

**UNIVERSITY OF THE WEST OF ENGLAND**

**THE BRITISH COLLEGE**



**Submitted By:**

**Albin Pokharel**

**Module:Programming in C++**

1.Write a program that assigns two positive values into two integer variables and displays the remainder of their division without using the % operator. Use only two variables.

#include <iostream>

using namespace std;

int remainder(int dividend, int divisor) {

while (dividend >= divisor) {

dividend -= divisor;

}

return dividend;

}

int main() {

int num1, num2;

// Taking input from the user

cout << "Enter the first positive integer: ";

cin >> num1;

cout << "Enter the second positive integer: ";

cin >> num2;

int remainders = remainder(num1, num2);

cout << "The remainder of " << num1 << " divided by " << num2 << " is: " << remainders <<

endl;

return 0;

}



2.Write a program that assigns two positive values into two float variables and displays the integer part of their division and the fractional part. For example, if they are assigned the values 7.2 and 5.4, the program should display 1 and 1.8, since 7.2 = (1×5.4)+1.8.[Hint: use math functions]

#include <iostream>

#include <cmath>

using namespace std;

int main() {

float num1, num2;

// Taking input from the user

cout << "Enter the first positive value: ";

cin >> num1;

cout << "Enter the second positive value: ";

cin >> num2;

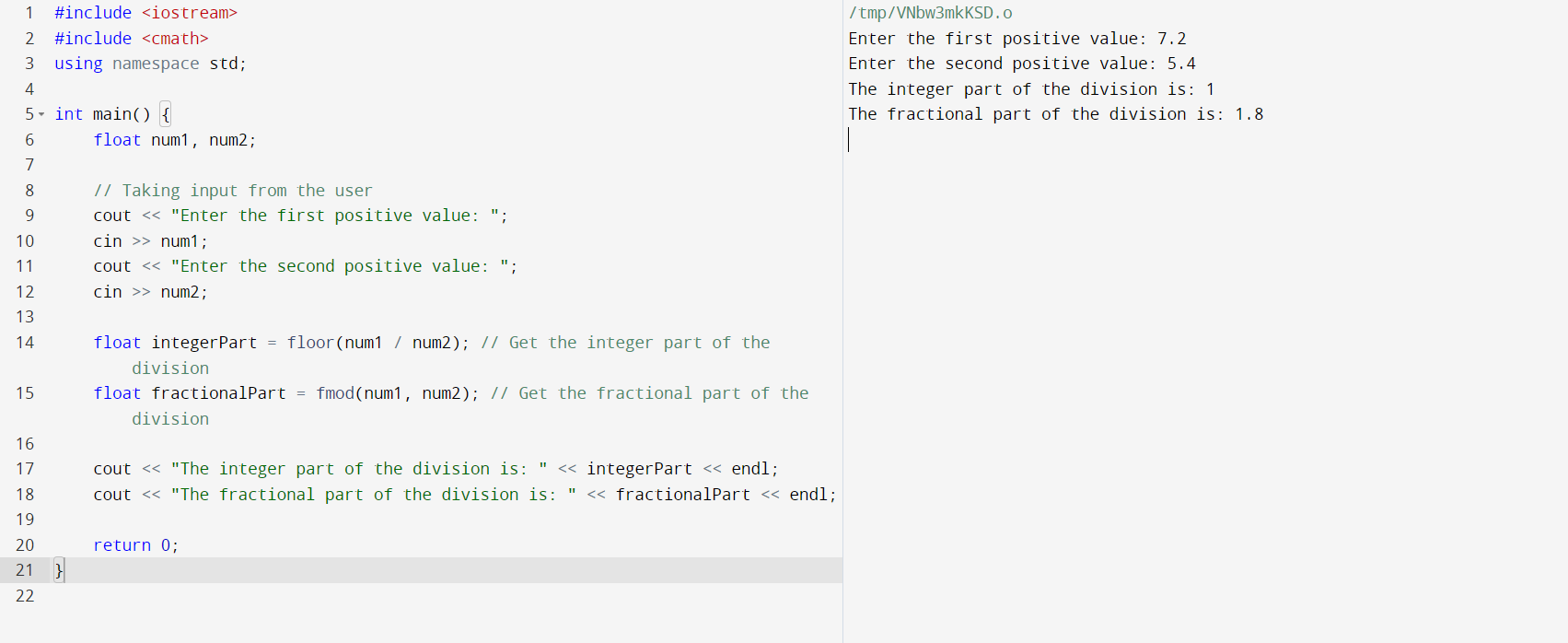
float integerPart = floor(num1 / num2); // Get the integer part of the division

float fractionalPart = fmod(num1, num2); // Get the fractional part of the division

cout << "The integer part of the division is: " << integerPart << endl;

cout << "The fractional part of the division is: " << fractionalPart << endl;

return 0;



