TASK 1:

#include <iostream>

using namespace std;

void printcommonelements (int arr1[], int arr2[], int m, int n){

int i = 0, j = 0;

while (i < m && j < n) {

if (arr1[i] < arr2[j])

i++;

else if (arr2[j] < arr1[i])

j++;

else {

cout << arr2[j] << " ";

i++;

j++;

}

}

}

int main(){

int size1, size2;

int arr1[100];

int arr2[100];

cout << "Enter size of array 1 : ";

cin >> size1;

cout << "Enter size of array 2 : ";

cin >> size2;

cout << "--------------"<<endl;

for (int i = 0; i < size1; i++) {

cout << "enter the value of array1 in index " << i<<" : ";

cin >> arr1[i]; //taking input in array 1

}

cout << "--------------"<<endl;

for (int i = 0; i < size2; i++) {

cout << "enter the value of array2 in index " << i << " : ";

cin >> arr2[i]; //taking input in array 2

}

int m = sizeof(arr1) / sizeof(arr1[0]);

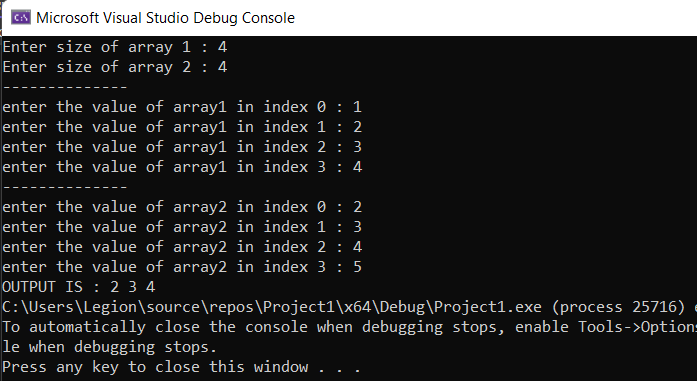
int n = sizeof(arr2) / sizeof(arr2[0]);

cout << "OUTPUT IS : ";

printcommonelements(arr1, arr2, m, n); //function calling

return 0;

}



TASK 2:

//

#include <iostream>

#include <string>

using namespace std;

int rows;

int\* numbers = new int[rows];

int i = 0;

void outputt(int\*\* array) { //Output function for array

//Output values of array

for (int i = 0; i < rows; i++)

{

for (int j = 0; j < numbers[i]; j++)

{

cout << array[i][j] << " "; //printing the jagged array

}

cout << endl;

}

delete array; //deleting dynamic array

}

void inputt() { //input function for array

int\*\* array = new int\* [rows]; //jagged array

while (i < rows)

{

cout << "Enter column in row " << i << ": ";

cin >> numbers[i];

array[i] = new int[numbers[i]]; //creating new dynamic memory

i++;

}

//Input values in array

for (int i = 0; i < rows; i++)

{

for (int j = 0; j < numbers[i]; j++)

{

cout << "Row " << i << ":- input value " << i \* numbers[i] + j << ": ";

cin >> array[i][j];

}

}

outputt(array);

}

using namespace std;

int main() {

cout << "ENTER rows of array: ";

cin >> rows;

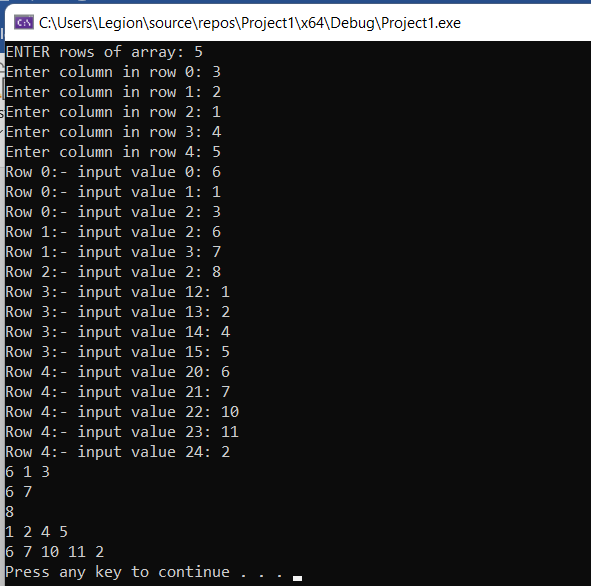
//declaration of array

inputt();

system("pause");

return 0;

}



TASK 3:

#include<iostream>

using namespace std;

int main()

{

/\*int size;

cin >> size;\*/

cout << "YOUR SENTINAL VALUE IS -1" << endl;

int\* ptr = new int[999], n;

for (int i = 0; i < 999; i++)

{

cout << "Enter Array1: ";

cin >> n;

if (n == -1) //SENTINAL VALUE IS -1

{

exit(0);

int\* ptr1 = new int[i];

for (int i = 0; i < i; i++)

{

ptr1[i] = ptr[i];

}

}

else

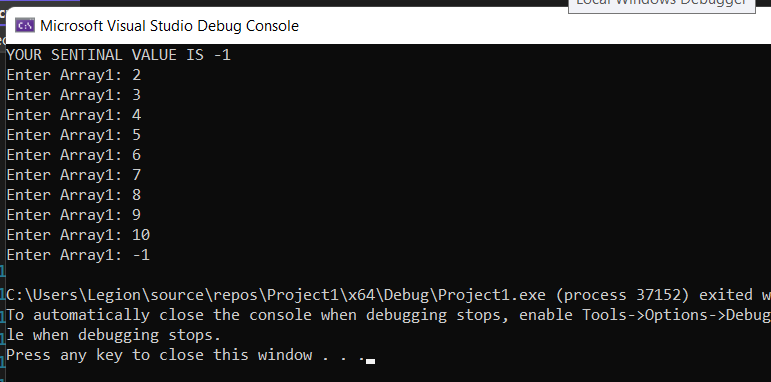
{

ptr[i] = n;

}

}

}



TASK 4:

#include <iostream>

using namespace std;

int main(){

void bubblesort(int\*, int);

int N; //array size

cout << "Enter the size of array : ";

cin >> N;

int arr[100]; //array in which we are storing

for (int i = 0; i < N; i++) {

cin >> arr[i];

}

bubblesort(arr, N); //sorting the array by calling this function

for (int j = 0; j < N; j++) //printing the sorted array

cout << arr[j] << " ";

cout << endl;

return 0;

}

void bubblesort(int\* ptr, int n){

void order(int\*, int\*);

int j, k; //indexes to array

for (j = 0; j < n - 1; j++)

for (k = j + 1; k < n; k++)

order(ptr + j, ptr + k);

}

void order(int\* numb1, int\* numb2) {

if (\*numb1 > \*numb2) //if 1st larger than 2nd,

{

int temp = \*numb1; //swap

\*numb1 = \*numb2;

\*numb2 = temp;

}

}

