

**Assignment No: 01**

Submitted by:

Name : Haseeb Ullah

ID: F20232661009

Section: V21

Submitted to:

Name: Sir M Owais Khan

PF LAB: 11

**Address:** C-II Block C 2 Phase 1 Johar Town, Lahore, Punjab 54770.

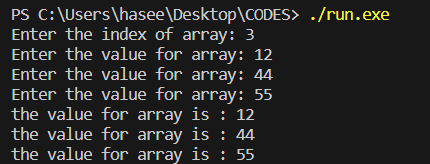
University of Management & Technology.

**EXAMPLE 1:**

Write a program that contains an array of five elements, take five values from user, assign the value to each element of array and display the all array elements.

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  int n;  cout << "Enter the index of array: ";  cin >> n;    int array [n];    for(int i = 0; i < n; i++)  {  cout << "Enter the value for array: ";  cin >> array[i];  }  for(int j = 0; j < n; j++)//this loop will print the array  {  cout << "the value for array is : " << array[j] << endl;    }  return 0;  } |

**Output**

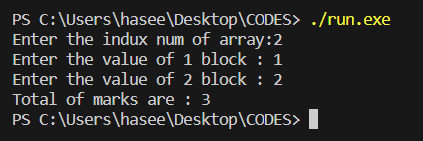
****

**EXAMPLE 2:**

Write a C++ program that adds all elements of array.

|  |
| --- |
| #include <iostream>  using namespace std;  //program that adds all elements of array.  int main() {  int sum = 0;  int n;  cout << "Enter the indux num of array:";  cin >> n;  int num[n];  for (int i = 0; i < n; i++)  {  cout << "Enter the value of "<< i + 1 << " block : " ;  cin >> num[i];  }  for (int i = 0; i <= n; i++)  {  sum += num[i];    }  cout << "Total of marks are : " << sum;  return 0;  } |

**Output**

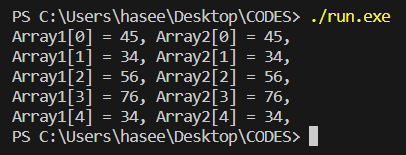


**EXAMPLE 3:**

Suppose there are two arrays of the same size. Array A consists of five elements while array B in not initialized yet. Write a program that copies the values of A into B. Verify your code by displaying the contents of B.

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  int array1[]={45,34,56,76,34};  int array2[5];  for (int i = 0; i < 5; i++)  {  array2[i] = array1[i];  cout<<"Array1["<<i<<"] = " <<array1[i]<<", ";  cout<<"Array2["<<i<<"] = " <<array2[i]<<", " << endl;  }  return 0;  } |

**Output**

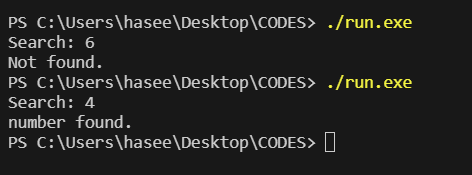
****

**Example 4:**

Write a program that takes an integer value from user and search it in array, If found then display message ‘Number Found’ else ‘Number Not Found.

|  |
| --- |
| #include <iostream>  using namespace std;  // Write a program that takes an integer value from user and search it in array,  // If found then display message ‘Number Found’ else ‘Number Not Found.  int main() {  int values [10]= {23,34,45,5,65,56,4,32,89,7};  int n,i;  cout << "Search: ";  cin >> n;  for (int i = 0; i < 10; i++)  {  if(values[i] == n)  {  cout << "number found."<<endl;  break;  }  else if(i == n-1)  {  cout << "Not found."<<endl;  }  }    return 0;  } |

**Output**

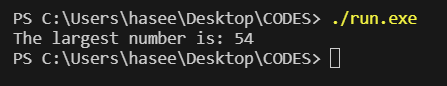
****

**Example 5:**

Write a Program that finds maximum number in an array

|  |
| --- |
| #include <iostream>  using namespace std;  // Write a Program that finds maximum number in an array  int main()  {  int i, max\_num;  int arr[4]={34,54,6,3};  max\_num=arr[0] ;  for (int i = 0; i < 4; i++)  {  if(arr[i]>max\_num)  {  max\_num=arr[i];  }  }  cout << "The largest number is: " << max\_num;    return 0;  } |

