ASSIGNMENT NO. = 1 (Array)

1) Write a program to implement array data structure with its operations.

→

```
#include<iostream>
#define  max  10
int  item[max], cnt = 0;
using namespace std;

void   insert(int ele){
   if (cnt == max)
      cout << "\nArray is Full.." << endl;
   else
      item[cnt++] = ele, cout << "Element is inserted.." << endl;
}

int    remove(int pos) {
   if (pos < 0 || pos >= cnt) return cout << "\nWrong Position..\n", 0;
   int ele = item[pos];
   for (int i = pos; i < cnt - 1; i++) item[i] = item[i + 1];
   cnt--;
   return ele;
}

void   display() {
   cout << "\nElements:\t";
```

```
    if (!cnt) cout << "Empty..\n";
    else for (int i = 0; i < cnt; i++) cout << item[i] << "\t";
    cout << endl;
}

void   reverse() {
    cout << "\nReversed elements:\t";
    if (!cnt) cout << "Empty..\n";
    else for (int i = cnt - 1; i >= 0; i--) cout << item[i] << "\t";
    cout << endl;
}

int  main() {
     int  ch, ele, pos;
     do {
        cout << "\n1: Insert\n2: Remove\n3: Display\n4: Reverse\n5: Exit\nEnter your
choice: ";
        cin >> ch;
        switch (ch)
        {
           case 1:
              cout << "\nEnter element: "; cin >> ele;
              insert(ele);
              break;
           case 2:
              cout << "\nEnter position to remove element: "; cin >> pos;
              cout << "\nRemoved element: " <<  remove(pos) << endl;
              break;
```

```
            case 3:
                display();
                break;
            case 4:
                reverse();
                break;
            case 5:
                cout << "\nExited.." << endl;
            default:
                cout << "\nWrong input.." << endl;
                break;
        }
    }while (ch != 5);
    return 0;
}
```

Output =>

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 1

Enter element: 10

Element is inserted..

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 1

Enter element: 20

Element is inserted..

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 1

Enter element: 30

Element is inserted..

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 1

Enter element: 40

Element is inserted..

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 3

Elements: 10 20 30 40

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 4

Reversed elemnts: 40 30 20 10

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 2

Enter position to remove element: 0

Removed element: 10

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 2

Enter position to remove element: 3

Wrong Position..

Removed element: 0

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 3

Elements: 20 30 40

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 2

Enter position to remove element: 2

Removed element: 40

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice:

3

Elements: 20 30

1: Insert

2: Remove

3: Display

4: Reverse

5: Exit

Enter your choice: 5

Exited..

Wrong input..

2) Write a program that print only even numbers in an array.

->

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

int main() {
    int n;
    cout << "Enter the size of the array: ";
    cin >> n;

    int *arr = new int[n];

    cout << "Enter " << n << " elements: ";
    for (int i = 0; i < n; i++) cin >> arr[i];

    cout << "Even numbers in the array: ";
    for (int i = 0; i < n; i++)
        if (arr[i] % 2 == 0) cout << arr[i] << " ";

    return 0;
}
```


Output =>

Enter the size of the array: 5

Enter 5 elements: 10 20 3 4 2

Even numbers in the array: 10 20 4 2

3) Write a program that print only odd numbers in an array.

->

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

int main() {
    int n;
    cout << "Enter the size of the array: ";
    cin >> n;

    int *arr = new int[n];

    cout << "Enter " << n << " elements: ";
    for (int i = 0; i < n; i++) cin >> arr[i];

    cout << "Odd numbers in the array: ";
    for (int i = 0; i < n; i++)
        if (arr[i] % 2 != 0) cout << arr[i] << " ";

    return 0;
}
```


Output =>

Enter the size of the array: 5

Enter 5 elements: 10 20 3 4 2

Odd numbers in the array: 3

4) Write a program that print maximum & minimum number in an array.

