AQ1

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

int main() {
    int arr[100];
    int n = 0;
    int choice, i, pos, x, found;

    while (true) {
        cout << "\n----- MENU -----\n";
        cout << "1. CREATE\n";
        cout << "2. DISPLAY\n";
        cout << "3. INSERT\n";
        cout << "4. DELETE\n";
        cout << "5. LINEAR SEARCH\n";
        cout << "6. EXIT\n";
        cout << "Enter your choice: ";
        cin >> choice;

        if (choice == 1)
        {
            cout << "Enter number of elements: ";
            cin >> n;
            cout << "Enter " << n << " elements:\n";
            for (i = 0; i < n; i++) {
                cin >> arr[i];
            }
        }
        else if (choice == 2)
        {
            if (n == 0)
                cout << "Array is empty.\n";
            else
            {
                cout << "Array elements: ";
                for (i = 0; i < n; i++)
                    cout << arr[i] << " ";
                cout << endl;
            }
        }
        else if (choice == 3)
        {
            cout << "Enter position (1 to " << n+1 << "): ";
            cin >> pos;
            cout << "Enter element: ";
```

```
            cin >> x;
            if (pos < 1 || pos > n+1)
            {
                cout << "Invalid position!\n";
            }
            else
            {
                for (i = n; i >= pos; i--) {
                    arr[i] = arr[i - 1];
                }
                arr[pos - 1] = x;
                n++;
                cout << "Element inserted.\n";
            }
        }
        else if (choice == 4)
        {
            cout << "Enter position (1 to " << n << "): ";
            cin >> pos;
            if (pos < 1 || pos > n)
            {
                cout << "Invalid position!\n";
            }
            else
            {
                x = arr[pos - 1];
                for (i = pos - 1; i < n - 1; i++) {
                    arr[i] = arr[i + 1];
                }
                n--;
                cout << "Deleted element: " << x << endl;
            }
        }
        else if (choice == 5)
        {
            cout << "Enter element to search: ";
            cin >> x;
            found = -1;
            for (i = 0; i < n; i++)
            {
                if (arr[i] == x)
                {
                    found = i;
                    break;
                }
            }
            if (found == -1)
                cout << "Element not found.\n";
```

```
            else
                cout << "Element found at position " << found + 1 << endl;
        }
        else if (choice == 6)
        {
            cout << "Exiting program.\n";
            break;
        }
        else
        {
            cout << "Invalid choice! Try again.\n";
        }
    }
    return 0;
}
```

Output1:

```
----- MENU -----
1. CREATE
2. DISPLAY
3. INSERT
4. DELETE
5. LINEAR SEARCH
6. EXIT
Enter your choice: 1
Enter number of elements: 4
Enter 4 elements:
1
2
3
4

----- MENU -----
1. CREATE
2. DISPLAY
3. INSERT
4. DELETE
5. LINEAR SEARCH
6. EXIT
Enter your choice: 3
Enter position (1 to 5): 2
Enter element:
66
Element inserted.

----- MENU -----
1. CREATE
2. DISPLAY
3. INSERT
4. DELETE
5. LINEAR SEARCH
6. EXIT
Enter your choice: 2
Array elements: 1 66 2 3 4

----- MENU -----
1. CREATE
2. DISPLAY
3. INSERT
4. DELETE
5. LINEAR SEARCH
6. EXIT
Enter your choice: 6
Exiting program.
PS F:\Work\SEM3\DSA\LAB\1>
```

AQ2

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

int main()
{
    int n;
    cout << "Enter number of elements: ";
    cin >> n;
    int arr[n];
    cout << "Enter " << n << " elements:\n";
    for (int i = 0; i < n; i++)
    {
        cin >> arr[i];
    }
    cout << "Original Array:\n";
    for (int i = 0; i < n; i++)
    {
        cout << arr[i] << " ";
    }
    for (int i = 0; i < n; i++)
    {
        for (int j = i + 1; j < n; j++)
        {
            if (arr[i] == arr[j])
            {
                for (int k = j; k < n - 1; k++)
                {
                    arr[k] = arr[k + 1];
                }
                n--;
                j--;
            }
        }
    }
    cout << "\nArray after removing duplicates:\n";
    for (int i = 0; i < n; i++)
    {
        cout << arr[i] << " ";
    }
    cout << endl;
    return 0;
}
```