**Output**

****

**Exercise 1:**

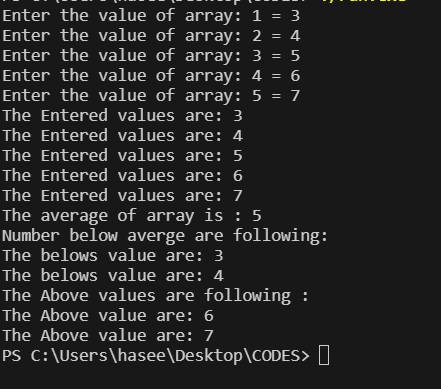
Write a program that declares an array of 5 double numbers. Use a loop to read 5 real numbers from user and fill the array. Then print the following on screen:

* + 1. All the elements of array
    2. The average of all the numbers in the array
    3. The numbers below average
    4. The numbers above average.

**PART (a – b – c – d )**

|  |
| --- |
| #include <iostream>  using namespace std;  // Write a program that declares an array of 5 double numbers.  // Use a loop to read 5 real numbers from user and fill the array. Then print the following on screen:  // a. All the elements of array  // b. The average of all the numbers in the array  // c. The numbers below average  // d. The numbers above average.  int main() {    double arr[5];  double sum = 0, average = 0;  for (int i = 0; i < 5; i++)  {  cout << "Enter the value of array: "<< i + 1 << " = ";  cin >> arr[i];  }  for (int i = 0; i < 5; i++)  {  cout << "The Entered values are: " << arr[i]<<endl;  }  for (int i = 0; i <= 5; i++)  {  sum += arr[i];  }  average = sum / 5; // which can also be N number input by user.  cout << "The average of array is : "<< average << endl;  cout << "Number below averge are following: " << endl;  for (int i = 0; i < 5; i++)  {  if (arr[i] < average)  {  cout << "The belows value are: " << arr[i] << endl;  }  }  cout << "The Above values are following : " << endl;    for (int i = 0; i < 5; i++)  {  if (arr[i] > average)  {  cout << "The Above value are: " << arr[i] << endl;  }  }  return 0;  } |

**Output:**

****

**Exercises 2:**

Consider the following list of student’s grade

64 36 56 47 40 54 61 60 58 64 54

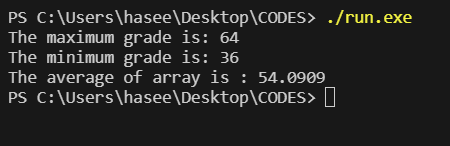
48 59 45 63 54 50 49 51 60 58 59

Initialize an array with above grades and find the following things about the above data.

* + 1. The minimum grade
    2. The maximum grade
    3. Average

|  |
| --- |
| #include <iostream>  using namespace std;  // Exercises 2:  // Consider the following list of student’s grade  // 64 36 56 47 40 54 61 60 58 64 54  // 48 59 45 63 54 50 49 51 60 58 59  // Initialize an array with above grades and find the following things about the above data.  // a. The minimum grade  // b. The maximum grade  // c. Average  int main() {  const int indux\_of\_array = 22;  long double arr[indux\_of\_array]={64,36, 56, 47, 40, 54, 61, 60, 58, 64, 54, 48, 59, 45, 63, 54, 50, 49, 51, 60, 58, 59};  double sum = 0, average = 0, max, min;  max = arr[0];  // b. The maximum grade  for (int i = 0; i < indux\_of\_array; i++)  {  if (arr[i] > max)  {  max = arr[i];  }    }  cout << "The maximum grade is: " << max <<endl;  min = arr[0];  for (int i = 0; i < indux\_of\_array; i++)  {  if (arr[i] < min)  {  min = arr[i];  }    }  cout << "The minimum grade is: " << min <<endl;  for (int i = 0; i < indux\_of\_array; i++)  {  sum += arr[i];  }  average = sum / indux\_of\_array; // indux, which is number of array.  cout << "The average of array is : "<< average << endl;  return 0;  } |

