

## AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)

**FACULTY OF SCIENCE & TECHNOLOGY**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**DATA STRUCTURE LAB**

**Supervised By**

**MD. ISMAIL HOSSEN**

**Submitted By**

|  |  |
| --- | --- |
| **NAME** | **ID** |
| Md.Saiduzzaman Sohag | 22-46005-1 |

**Date of Submission:** **March 11, 2023**

**Question : 1:**

#include <iostream>

using namespace std;

void BubbleSort(int a[], int size)

{

    for (int i = 0; i < size - 1; i++)

    {

        for (int j = 0; j < size - 1 - i; j++)

        {

            if (a[j] > a[j + 1])

            {

                int temp = a[j];

                a[j] = a[j + 1];

                a[j + 1] = temp;

            }

        }

    }

}

int main()

{

    int arr[6] = {5, 4, 2, 8, 3, 1};

    BubbleSort(arr, 6);

    cout << "Sorted array: ";

    for (int i = 0; i < 6; i++)

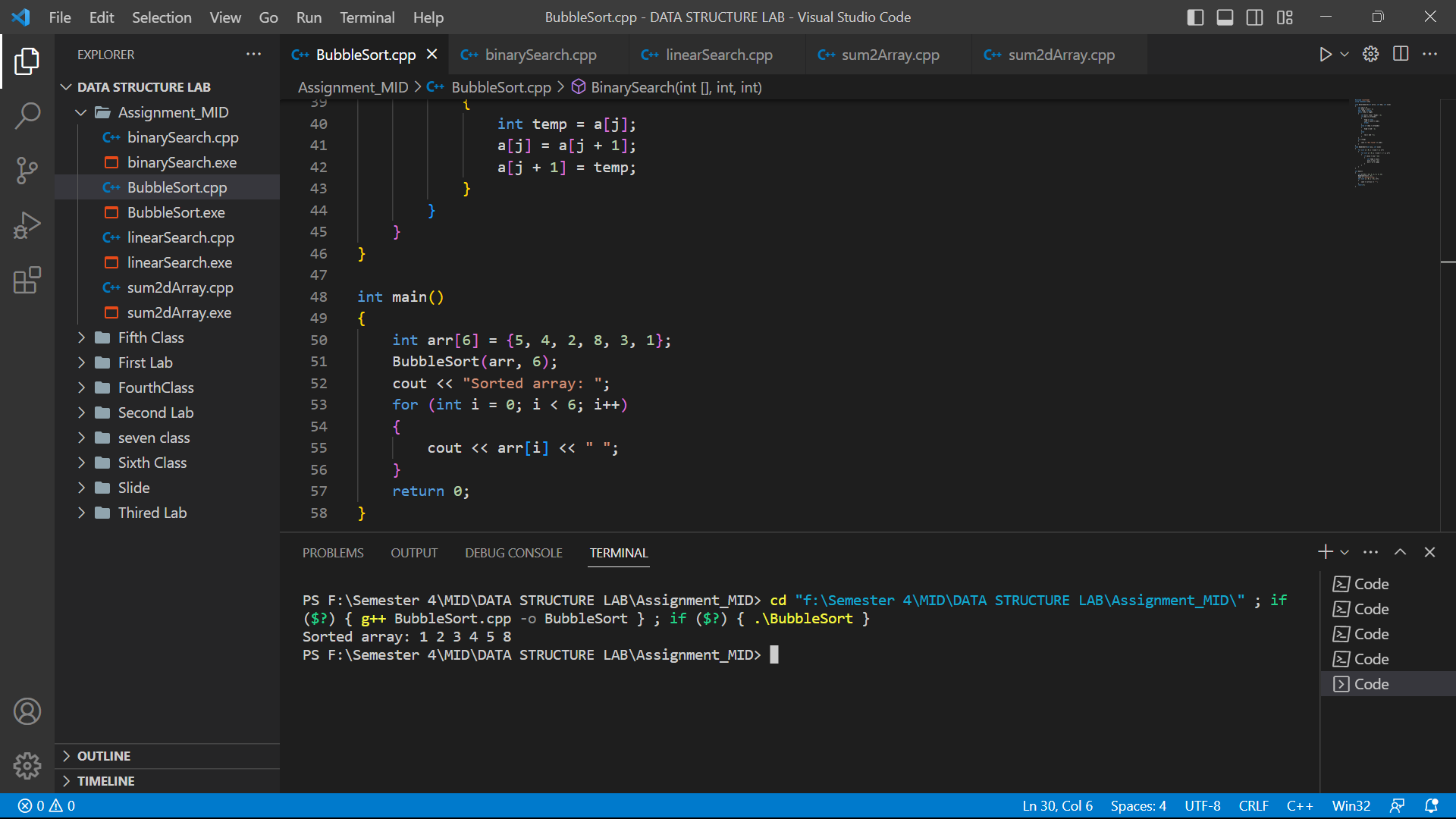
    {

        cout << arr[i] << " ";

    }

    return 0;

}



**Question : 2:**

#include <iostream>

using namespace std;

int BinarySearch(int arr[], int key, int size)

{

  int low = 0;

  int high = size - 1;

  bool flag = false;

  while (low <= high)

  {

    int mid = (low + high) / 2;

    if (key == arr[mid])

    {

      return mid;

      break;

    }

    else if (key < arr[mid])

    {

      high = mid - 1;

    }

    else

    {

      low = mid + 1;

    }

  }

  return -1;

}

int main()

{

  int arr[6] = {1, 2, 3, 4, 5, 6};

  int result = BinarySearch(arr, 5, 6);

  if (result == -1)

  {

    cout << "Element not found";