3. Write a program that reads three integers, stores them in three variables, and rotates them one place right. For example, if the user enters the numbers 1, 2, and 3 and they are stored in variables a1, a2, and a3, the program should rotate their values one place right, so that a1 becomes 3, a2 becomes 1, and a3 becomes 2. [Hint: Use swapping logic we learned in class].

#include <iostream>

using namespace std;

int main() {

int a1, a2, a3;

// Taking input from the user

cout << "Enter three integers: ";

cin >> a1 >> a2 >> a3;

int temp = a3; // Store the last value in a temporary variable

// Move the first two values one place right

a3 = a2;

a2 = a1;

// Assign the temporary variable value to the first variable

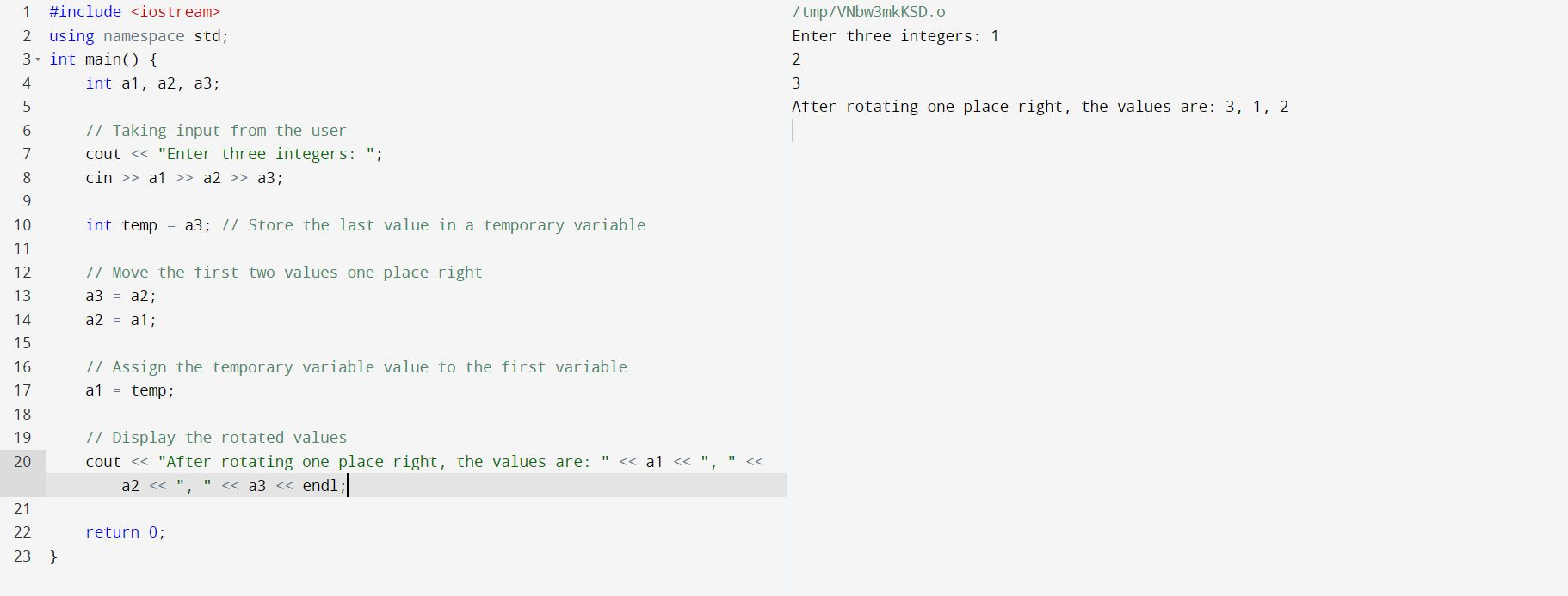
a1 = temp;

// Display the rotated values

cout << "After rotating one place right, the values are: " << a1 << ", " << a2 << ", " << a3 << endl;

return 0;

}



4. To change the PIN code , the user is asked to enter the current PIN code which is constant 1234 and then compare that code with code in SIM card. If they are the same, the user is asked to enter the new PIN code twice for verification, and if the same number is entered twice, it is stored in the SIM card. W.A.P that simulates this process..

