```cpp
//Q1)
#include<iostream>
using namespace std;

const int MAX = 100;
int arr[MAX], size = 0;


void create() {
    cout << "Enter number of elements: ";
    cin >> size;
    cout << "Enter elements:\n";
    for(int i = 0; i < size; i++) {
        cin >> arr[i];
    }
}


void display() {
    if(size == 0) {
        cout << "Array is empty.\n";
        return;
    }
    cout << "Array elements: ";
    for(int i = 0; i < size; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
}


void insert() {
    if(size >= MAX) {
        cout << "Array is full. Cannot insert.\n";
        return;
    }
    int pos, val;
    cout << "Enter position (0 to " << size << "): ";
    cin >> pos;
    if(pos < 0 || pos > size) {
```

```
        cout << "Invalid position.\n";
        return;
    }
    cout << "Enter value to insert: ";
    cin >> val;
    for(int i = size; i > pos; i--) {
        arr[i] = arr[i - 1];
    }
    arr[pos] = val;
    size++;
    cout << "Element inserted.\n";
}


void deleteElement() {
    if(size == 0) {
        cout << "Array is empty. Nothing to delete.\n";
        return;
    }
    int pos;
    cout << "Enter position to delete (0 to " << size-1 << "): ";
    cin >> pos;
    if(pos < 0 || pos >= size) {
        cout << "Invalid position.\n";
        return;
    }
    for(int i = pos; i < size - 1; i++) {
        arr[i] = arr[i + 1];
    }
    size--;
    cout << "Element deleted.\n";
}


void linearSearch() {
    if(size == 0) {
        cout << "Array is empty.\n";
        return;
    }
    int key, found = 0;
```

```
    cout << "Enter element to search: ";
    cin >> key;
    for(int i = 0; i < size; i++) {
        if(arr[i] == key) {
            cout << "Element found at position " << i << endl;
            found = 1;
            break;
        }
    }
    if(!found)
        cout << "Element not found.\n";
}


int main() {
    int choice;
    do {
        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;

        switch(choice) {
            case 1: create(); break;
            case 2: display(); break;
            case 3: insert(); break;
            case 4: deleteElement(); break;
            case 5: linearSearch(); break;
            case 6: cout << "Exiting program.\n"; break;
            default: cout << "Invalid choice. Try again.\n";
        }

    } while(choice != 6);

    return 0;
```

```
}
```

Q2)

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

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

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


    int unique[n];
    int uniqueCount = 0;

    for(int i=0; i<n; i++) {
        bool isDuplicate = false;

        for(int j=0; j<uniqueCount; j++) {
            if(arr[i] == unique[j]) {
                isDuplicate = true;
                break;
            }
        }

        if(!isDuplicate) {
            unique[uniqueCount] = arr[i];
            uniqueCount++;
        }
    }

    cout << "Array after removing duplicates:\n";
    for(int i=0; i<uniqueCount; i++) {
        cout << unique[i] << " ";
```

```
    }
    cout << endl;

    return 0;
}
```

Q4)
REVERSE A LOOP

```
#include<iostream>
using namespace std;
int main(){
    int arr[5];
    int temp;
    for(int i=0;i<5;i++){
       cout<<"ENter element "<<i+1<<"here :";
       cin>>arr[i];
    }
    for(int i=0;i<5/2;i++){
        temp = arr[i];
        arr[i] = arr[4-i];
        arr[4-i] = temp;
    }
    for(int i=0;i<5;i++){
        cout<<arr[i]<<endl;
    }
    return 0;
}
```

B) MATRIX MULTIPLICATION

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

int main() {
    int r1, c1, r2, c2;
    cout << "Enter rows and columns of first matrix: ";
    cin >> r1 >> c1;

    cout << "Enter rows and columns of second matrix: ";
    cin >> r2 >> c2;

    if (c1 != r2) {
        cout << "Matrix multiplication not possible." << endl;
        return 0;
    }

    int A[r1][c1], B[r2][c2], result[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 << "Enter elements of second matrix:\n";
    for (int i = 0; i < r2; i++) {
        for (int j = 0; j < c2; j++) {
            cin >> B[i][j];
        }
    }

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

    for (int i = 0; i < r1; i++) {
```

```
        for (int j = 0; j < c2; j++) {
            for (int k = 0; k < c1; k++) {
                result[i][j] += A[i][k] * B[k][j];
            }
        }
    }

    cout << "\nResultant Matrix:\n";
    for (int i = 0; i < r1; i++) {
        for (int j = 0; j < c2; j++) {
            cout << result[i][j] << " ";
        }
        cout << endl;
    }

    return 0;
}
```

C)TRANSPOSE OF A MATRIX

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

int main() {
    int r, c;
    cout << "Enter rows: ";
    cin >> r;
    cout << "Enter columns: ";
    cin >> c;

    int arr[r][c];
    cout << "Enter elements:\n";
    for (int i = 0; i < r; i++) {
        for (int j = 0; j < c; j++) {
            cin >> arr[i][j];
        }
    }


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


    int brr[c][r];
    for (int i = 0; i < c; i++) {
        for (int j = 0; j < r; j++) {
            brr[i][j] = arr[j][i];
        }
    }


    cout << "Transposed Matrix: "<<endl;
    for (int i = 0; i < c; i++) {
        for (int j = 0; j < r; j++) {
```

```
            cout << brr[i][j] << " "; // Added space for clarity
        }
        cout << endl;
    }

    return 0;
}
```

Q5)

```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 matrix elements:\n";
    for (int i = 0; i < r; i++) {
        for (int j = 0; j < c; j++) {
            cin >> arr[i][j];
        }
    }

    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;
}
```