  }

  else

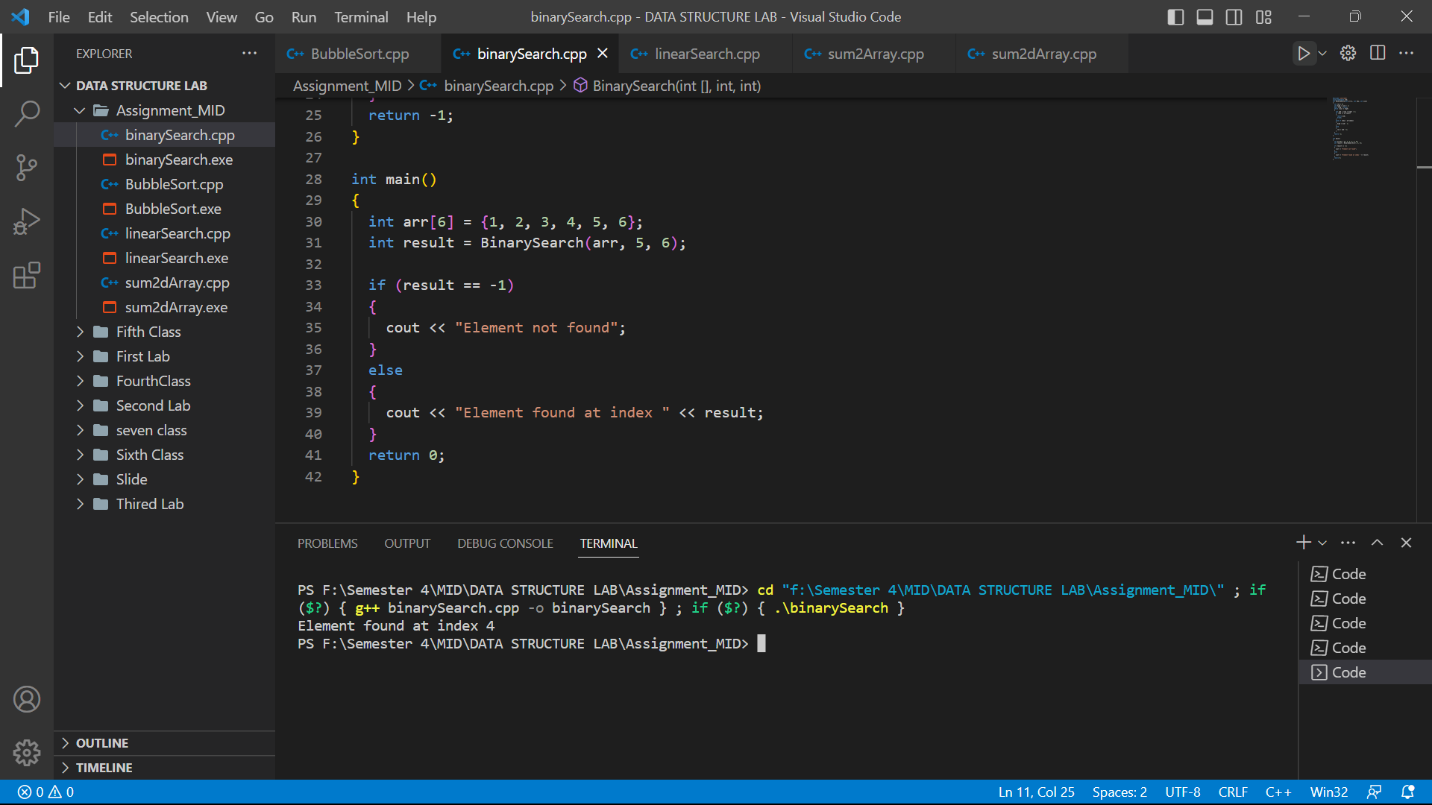
  {

    cout << "Element found at index " << result;

  }

  return 0;

}



**Question : 4:**

#include <iostream>

using namespace std;

int main()

{

    int row, column;

    cout << "Enter number of rows: ";

    cin >> row;

    cout << "Enter number of columns: ";

    cin >> column;

    string arr[row][column];

    cout << "Enter elements of array: " << endl;

    for (int i = 0; i < row; i++)

    {

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

        {

            cin >> arr[i][j];

        }

    }

    int count = 0;

    string newArr[row \* column];

    for (int i = 0; i < row; i++)

    {

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

        {

            if (arr[i][j].length() <= 4)

            {

                newArr[count++] = arr[i][j];

