UCS301 Data Structures

Lab Assignment 1

Suneet Singh Arora 1024030174

Q1)

```cpp
#include <iostream> //Including header file
using namespace std;

//declaring variables and array
const int MAX = 50;
int arr[MAX];
int size = 0;

//Create function
void create() {
    cout << "Enter size of array: ";
    cin >> size;
    cout << "Enter " << size << " elements: ";
    for(int i = 0; i < size; i++) cin >> arr[i];
}

//Display Function
void display() {
    cout << "Array: ";
    for(int i = 0; i < size; i++) cout << arr[i] << " ";
    cout << endl;
}

```

```cpp
//Insert Function
void insert() {
    int pos, val;
    cout << "Enter position and value to insert: ";
    cin >> pos >> val;
    for(int i = size; i > pos; i--) arr[i] = arr[i-1];
    arr[pos] = val;
    size++;
}

//Delete Function
void del() {
    int pos;
    cout << "Enter position to delete: ";
    cin >> pos;
    for(int i = pos; i < size - 1; i++) arr[i] = arr[i + 1];
    size--;
}

```

```
43  //Search Function
44  void search() {
45      int val, found = 0;
46      cout << "Enter value to search: ";
47      cin >> val;
48      for(int i = 0; i < size; i++) {
49          if(arr[i] == val) {
50              cout << "Found at position " << i << endl;
51              found = 1;
52          }
53      }
54      if(!found) cout << "Not found\n";
55  }
56
57  //Starting main function
58  int main() {
59      int choice;
60      //Starting menu using while loop
61      while(true) {
62          cout << "\n1. Create\n2. Display\n3. Insert\n4. Delete\n5. Search\n6.
                Exit\n";
63          cout << "Enter choice: ";
64          cin >> choice;
```

```
65
66          switch(choice) {
67              case 1: create(); break;
68              case 2: display(); break;
69              case 3: insert(); break;
70              case 4: del(); break;
71              case 5: search(); break;
72              case 6: return 0;
73              default: cout << "Invalid choice\n";
74          }
75      }
76  }
```

Q2)

```cpp
#include <bits/stdc++.h>//including headerfiles
using namespace std;

//Starting main function
int main() {
    //Getting inputs from user for size of array
    int n;
    cout << "Enter size of array: ";
    cin >> n;

    //Getting inputs from user for elements present in array
    int arr[100];
    cout << "Enter elements of array: ";
    for (int i = 0; i < n; i++) {
        cin >> arr[i];
    }

    //Detecting duplicate array elements
    for (int i = 0; i < n; i++) {
        for (int j = i + 1; j < n; ) {
            if (arr[i] == arr[j]) {
                for (int k = j; k < n - 1; k++) {
                    arr[k] = arr[k + 1];
                }
                n--; // decreasing the size of array
            } else {
                j++; // incrementing when there is no deletion
            }
        }
    }
    //printing the resultant array
    cout << "Array after removing the duplicate elements: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }

    return 0;
}
```

Q3) Output : 10000

Q4) a)

```cpp
#include <bits/stdc++.h>//Incuding headerfiles
using namespace std;

//starting main function
int main() {
    //Getting input from user for size of array
    int n;
    cout << "Enter size of array: ";
    cin >> n;

    //Getting input from user for elements of array
    int arr[n];
    cout << "Enter elements for array: ";
    for(int i = 0; i < n; i++) {
        cin >> arr[i];
    }

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

    //Printing the output array
    for(int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }

    return 0;
}