->

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

int main() {
    int n;
    cout << "Enter the size of the array: ";
    cin >> n;

    int *arr = new int[n];

    cout << "Enter " << n << " elements: ";
    for (int i = 0; i < n; i++) cin >> arr[i];

    int max = arr[0], min = arr[0];

    for (int i = 0; i < n; i++) {
        if (arr[i] > max) max = arr[i];
        if (arr[i] < min) min = arr[i];
    }

    cout << "Maximum number: " << max << "\nMinimum number: " << min << endl;

    return 0;
```

```
}
```

Output =>

Enter the size of the array: 5

Enter 5 elements: 10 2 200 199 1

Maximum number: 200

Minimum number: 1

5) Write a program to find addition of two matrices.

->

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

int main() {
    int rows, cols;
    cout << "Enter the number of rows and columns: ";
    cin >> rows >> cols;

    int mat1[rows][cols], mat2[rows][cols], result[rows][cols];

    cout << "Enter elements of 1st matrix:\n";
    for (int i = 0; i < rows; i++)
        for (int j = 0; j < cols; j++)
            cin >> mat1[i][j];

    cout << "Enter elements of 2nd matrix:\n";
```

```
    for (int i = 0; i < rows; i++)
        for (int j = 0; j < cols; j++)
            cin >> mat2[i][j];

    cout << "Result:\n";
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            result[i][j] = mat1[i][j] + mat2[i][j];
            cout << result[i][j] << "\t";
        }
        cout << endl;
    }

    return 0;
}
```

Output =>

```
Enter the number of rows and columns: 2 2
Enter elements of 1st matrix:
10 20
30 40
Enter elements of 2nd matrix:
1 2
3 4
Result:
11      22
33      44
```

6) Write a program to find subtraction of two matrices.

->

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

int main() {
    int rows, cols;
    cout << "Enter the number of rows and columns: ";
    cin >> rows >> cols;

    int mat1[rows][cols], mat2[rows][cols], result[rows][cols];

    cout << "Enter elements of 1st matrix:\n";
    for (int i = 0; i < rows; i++)
        for (int j = 0; j < cols; j++)
            cin >> mat1[i][j];

    cout << "Enter elements of 2nd matrix:\n";
    for (int i = 0; i < rows; i++)
        for (int j = 0; j < cols; j++)
            cin >> mat2[i][j];

    cout << "Result:\n";
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
```

```
            result[i][j] = mat1[i][j] - mat2[i][j];
            cout << result[i][j] << "\t";
        }
        cout << endl;
    }

    return 0;
}
```

Output =>

Enter the number of rows and columns: 2 2

Enter elements of 1st matrix:

10 20

30 40

Enter elements of 2nd matrix:

1 2

3 4

Result:

9 18

27 36

7) Write a program to find multiplication of two matrices.

->

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

```
int main() {
    int rows1, cols1, rows2, cols2;

    cout << "Enter the number of rows and columns of 1st matrix: ";
    cin >> rows1 >> cols1;
    cout << "Enter the number of rows and columns of 2nd matrix: ";
    cin >> rows2 >> cols2;

    if (cols1 != rows2) {
        cout << "Matrix multiplication not possible" << endl;
        return 0;
    }

    int mat1[rows1][cols1], mat2[rows2][cols2], result[rows1][cols2];

    cout << "Enter elements of 1st matrix:\n";
    for (int i = 0; i < rows1; i++)
        for (int j = 0; j < cols1; j++)
            cin >> mat1[i][j];

    cout << "Enter elements of 2nd matrix:\n";
    for (int i = 0; i < rows2; i++)
        for (int j = 0; j < cols2; j++)
            cin >> mat2[i][j];

    for (int i = 0; i < rows1; i++)
```

```
        for (int j = 0; j < cols2; j++)
            result[i][j] = 0;

    for (int i = 0; i < rows1; i++)
        for (int j = 0; j < cols2; j++)
            for (int k = 0; k < cols1; k++)
                result[i][j] += mat1[i][k] * mat2[k][j];

    cout << "Result:\n";
    for (int i = 0; i < rows1; i++) {
        for (int j = 0; j < cols2; j++) {
            cout << result[i][j] << "\t";
        }
        cout << endl;
    }

    return 0;
}
```


Output =>

Enter the number of rows and columns of 1st matrix: 2 2

Enter the number of rows and columns of 2nd matrix: 2 2

Enter elements of 1st matrix:

1 2 3 4

Enter elements of 2nd matrix:

5 6 7 8

Result:

19 22

43 50