            }

        }

    }

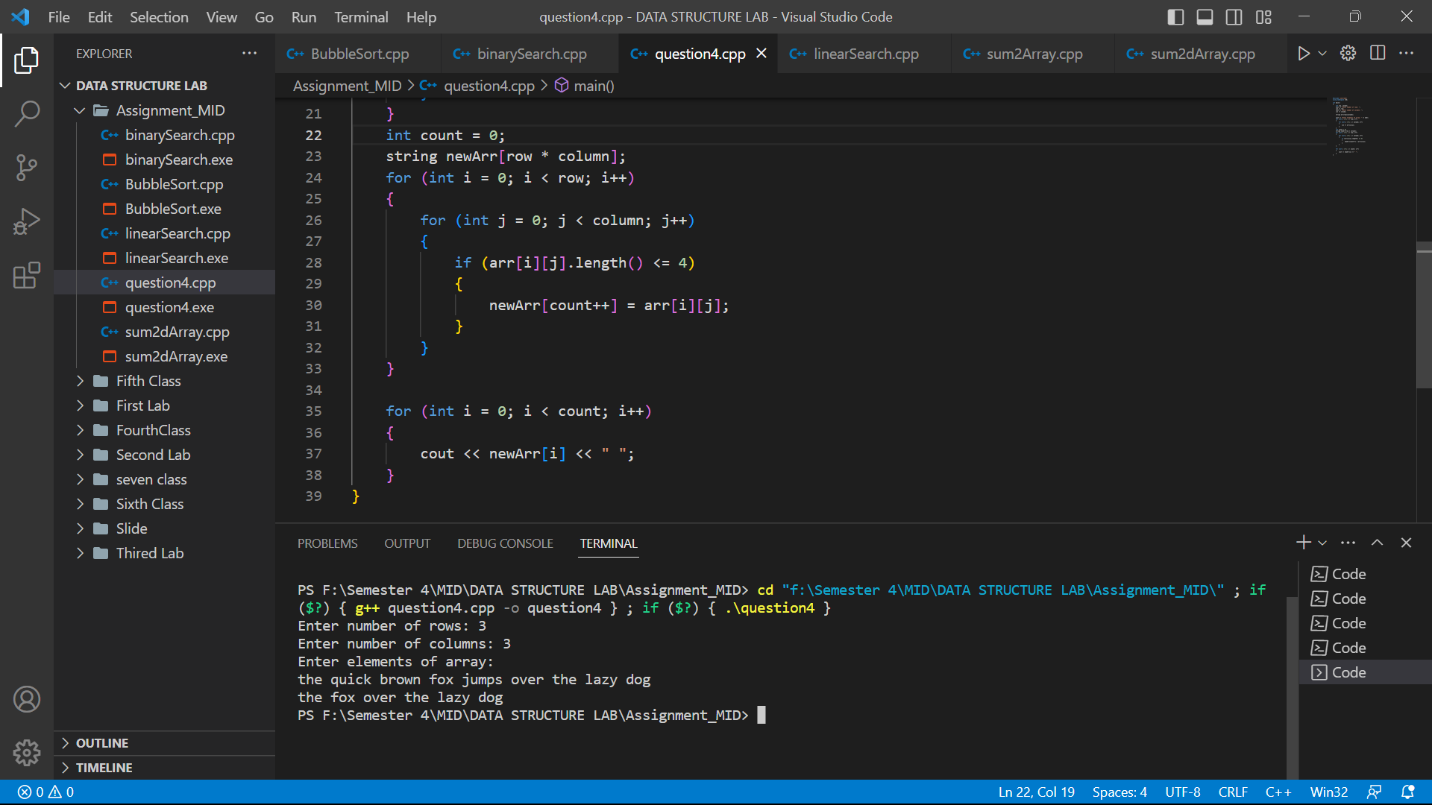
    for (int i = 0; i < count; i++)

    {

        cout << newArr[i] << " ";

    }

}



**Question : 5:**

#include <iostream>

using namespace std;

int main()

{

  int row1, column1;

  cout << "Enter the first arry row number: ";

  cin >> row1;

  cout << "Enter the first arry column number: ";

  cin >> column1;

  int row2, column2;

  cout << "Enter the second array row number: ";

  cin >> row2;

  cout << "Enter the second array column number: ";

  cin >> column2;

  if (row1 != row2 || column1 != column2)

  {

    cout << "Enter same row and column two array " << endl;

  }

  else

  {

    int arr1[row1][column1];

    int arr2[row2][column2];

    cout << "Enter the first array elemrnt : ";

    for (int i = 0; i < row1; i++)

    {

      for (int j = 0; j < column1; j++)

      {

        cin >> arr1[i][j];

      }

    }

    cout << "Enter the second array elemrnt : ";

    for (int i = 0; i < row1; i++)

    {

      for (int j = 0; j < column1; j++)

      {

        cin >> arr2[i][j];

      }

    }

    int result[row1][column1];

    for (int i = 0; i < row1; i++)

    {

      for (int j = 0; j < column1; j++)

      {

        result[i][j] = arr1[i][j] + arr2[i][j];

      }

    }

    cout << "Resultant 2D array: " << endl;

    for (int i = 0; i < row1; i++)

    {

      for (int j = 0; j < column1; j++)

      {

        cout << result[i][j] << " ";