```

b)

```cpp
#include <bits/stdc++.h>//Including header files
using namespace std;

int main() {
    //Getting user input for sizes of the 2 matrices
    int a, b, c, d;
    cout << "Enter row size for first matrix: ";
    cin >> a;
    cout << "Enter column size for first matrix: ";
    cin >> b;
    cout << "Enter row size for second matrix: ";
    cin >> c;
    cout << "Enter column size for second matrix: ";
    cin >> d;

    //Checking if matrix multiplication is possible
    if (b != c) {
        cout << "Matrix multiplication not possible.\n";
        return 0;
    }

    //Making arrays based on user given size
    int arr[a][b];
    int brr[c][d];

    //Getting input for first matrix
    cout << "Enter first matrix:\n";
    for (int i = 0; i < a; i++) {
        for (int j = 0; j < b; j++) {
            cin >> arr[i][j];
        }
    }

    //Getting input for second matrix
    cout << "Enter second matrix:\n";
    for (int i = 0; i < c; i++) {
        for (int j = 0; j < d; j++) {
            cin >> brr[i][j];
        }
    }

    //Making third matrix for multiplication product
    int crr[a][d];

    //Finding elements for the multiplication product
    for (int i = 0; i < a; i++) {
        for (int j = 0; j < d; j++) {
            crr[i][j] = 0;
            for (int k = 0; k < b; k++) {
                crr[i][j] += arr[i][k] * brr[k][j];
            }
        }
```

```
    }

    //Printing the output matrix
    cout << "Resultant matrix after multiplication is :\n";
    for (int i = 0; i < a; i++) {
        for (int j = 0; j < d; j++) {
            cout << crr[i][j] << " ";
        }
        cout << endl;
    }

    return 0;
}
```

c)

```
#include <bits/stdc++.h> //Including header
using namespace std;

int main() {
    //Getting user input for sizes of the 2 matrices
    int a, b, c, d;
    cout << "Enter number of rows for the first matrix: ";
    cin >> a;
    cout << "Enter number of columns for the first matrix: ";
    cin >> b;
    cout << "Enter number of rows for the second matrix: ";
    cin >> c;
    cout << "Enter number of columns for the second matrix: ";
    cin >> d;

    //Checking if matrix multiplication is possible
    if (b != c) {
        cout << "Error: Cannot multiply. Columns of the first matrix must
            equal rows of the second.\n";
        return 0;
    }

```

```
    //Making arrays based on user given size
    int arr[a][b];
    int brr[c][d];

    //Getting input for first matrix
    cout << "\nEnter elements for the first matrix:\n";
    for (int i = 0; i < a; i++) {
        for (int j = 0; j < b; j++) {
            cin >> arr[i][j];
        }
    }

    //Getting input for second matrix
    cout << "\nEnter elements for the second matrix:\n";
    for (int i = 0; i < c; i++) {
        for (int j = 0; j < d; j++) {
            cin >> brr[i][j];
        }
    }

    //Making third matrix for multiplication product
    int crr[a][d];
```

```

    //Finding elements for the multiplication product
    for (int i = 0; i < a; i++) {
        for (int j = 0; j < d; j++) {
            crr[i][j] = 0;
            for (int k = 0; k < b; k++) {
                crr[i][j] += arr[i][k] * brr[k][j];
            }
        }
    }

    //Printing the output matrix
    cout << "\nThe resulting matrix after multiplication is:\n";
    for (int i = 0; i < a; i++) {
        for (int j = 0; j < d; j++) {
            cout << crr[i][j] << " ";
        }
        cout << endl;
    }

    return 0;
}
```

Q5)

```cpp
#include <bits/stdc++.h> // Including all standard headers
using namespace std;

int main() {
    // Taking user input for matrix size
    int a, b;
    cout << "Enter number of rows: ";
    cin >> a;
    cout << "Enter number of columns: ";
    cin >> b;

    // Creating 2D array with given dimensions
    int arr[a][b];

    // Taking input values for the matrix
    cout << "\nEnter elements for the matrix:\n";
    for (int i = 0; i < a; i++) {
        for (int j = 0; j < b; j++) {
            cin >> arr[i][j];
        }
    }
```

```cpp

    // Calculating and printing sum of each row
    cout << "\nSum of each row:\n";
    for (int i = 0; i < a; i++) {
        int row_add = 0;
        for (int j = 0; j < b; j++) {
            row_add += arr[i][j];
        }
        cout << "Row " << i + 1 << " = " << row_add << endl;
    }

    // Calculating and printing sum of each column
    cout << "\nSum of each column:\n";
    for (int i = 0; i < b; i++) {
        int col_add = 0;
        for (int j = 0; j < a; j++) {
            col_add += arr[j][i];
        }
        cout << "Column " << i + 1 << " = " << col_add << endl;
    }

    return 0;
```