Output2:

```
Enter number of elements: 7
Enter 7 elements:
1
2
1
3
4
5
3
Original Array:
1 2 1 3 4 5 3
Array after removing duplicates:
1 2 3 4 5
PS F:\Work\SEM3\DSA\LAB\1>
```

AQ3

```c
#include <stdio.h>
int main()
{
    int i;
    int arr[5] = {1};
    for (i = 0; i < 5; i++)
        printf("%d", arr[i]);
    return 0;
}
// OUTPUT: 10000
```

Output3:

```
> cd "f:\Work\SEM3\DSA\LAB\1\" ; if ($?) { g++ AQ3.cpp -o AQ3 } ; if ($?) { .\AQ3 }
10000
PS F:\Work\SEM3\DSA\LAB\1>
```

AQ4

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

int main()
{
    cout << "a) Reverse the elements of an array"<<endl;
    int n;
    cout << "Enter size of array: ";
    cin >> n;
    int arr[n];
    cout << "Enter " << n << " elements:\n";
    for (int i = 0; i < n; i++)
    {
        cin >> arr[i];
    }

    cout << "Original Array:\n";
    for (int i = 0; i < n; i++)
    {
        cout << arr[i] << " ";
    }
    cout << "\n";

    for (int i = 0; i < n / 2; i++)
    {
        int temp = arr[i];
        arr[i] = arr[n - 1 - i];
        arr[n - 1 - i] = temp;
    }

    cout << "Array after reversing:\n";
    for (int i = 0; i < n; i++)
    {
        cout << arr[i] << " ";
    }
    cout << "\n\n";

    cout << "b) Find the matrix multiplication"<<endl;
    int r1, c1, r2, c2;
    cout << "Enter rows and cols of first matrix: ";
    cin >> r1 >> c1;
    cout << "Enter rows and cols of second matrix: ";
    cin >> r2 >> c2;

    if (c1 != r2)
```

```
    {
        cout << "Matrix multiplication not possible!\n\n";
    }
    else
    {
        int A[r1][c1], B[r2][c2], C[r1][c2];
        cout << "Enter elements of first matrix:\n";
        for (int i = 0; i < r1; i++)
        {
            for (int j = 0; j < c1; j++)
            {
                cin >> A[i][j];
            }
        }
        cout << "First Matrix:\n";
        for (int i = 0; i < r1; i++)
        {
            for (int j = 0; j < c1; j++)
            {
                cout << A[i][j] << " ";
            }
            cout << endl;
        }

        cout << "Enter elements of second matrix:\n";
        for (int i = 0; i < r2; i++)
        {
            for (int j = 0; j < c2; j++)
            {
                cin >> B[i][j];
            }
        }
        cout << "Second Matrix:\n";
        for (int i = 0; i < r2; i++)
        {
            for (int j = 0; j < c2; j++)
            {
                cout << B[i][j] << " ";
            }
            cout << endl;
        }

        for (int i = 0; i < r1; i++)
        {
            for (int j = 0; j < c2; j++)
            {
                C[i][j] = 0;
            }
```

```
        }
        for (int i = 0; i < r1; i++)
        {
            for (int j = 0; j < c2; j++)
            {
                for (int k = 0; k < c1; k++)
                {
                    C[i][j] += A[i][k] * B[k][j];
                }
            }
        }
        cout << "Result of multiplication:\n";
        for (int i = 0; i < r1; i++)
        {
            for (int j = 0; j < c2; j++)
            {
                cout << C[i][j] << " ";
            }
            cout << endl;
        }
        cout << "\n";
    }