      }

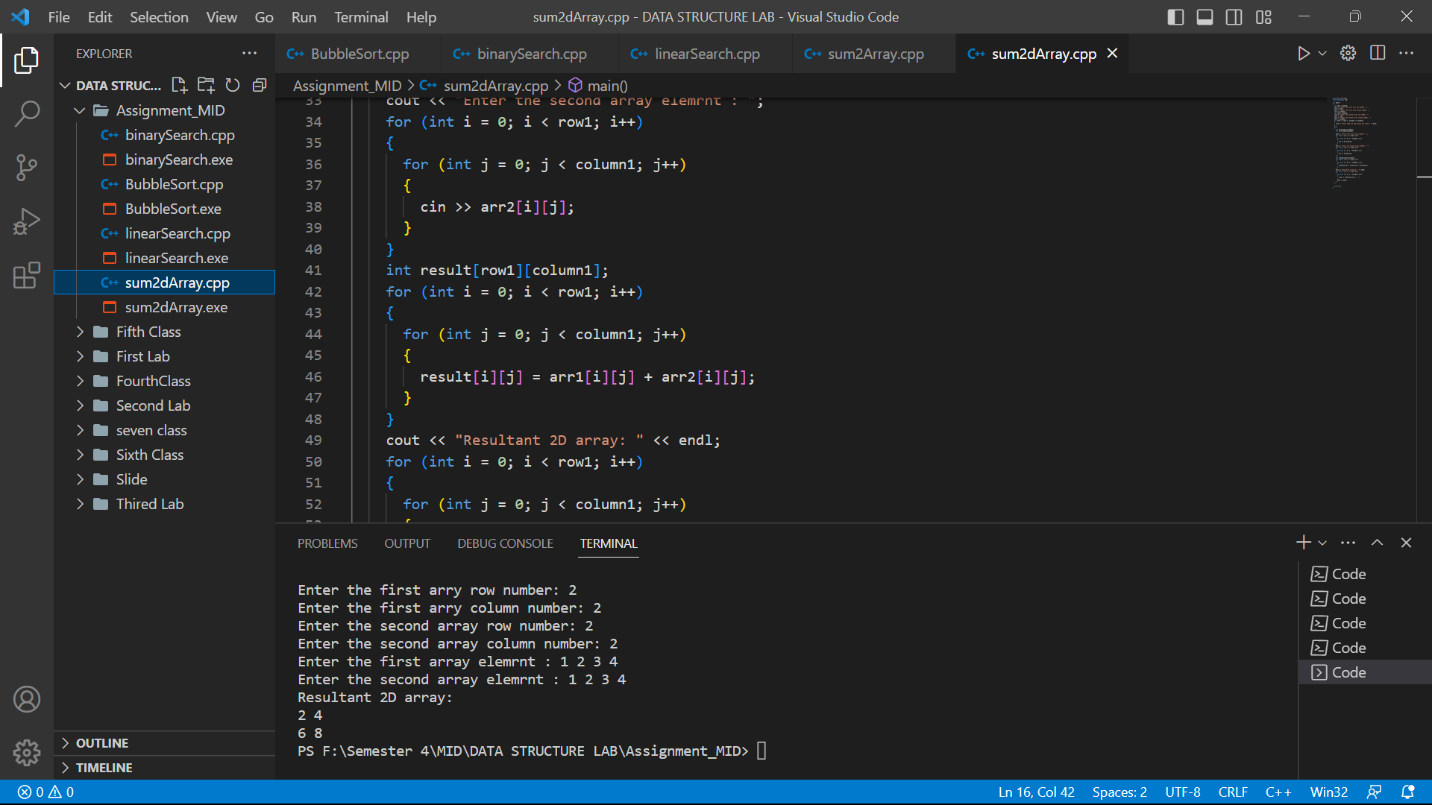
      cout << endl;

    }

  }

  return 0;

}



**Question : 6:**

#include <iostream>

using namespace std;

int LinearSearching(int arr[], int key, int size)

{

  for (int i = 0; i < size; i++)

  {

    if (arr[i] == key)

    {

      return i;

      break;

    }

  }

  return -1;

}

int main()

{

  int arr[6] = {5, 4, 2, 8, 3, 1};

  int result = LinearSearching(arr, 8, 6);

  if (result == -1)