#include <iostream>

using namespace std;

int main() {

int pin=1234;

int currentPIN;

int newPIN1,newPIN2;

cout<< "enter the current pin code:";

cin>> currentPIN;

if (currentPIN == pin){

cout<< "enter the new PIN code:";

cin >>newPIN1;

cout<<"enter the new pin code again:";

cin >>newPIN2;

if (newPIN1==newPIN2){

pin = newPIN1;

cout<< "pin changed succesfully. new pin is "<<pin<<endl;

}

else{

cout<<"pin verification failed.the new PINs do not match "<<endl;

}

}

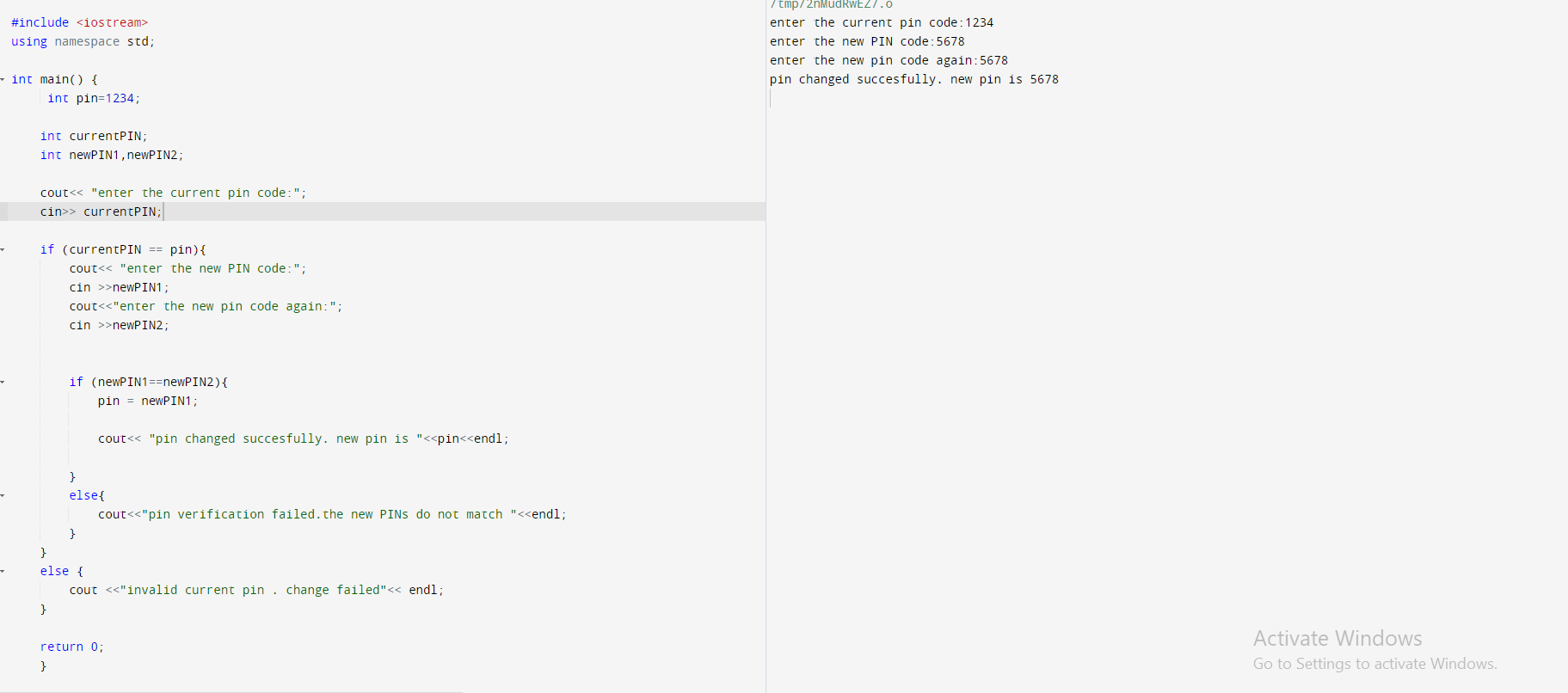
else {

cout <<"invalid current pin . change failed"<< endl;

}

return 0;

}



**Task 2: Programming Exercises:[Control Statements]**

**1.**

Write a program that reads the number of students in a class and their grades on a test. The program should display the average grade of the passed students, the average grade of the failed students, the minimum and maximum grade, and how many students got the same maximum grade. A student passes the exams with a grade ≥ 5. If the input grade is out of [0, 10] it should be ignored and the program should inform the user via a message, in order to enter a new value. Also, if the user enters -1, the insertion of grades should end.

