**Module 3**

#include <iostream>

using namespace std;

class LS **Topic 1**

{

private:

    int x;

public:

    void get\_data(int a)

    {

        x = a;

    }

    int lsearch(int a[], int s)

    {

        for (int i = 0; i < s; i++)

        {

            if (a[i] == x)

            {

                return i;

            }

        }

        return -1;

    }

};

int main()

{

    int n, x, y;

    cin >> n;

    int u[n];

    for (int i = 0; i < n; i++)

    {

        cin >> u[i];

    }

    cout << "enter value to search" << endl;

    cin >> x;

    LS test;

    test.get\_data(x);

    y = test.lsearch(u, n);

    if (y == -1)

    {

        cout << "value not found" << endl;

    }

    else

    {

        cout << "the position is " << y << endl;

    }

}

#include <iostream>

using namespace std;

class Binarys

{ Topic 2

private:

    int b, c, x;

public:

    void get\_data(int beg, int end, int search)

    {

        b = beg;

        c = end;

        x = search;

    }

    int binar(int a[])

    {

        while (b <= c)

        {

            int m = (b + c) / 2;

            if (a[m] == x)

            {

                return m;

            }

            else if (a[m] < x)

            {

                b = m + 1;

            }

            else if (a[m] > x)

            {

                c = m - 1;

            }

        }

        return -1;

    }

};

int main()

{

    int n, x, result;

    cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\* BINARY SEARCH \*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

    cout << "\n\n";

    cout << "Enter Your Array Size :=" << endl;

    cin >> n;

    int beg = 0, end = n - 1;

    int y[n];

    cout << "Enter The Array Elements in a Sorted Manner :=" << endl;

    for (int i = 0; i < n; i++)

        cin >> y[i];

    cout << "Enter number to be searched :=" << endl;

    cin >> x;

    Binarys test;

    test.get\_data(beg, end, x);

    result = test.binar(y);

    if (result == -1)

    {

        cout << "Value not found in the existing Array Or Your Array is not Sorted" << endl;

    }

    else

    {

        cout << "The Number is located at Index = " << result << endl;

    }

}

#include <iostream>

using namespace std;

class Sp

{ **Topic 3**

private:

    int k = 0;

    int n, m;

    int s[100][3], z[100][100];

public:

    void get\_data\_display(int a, int b)

    {

        n = a;

        m = b;

        for (int i = 0; i < n; i++)

            for (int j = 0; j < m; j++)

                cin >> z[i][j];

        cout << "Displaying Main Matrix " << endl;

        for (int i = 0; i < n; i++)

        {

            for (int j = 0; j < m; j++)

                cout << z[i][j] << " ";

            cout << endl;

        }

    }

    void sparsing()

    {

        for (int i = 0; i < n; i++)

        {

            for (int j = 0; j < m; j++)

            {

                if (z[i][j] != 0)

                {

                    k++;

                    s[k][0] = i;

                    s[k][1] = j;

                    s[k][2] = z[i][j];

                }

            }

        }

        s[0][0] = n;

        s[0][1] = m;

        s[0][2] = k;

    }

    void display\_sparsing()

    {

        cout << "Displaying Sparse Matrix Table...." << endl;

        cout << "Rows Columns Elements" << endl;

        for (int i = 0; i <= k; i++)

        {

            for (int j = 0; j < 3; j++)

            {

                cout << s[i][j] << " ";

            }

            cout << endl;

        }

    }

};

int main()

{

    Sp test;

    int x, y;

    cout << "Enter Row and Column Size (max: 100)" << endl;

    cin >> x >> y;

    cout << "Enter The Main Matrix elements: " << endl;

    test.get\_data\_display(x, y);

    test.sparsing();

    test.display\_sparsing();

}

#include <iostream>

using namespace std;

**Menu Program**

class LS

{

private:

    int x, n, b, c;

    int a[100];

public:

    void set\_data()

    {

        int a;

        cout << "Enter value to search: " << endl;

        cin >> a;

        x = a;

    }

    void display()

    {

        for (int i = 0; i < n; i++)

        {

            cout << a[i] << endl;

        }

    }

    void get\_data()

    {

        cout << "Enter your array size:" << endl;

        cin >> n;

        b = 0;

        c = n - 1;

        cout << "Enter the array elements in sorted manner(max 100):" << endl;

        for (int i = 0; i < n; i++)

        {

            cin >> a[i];

        }

    }

    int lsearch()

    {

        for (int i = 0; i < n; i++)

        {

            if (a[i] == x)

            {

                return i;

            }

        }

        return -1;

    }

    int binar()

    {

        while (b <= c)

        {

            int m = (b + c) / 2;

            if (a[m] == x)

            {

                return m;

            }

            else if (a[m] < x)

            {

                b = m + 1;

            }

            else if (a[m] > x)

            {

                c = m - 1;

            }

        }

        return -1;

    }

};

int main()

{

    int x, n, y, choice;

    LS test;

    while (choice != 5)

    {

        cout << "\*\*\* Menu \*\*\*" << endl;

        cout << "1. Create Array" << endl;

        cout << "2.linear search" << endl;

        cout << "3. Binary Search" << endl;

        cout << "4. Display Array" << endl;

        cout << "5. Exit" << endl;

        cout << "\n\nEnter your Option" << endl;

        cin >> choice;

        switch (choice)

        {

        case 1:

            test.get\_data();

            break;

        case 2:

            test.set\_data();

            y = test.lsearch();

            if (y == -1)

            {

                cout << "value not found" << endl;

            }

            else

            {

                cout << "the position is " << y << endl;

            }

            break;

        case 3:

            test.set\_data();

            test.binar();

            y = test.lsearch();

            if (y == -1)

            {

                cout << "value not found" << endl;

            }

            else

            {

                cout << "the position is " << y << endl;

            }

            break;

        case 4:

            test.display();

        case 5:

            cout << "exiting..." << endl;

        }

    }

}