**Output:**

****

**HOME WORK**

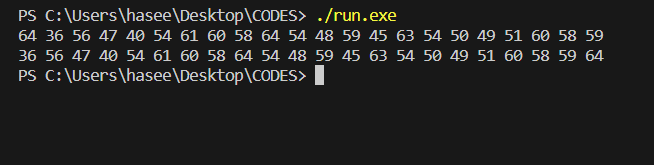
1. Apply Bubble sort algorithms to sort elements of a given array.

64 36 56 47 40 54 61 60 58 64 54

48 59 45 63 54 50 49 51 60 58 59

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  int marks[22] = {64, 36, 56, 47, 40, 54, 61, 60, 58, 64, 54, 48, 59, 45, 63, 54, 50, 49, 51, 60, 58, 59};  int temp;  for (int i = 0; i < 22; i++)  {  cout << marks[i] << " ";  }    cout << endl;  for (int j = 0; j < 22 - 1; j++) {  temp = marks[j];  marks[j] = marks[j + 1];  marks[j + 1] = temp;  }  for (int i = 0; i < 22; i++) {  cout << marks[i] << " ";  }  return 0;  } |

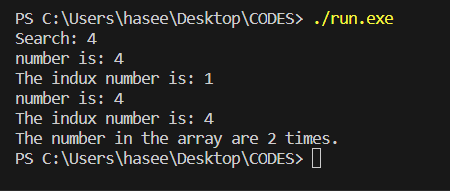
**Output**

****

1. Write a program to search a value in an array. Your program should also notify:
   1. Index of the array where value found
   2. Number of times the value was found in array

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  int numbers[5]= {34, 4, 34,56,4};  int search,times = 0;  cout << "Search: ";  cin >> search;  for (int i = 0; i < 5; i++)  {  if (search == numbers[i])  {  cout << "number is: " << numbers[i]<<endl;  cout << "The indux number is: " << i <<endl;  times++ ;  }    }  cout << "The number in the array are " << times << " times."<<endl;    return 0;  } |

**Output**

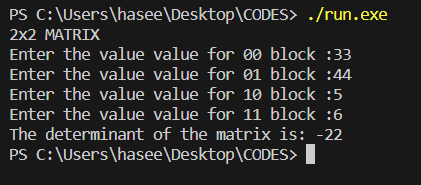
****

**Two Dimensional Arrays**

**Exercise 11.4** A program to input a matrix and find its determent (by using function).

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  double marks[2][2];  cout << "2x2 MATRIX"<<endl;  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  cout << "Enter the value value for "<<i<<j << " block :";  cin >> marks[i][j];  }    }    double determinant=0;  determinant = ((marks[0][0]\*marks[1][1]) - (marks[1][0]\*marks[0][1]));  cout <<"The determinant of the matrix is: "<< determinant;  return 0;  } |