A partial solution of the above program has been given below. Some part of it has been left with \_\_\_\_\_\_\_ to be filled by you. You have to write the required codes according to comments written against the lines. **[25 marks]**

Solution:

#include <iostream>

using std::cout;

using std::cin;

int main()

{

 int i, studs\_num, suc, fail, times;

 float grd, sum\_suc, sum\_fail, min\_grd, max\_grd;

 cout << "Enter number of students: ";

 cin >> studs\_num;

 if(studs\_num <= 0)

 {

 cout << "Wrong number of students\n";

 return 0; // Program termination.

 }

// 1. Initialize the variables with appropriate values**[5 marks]**

 suc = fail = \_\_\_\_\_\_\_;

 sum\_suc = sum\_fail = \_\_\_\_\_\_\_\_;

 min\_grd = \_\_\_\_\_\_\_\_;

 max\_grd = \_\_\_\_\_\_\_\_;

 for(i = 0; i < studs\_num; i++)

 {

 cout << "Enter grade: ";

 cin >> grd;

 if(grd == -1)

 break;

 if(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_) // 2. Write condition**[3 marks]**

 {

 cout << "Wrong grade, try again ...\n";

 i--; /\* If the input grade is out of [0, 10], the grade

is ignored and i is decremented to repeat the insertion. \*/

 continue;

 }

 if(grd >= 5)

 {

**//3. Insert your code here [5 marks]**

 }

 else

 {

**//4. Insert your code here [5 marks]**

 }

 if(grd < min\_grd)

 min\_grd = \_\_\_\_\_\_\_\_; //5. Insert code here **[2 marks]**

 if(grd > max\_grd)

 {

 max\_grd = \_\_\_\_\_\_\_\_; //6. Insert code here **[2 marks]**

 times = 1; // First appearance of the new maximum grade.

 }

else if(\_\_\_\_\_\_\_\_\_\_\_\_\_) // 7. Write condition**[3 marks]**

 times++;

 }

 if(i) // Check that at least one grade has been entered.

 {

 if(suc)

 cout << "Avg(+): " << sum\_suc/suc << '\n';

 else

 cout << "Everybody failed\n";

 if(fail)

 cout << "Avg(-): " << sum\_fail/fail << '\n';

 else

 cout << "None failed\n";

 cout << "Min: " << min\_grd << " Max: " << max\_grd << " (appeared "

<< times << " times)\n";

 }

 return 0;

}

ANSWER

#include <iostream>

using std::cout;

using std::cin;

int main()

{

int i, studs\_num, suc, fail, times;

float grd, sum\_suc, sum\_fail, min\_grd, max\_grd;

cout << "Enter number of students: ";

cin >> studs\_num;

if (studs\_num <= 0)

{

cout << "Wrong number of students\n";

return 0; // Program termination.

}

// 1. Initialize the variables with appropriate values

suc = fail = 0;

sum\_suc = sum\_fail = 0.0f;

min\_grd = 10.0f; // Set to maximum value initially

max\_grd = 0.0f;

times = 0;

for (i = 0; i < studs\_num; i++)

{

cout << "Enter grade: ";

cin >> grd;

if (grd == -1)

break;

if (grd < 0 || grd > 10) // 2. Check if the grade is valid

{

cout << "Wrong grade, try again ...\n";

i--; // Decrement i to repeat the insertion

continue;

}

if (grd >= 5)

{

sum\_suc += grd; // 3. Add grade to the sum of passing grades

suc++;

}

else

{

sum\_fail += grd; // 4. Add grade to the sum of failing grades

fail++;

}

if (grd < min\_grd)

min\_grd = grd; // 5. Update minimum grade

if (grd > max\_grd)

{

max\_grd = grd; // 6. Update maximum grade

times = 1; // Reset times for new maximum grade

}

else if (grd == max\_grd) // 7. Check if grade is the same as current maximum

times++;

}

if (i > 0) // Check that at least one grade has been entered.

