i++)
        if (arr[i] == num)
            flag = 1;
    if (flag)
        cout << "Element found!";
    else
        cout << "Element not found!";
    return 0;
}
```

Output:

Enter 5 elements: 1 3 5 7 9

Enter element to search: 5

Element found!

## Program 15: Binary Search in 1D Array

```
#include <iostream>
using namespace std;
int main() {
    int arr[5] = {1, 3, 5, 7, 9}, num, low = 0, high = 4, mid;
    cout << "Enter element to search: ";
    cin >> num;
    while (low <= high) {
        mid = (low + high) / 2;
        if (arr[mid] == num) {
            cout << "Element found at index " << mid;
            return 0;
        } else if (arr[mid] < num)
            low = mid + 1;
        else
            high = mid - 1;
    }
    cout << "Element not found!";
    return 0;
}
```

Output:

Enter element to search: 7

Element found at index 3

## Program 16: Count Occurrences of an element(Linear Search)

```
#include <iostream>
using namespace std;
int main() {
    int arr[6], num, count = 0;
    cout << "Enter 6 elements: ";
    for (int i = 0; i < 6; i++) cin >> arr[i];
    cout << "Enter element to count: ";
    cin >> num;
    for (int i = 0; i < 6; i++)
        if (arr[i] == num)
            count++;
    cout << "Element " << num << " occurred " << count
    << " times.";
    return 0;
}
```

Output:

Enter 6 elements: 1 2 3 2 4 2

Enter element to count: 2

Element 2 occurred 3 times.

## Program 17: Check if element is present(Linear Search)

```
#include <iostream>
using namespace std;
int main() {
    int arr[5], num;
    bool found = false;
    cout << "Enter 5 elements: ";
    for (int i = 0; i < 5; i++) cin >> arr[i];
    cout << "Enter number to check: ";
    cin >> num;
    for (int i = 0; i < 5; i++)
        if (arr[i] == num)
            found = true;
    if (found)
        cout << "Number is present.";
    else
        cout << "Number is not present.";
    return 0;
}
```

Output:

Enter 5 elements: 10 20 30 40 50

Enter number to check: 30

Number is present.

## Program 18: Binary Search(Integer Array)

```
#include <iostream>
using namespace std;
int main() {
    int arr[6] = {2, 4, 6, 8, 10, 12};
    int num, low = 0, high = 5, mid;
    cout << "Enter number to search: ";
    cin >> num;
    while (low <= high) {
        mid = (low + high) / 2;
        if (arr[mid] == num) {
            cout << "Number found at position " << mid + 1;
            return 0;
        } else if (arr[mid] < num)
            low = mid + 1;
        else
            high = mid - 1;
    }
    cout << "Number not found.";
    return 0;
}
```

Output:

```
Enter number to search: 10
Number found at position 5
```

## Program 19: Binary Search(Floating Numbers)

```
#include <iostream>
using namespace std;
int main() {
    float arr[5] = {1.2, 2.4, 3.6, 4.8, 6.0};
    float num;
    int low = 0, high = 4, mid;
    cout << "Enter number to search: ";
    cin >> num;
    while (low <= high) {
        mid = (low + high) / 2;
        if (arr[mid] == num) {
            cout << "Number found at position " << mid + 1;
            return 0;
        } else if (arr[mid] < num)
            low = mid + 1;
        else
            high = mid - 1;
    }
    cout << "Number not found.";
    return 0;
}
```

Output:

Enter number to search: 4.8

Number found at position 4

## Program 20: Search Largest Element in Sorted Array(Binary Search Logic)

```
#include <iostream>
using namespace std;
int main() {
    int arr[6] = {3, 7, 12, 18, 25, 31};
    int low = 0, high = 5;
    cout << "Largest element in sorted array is: " <<
arr[high];
    return 0;
}
```

Output:

Largest element in sorted array is: 31