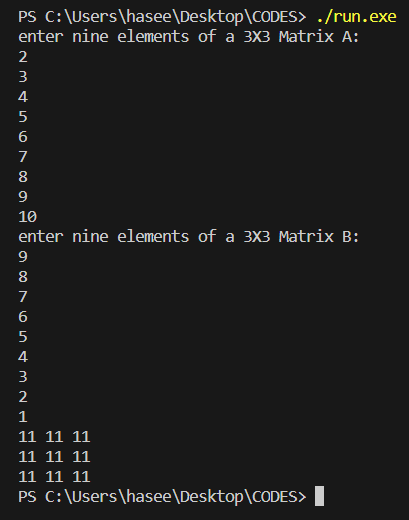
**Output**

****

**Example 11.2:** Write a program that will add two matrixes entered by the user and print it.

|  |
| --- |
| #include <iostream>  using namespace std;  int main()  {      int firstarr[3][3],secondarr[3][3],thridarr[3][3];      int i,j;      cout<<"enter nine elements of a 3X3 Matrix A:";      for(i=0; i<3; i++)      {          for(j=0; j<3; j++)          cin>>firstarr[i][j];      }      cout<<"enter nine elements of a 3X3 Matrix B:";      for(i=0; i<3; i++)      {          for(j=0; j<3; j++)          cin>>secondarr[i][j];      }      for(i=0 ; i<3 ; i++)      {          for(j=0 ; j<3 ; j++)          {              thridarr[i][j]=firstarr[i][j]+secondarr[i][j];              cout<<thridarr[i][j]<<" ";          }          cout<<"\n";      }      return 0;  } |

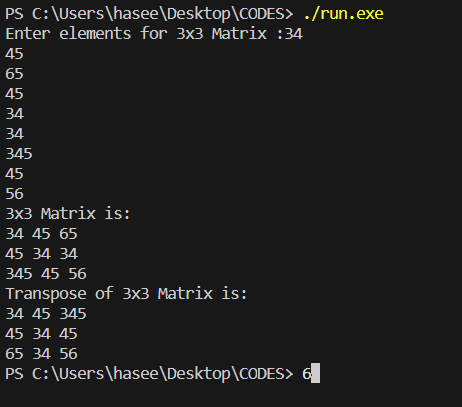
**Output**

****

**Example 11.3:** A program to input a matrix and print its transpose**.**

|  |
| --- |
| #include <iostream>  using namespace std;  int main()  {  int arr[3][3];  int i,j;    cout<<"Enter elements for 3x3 Matrix :";  for(i=0; i<3; i++)  {  for(j=0; j<3; j++)  cin>>arr[i][j];  }  cout<<"3x3 Matrix is:"<<endl;  for(i=0 ; i<3 ; i++)  {  for(j=0 ; j<3 ; j++)  {  cout<<arr[i][j]<<" ";  }  cout<<"\n";  }  cout<<"Transpose of 3x3 Matrix is:"<<endl;  for(i=0 ; i<3 ; i++)  {  for(j=0 ; j<3 ; j++)  {  cout<<arr[j][i]<<" ";  }  cout<<"\n";  }  return 0;  } |

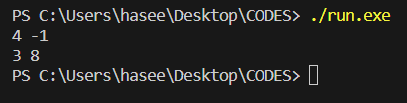
**Output**

****

**Exercise 11.2** Multiply two matrices and store result in third matric.

|  |
| --- |
| // Exercise 11.2 Multiply two matrices and store result in third matric.  #include <iostream>  using namespace std;  int main()  {  int mat1[2][2]={  {1,1},  {3,-2}  };  int mat2[2][2]={  {1,1},  {3,-2}  };  int mat3[2][2];  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  mat3[i][j] = 0;  mat3[i][j]=mat1[i][j]\*mat2[i][j];  for (int k = 1; k < 2; k++)  {  mat3[i][j] += mat1[i][k]\*mat2[k][j];  }  }    }    for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  cout << mat3[i][j]<< " ";  }  cout<<endl;  }  return 0;  } |

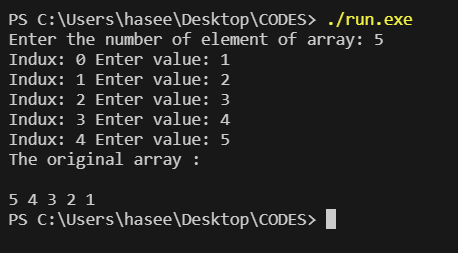
**Output**

****

**Exercise 11.3** Write a program that will sort 2d array in ascending order and descending order.

|  |
| --- |
| #include <iostream>  using namespace std;  // Exercise 11.3 Write a program that will sort 2d array in ascending order and descending order.  int main(){  int n;  cout << "Enter the number of element of array: ";  cin >> n;  int arr[n];  int temp;  for (int i = 0; i < n; i++)  {  cout << "Indux: "<< i << " Enter value: ";  cin >> arr[i];  }  cout << "The original array :" << endl;    cout << endl;  for (int i = 0; i < n - 1; i++) {  for (int j = 0; j < n - i - 1; j++) {  temp = arr[j];  arr[j] = arr[j + 1];  arr[j + 1] = temp;  }  }  for (int i = 0; i < n; i++) {  cout << arr[i] << " ";  }  return 0;  } |