    cout << "c) Find the Transpose of a Matrix"<<endl;
    int r, c;
    cout << "Enter rows and cols of matrix: ";
    cin >> r >> c;
    int M[r][c];
    cout << "Enter elements of matrix:\n";
    for (int i = 0; i < r; i++)
    {
        for (int j = 0; j < c; j++)
        {
            cin >> M[i][j];
        }
    }

    cout << "Original Matrix:\n";
    for (int i = 0; i < r; i++)
    {
        for (int j = 0; j < c; j++)
        {
            cout << M[i][j] << " ";
        }
        cout << endl;
    }

    if (r == c)
```

```
    {
        for (int i = 0; i < r; i++)
        {
            for (int j = i + 1; j < c; j++)
            {
                int temp = M[i][j];
                M[i][j] = M[j][i];
                M[j][i] = temp;
            }
        }
        cout << "Transpose of matrix:\n";
        for (int i = 0; i < r; i++)
        {
            for (int j = 0; j < c; j++)
            {
                cout << M[i][j] << " ";
            }
            cout << endl;
        }
    }
    else
    {
        int T[c][r];
        for (int i = 0; i < r; i++)
        {
            for (int j = 0; j < c; j++)
            {
                T[j][i] = M[i][j];
            }
        }
        cout << "Transpose of matrix:\n";
        for (int i = 0; i < c; i++)
        {
            for (int j = 0; j < r; j++)
            {
                cout << T[i][j] << " ";
            }
            cout << endl;
        }
    }
    return 0;
}
```

Output4:

```
                        > cd "f:\Work\SEM3\DSA\LAB\1\" ; if ($?) { g++ AQ4.cpp -
o AQ4 } ; if ($?) { .\AQ4 }
a) Reverse the elements of an array
Enter size of array: 4
Enter 4 elements:
1
2
3
4
Original Array:
1 2 3 4
Array after reversing:
4 3 2 1

b) Find the matrix multiplication
Enter rows and cols of first matrix: 3
2
Enter rows and cols of second matrix: 2
3
Enter elements of first matrix:
1
2
1
2
3
3
First Matrix:
1 2
1 2
3 3
Enter elements of second matrix:
1
2
3
1
2
3
Second Matrix:
1 2 3
1 2 3
Result of multiplication:
3 6 9
3 6 9
6 12 18

c) Find the Transpose of a Matrix
Enter rows and cols of matrix: 3
2
Enter elements of matrix:
1
2
2
3
4
5
Original Matrix:
1 2
2 3
4 5
Transpose of matrix:
1 2 4
2 3 5
PS F:\Work\SEM3\DSA\LAB\1>
```

AQ5

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

int main() {
    int r, c;
    cout << "Enter number of rows and columns: ";
    cin >> r >> c;
    int arr[r][c];
    cout << "Enter elements of the matrix:\n";
    for (int i = 0; i < r; i++) {
        for (int j = 0; j < c; j++) {
            cin >> arr[i][j];
        }
    }
    cout << "\nMatrix:\n";
    for (int i = 0; i < r; i++) {
        for (int j = 0; j < c; j++) {
            cout << arr[i][j] << " ";
        }
        cout << endl;
    }
    cout << "\nSum of each row:\n";
    for (int i = 0; i < r; i++) {
        int rowSum = 0;
        for (int j = 0; j < c; j++) {
            rowSum += arr[i][j];
        }
        cout << "Row " << i + 1 << ": " << rowSum << endl;
    }
    cout << "\nSum of each column:\n";
    for (int j = 0; j < c; j++) {
        int colSum = 0;
        for (int i = 0; i < r; i++) {
            colSum += arr[i][j];
        }
        cout << "Column " << j + 1 << ": " << colSum << endl;
    }
    return 0;
}
```

Output:

```
Enter number of rows and columns: 3
4
Enter elements of the matrix:
1
2
3
4
1
2
3
4
12
2
3
4

Matrix:
1 2 3 4
1 2 3 4
12 2 3 4

Sum of each row:
Row 1: 10
Row 2: 10
Row 3: 21

Sum of each column:
Column 1: 14
Column 2: 6
Column 3: 9
Column 4: 12
PS F:\Work\SEM3\DSA\LAB\1>
```