{

if (suc > 0)

cout << "Avg(+): " << sum\_suc / suc << '\n';

else

cout << "Everybody failed\n";

if (fail > 0)

cout << "Avg(-): " << sum\_fail / fail << '\n';

else

cout << "None failed\n";

cout << "Min: " << min\_grd << " Max: " << max\_grd << " (appeared " << times << " times)\n";

}

return 0;

}

**Task 3: Programming Exercises on Functions**

Write a function that takes as parameter a two-dimensional array and sorts its rows in ascending order and another function that sorts its columns in descending order. Write a program that generates 20 random integers and stores them in a 4×5 array. Then, the program should read an integer and if it is 1, the program should call the first function, otherwise the second one.

A partial solution of the above program has been given below. Some part of it has been left with \_\_\_\_\_\_\_ to be filled by you. You have to write the required codes according to comments written against the lines. **[25 marks]**

Solution:

#include <iostream>

#include <iomanip>

#include <cstdlib>

#include <ctime>

using namespace std;

const int ROWS = 4;

const int COLS = 5;

**//1. Declare prototype of two functions here [5 marks]**

int main()

{

 int i, j, type, arr[ROWS][COLS];

 srand(time(\_\_\_\_\_)); //2. Fill in the blank **[2 marks]**

 for(i = 0; i < ROWS; i++)

 for(j = 0; j < COLS; j++)

// The array must be initialized with random number

 arr[i][j] = \_\_\_\_\_\_\_\_; **// 3. Write your code here [2 marks]**

 cout << "Enter sort type (1: rows): ";

 cin >> type;

 if(type == \_\_\_\_\_\_\_) **// 4. Insert appropriate value [2 marks]**

 sort\_rows(arr);

 else

 sort\_cols(\_\_\_\_\_\_\_);**// 5. Fill the blank [2 marks]**

 for(i = 0; i < ROWS; i++)

 {

 for(j = 0; j < COLS; j++)

 cout << setw(10) << arr[i][j];

 cout << '\n';

 }

 return 0;

}

void sort\_rows(int arr[][COLS])

{

 int i, j, k, temp;

 for(i = 0; i < ROWS; i++)

 {

 for(j = 0; j <\_\_\_\_\_\_\_\_\_\_; j++) // **6. Fill the condition part to make the loop run properly and give the desired output.[2 marks]**

 {

 for(k = j+1; k < COLS; k++)

 {

 if(arr[i][j] > arr[i][k])

 {

**//7. The values must be swapped**

**//Insert swapping code here [5 marks]**

\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_

 }

 }

 }

 }

}

void sort\_cols(int arr[][COLS])

{

 int i, j, k, temp;

 for(j = 0; j < COLS; j++)

 {

 for(i = 0; i < ROWS-1; i++)

 {

 for(k = i+1; k < ROWS; k++)

 {

 if(arr[i][j] < arr[k][j])

 {

**//8. The values must be swapped**

//**Insert swapping code here [5 marks]**

\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_

 }

 }

 }

 }

}

ANSWER:

#include <iostream>

#include <iomanip>

#include <cstdlib>

#include <ctime>

using namespace std;

const int ROWS = 4;

const int COLS = 5;

// Declare prototype of two functions

void sort\_rows(int arr[][COLS]);

void sort\_cols(int arr[][COLS]);

int main()

{

int i, j, type, arr[ROWS][COLS];

srand(time(NULL)); // Seed random number generator

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

{

for (j = 0; j < COLS; j++)

{

arr[i][j] = rand() % 100; // Generate random integer between 0 and 99

cout << setw(5) << arr[i][j]; // Display the generated value

}

cout << endl;

}

cout << "Enter sort type (1: rows, 2: columns): ";

cin >> type;

if (type == 1)

sort\_rows(arr);

else if (type == 2)

sort\_cols(arr);

else

cout << "Invalid choice\n";

cout << "Sorted array:\n";

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

{

for (j = 0; j < COLS; j++)

{

cout << setw(5) << arr[i][j];

}

cout << endl;

}

return 0;

}

void sort\_rows(int arr[][COLS])

{

int temp;

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

{

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

{

for (int k = j + 1; k < COLS; k++)

{

if (arr[i][j] > arr[i][k])

{

temp = arr[i][j];

arr[i][j] = arr[i][k];

arr[i][k] = temp;

}

}

}

}

}

void sort\_cols(int arr[][COLS])

{

int temp;

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

{

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

{

for (int k = i + 1; k < ROWS; k++)

{

if (arr[i][j] < arr[k][j])

{

temp = arr[i][j];

arr[i][j] = arr[k][j];

arr[k][j] = temp;

}

}

}

}

}

Output