  {

    cout << "Element not found";

  }

  else

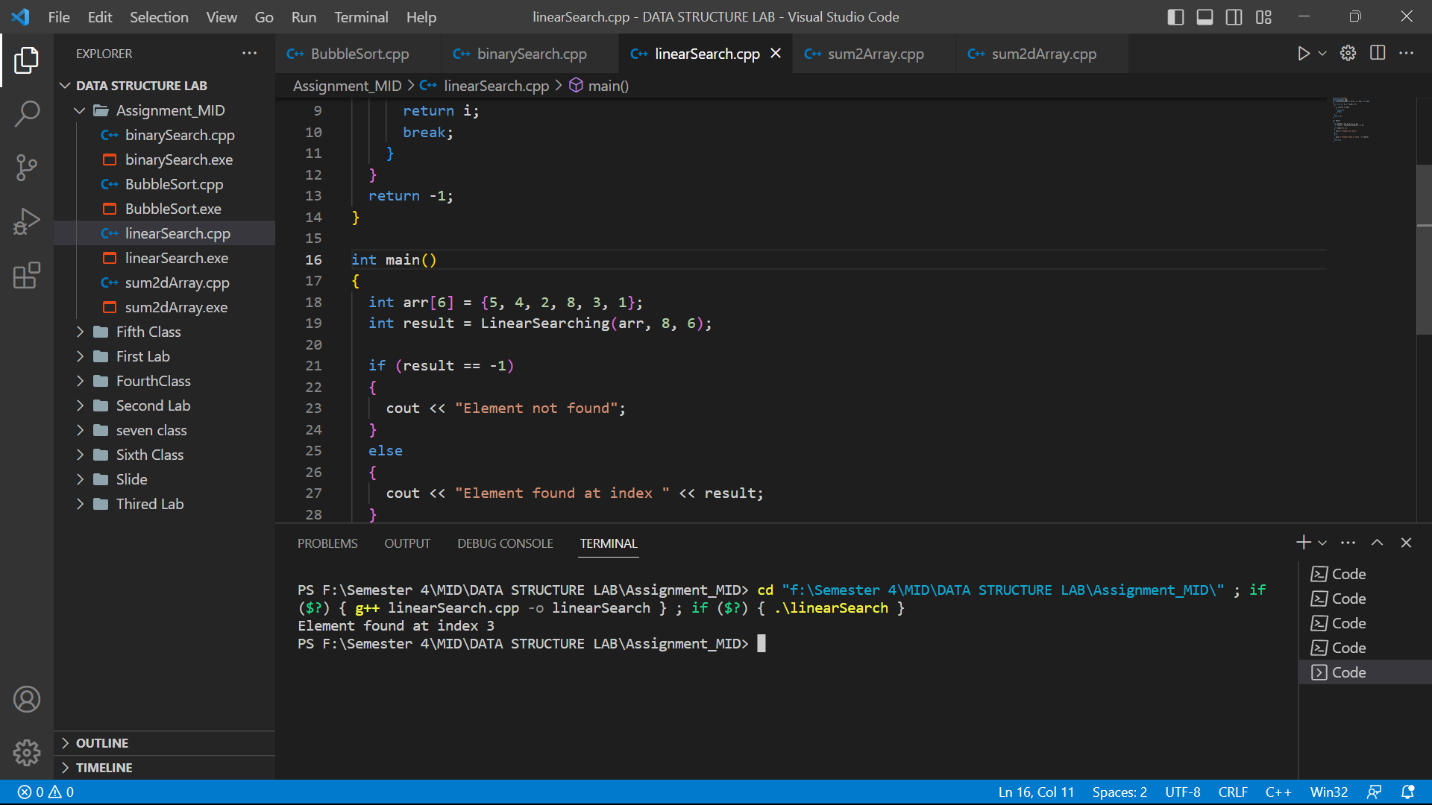
  {

    cout << "Element found at index " << result;

  }

  return 0;

}

s