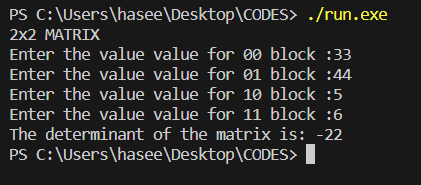
**Output**

****

**Exercise 11.4** A program to input a matrix and find its determent (by using function).

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  double marks[2][2];  cout << "2x2 MATRIX"<<endl;  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  cout << "Enter the value value for "<<i<<j << " block :";  cin >> marks[i][j];  }    }    double determinant=0;  determinant = ((marks[0][0]\*marks[1][1]) - (marks[1][0]\*marks[0][1]));  cout <<"The determinant of the matrix is: "<< determinant;  return 0;  } |

**Output**

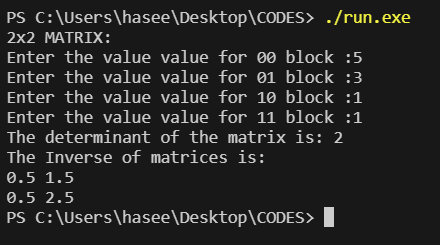
****

* 1. **Home Work**

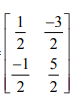
1. A program to input a matrix and find its inverse.

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  double marks[2][2];  cout << "2x2 MATRIX:"<<endl;  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  cout << "Enter the value value for "<<i<<j << " block :";  cin >> marks[i][j];  }    }    double determinant=0;  determinant = ((marks[0][0]\*marks[1][1]) - (marks[1][0]\*marks[0][1]));  cout <<"The determinant of the matrix is: "<< determinant << endl;  // A = 1 adj // (-1)1+n  // |A|  double adjuent, temp;  if (determinant != 0)  {    temp = marks[0][0];  marks[0][0] = marks[1][1];  marks[1][1] = temp;    (marks[0][1])\*(-1);  (marks[1][0])\*(-1);      for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  marks[i][j] = (1\* marks[i][j])/determinant;  }    }  cout << "The Inverse of matrices is:" <<endl;  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  cout << marks[i][j] << " ";  }  cout << endl;    }  }else {  cout << "its not possible"<<endl;  }  return 0;  } |

**Output**

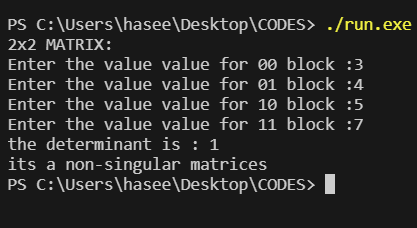
****

**Book Answer:**

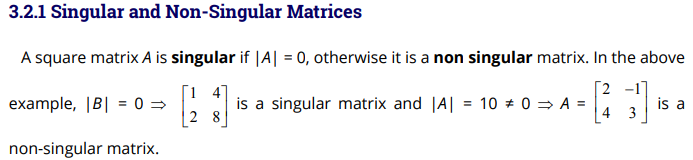
****

1. A program to input a matrix and find its singular or non-singular (by using function).

|  |
| --- |
| #include <iostream>  using namespace std;  // 1. A program to input a matrix and find its singular or non-singular (by using function).  int main() {  double marks[2][2];  cout << "2x2 MATRIX:"<<endl;  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  cout << "Enter the value value for "<<i<<j << " block :";  cin >> marks[i][j];  }    }      double determinant=0;  determinant = ((marks[0][0]\*marks[1][1]) - (marks[1][0]\*marks[0][1]));  cout << "the determinant is : " << determinant<<endl;  if (determinant == 0)  {  cout << " Its a singular matrics"<<endl;  }else  {  cout << "its a non-singular matrices";  }    return 0;  } |

**Output**

**Book formula:**

****