

LAB NO 1

HASNAIN RAZA

24K-0645

BCS-3A

SIR ABDULLAH YAQOOB

[TASK #01 by HASNAIN RAZA – 24K-645]

TASK # 01

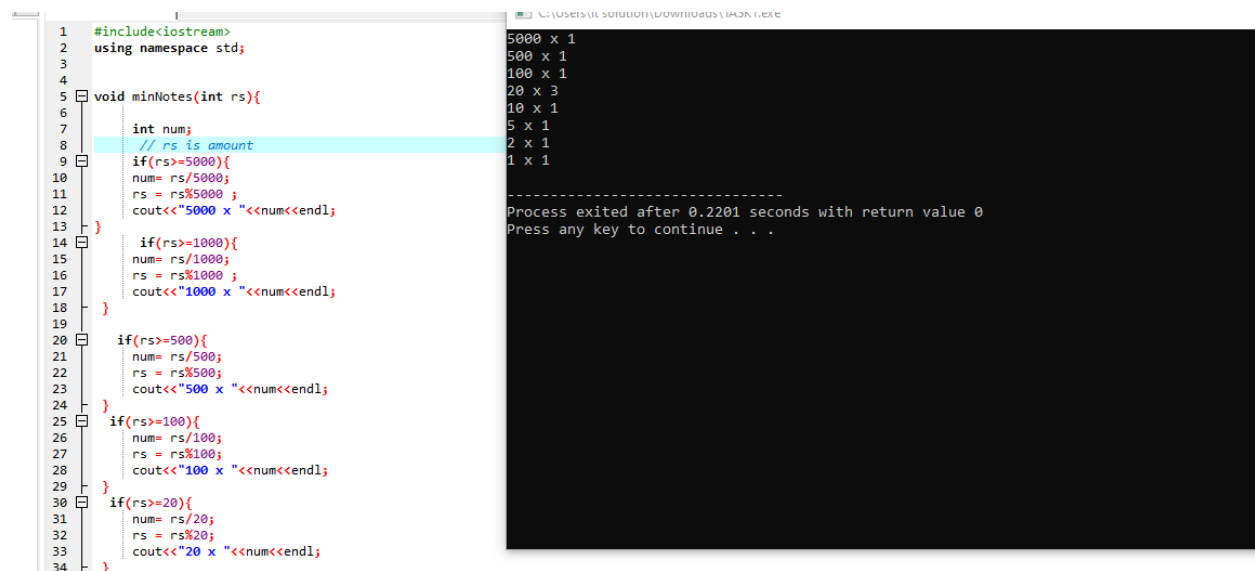
Write a C++ program that accepts an amount from the user and calculates the minimum number of notes required to make that amount using the available denominations (Rs. 5000, 1000, 500, 100, 20, 10, 5, 2, and 1). The program should determine and display the fewest notes needed for the given amount, based on the available denominations.

Input:

A positive integer representing the amount (in Rs.) for which the minimum number of notes needs to be calculated.

Output:

Display the minimum number of notes required, along with the denomination of each note used .



The image shows a C++ program in a code editor on the left and its execution output in a terminal window on the right. The program, named `minNotes`, takes an integer `rs` as input and calculates the minimum number of notes required using denominations of 5000, 1000, 500, 100, 20, 10, 5, 2, and 1. The output displays the count and denomination for each note used, followed by a confirmation message.

```
1 #include<iostream>
2 using namespace std;
3
4 void minNotes(int rs){
5     int num;
6     // rs is amount
7     if(rs>=5000){
8         num= rs/5000;
9         rs = rs%5000 ;
10        cout<<"5000 x "<<num<<endl;
11    }
12    if(rs>=1000){
13        num= rs/1000;
14        rs = rs%1000 ;
15        cout<<"1000 x "<<num<<endl;
16    }
17    if(rs>=500){
18        num= rs/500;
19        rs = rs%500;
20        cout<<"500 x "<<num<<endl;
21    }
22    if(rs>=100){
23        num= rs/100;
24        rs = rs%100;
25        cout<<"100 x "<<num<<endl;
26    }
27    if(rs>=20){
28        num= rs/20;
29        rs = rs%20;
30        cout<<"20 x "<<num<<endl;
31    }
32    if(rs>=10){
33        num= rs/10;
34        rs = rs%10;
35        cout<<"10 x "<<num<<endl;
36    }
37    if(rs>=5){
38        num= rs/5;
39        rs = rs%5;
40        cout<<"5 x "<<num<<endl;
41    }
42    if(rs>=2){
43        num= rs/2;
44        rs = rs%2;
45        cout<<"2 x "<<num<<endl;
46    }
47    if(rs>=1){
48        num= rs/1;
49        rs = rs%1;
50        cout<<"1 x "<<num<<endl;
51    }
52    cout<<"Total number of notes required: "<<num<<endl;
53    cout<<"Press any key to continue . . .";
54    getch();
55 }
```

Output:

```
5000 x 1
500 x 1
100 x 1
20 x 3
10 x 1
5 x 1
2 x 1
1 x 1
-----
Process exited after 0.2201 seconds with return value 0
Press any key to continue . . .
```

```

29 }
30 if(rs>=20){
31     num= rs/20;
32     rs = rs%20;
33     cout<<"20 x "<<num<<endl;
34 }
35 if(rs>=10){
36     num= rs/10;
37     rs = rs%10;
38     cout<<"10 x "<<num<<endl;
39 }
40 if(rs>=5){
41     num= rs/5;
42     rs = rs%5;
43     cout<<"5 x "<<num<<endl;
44 }
45 if(rs>=2){
46     num= rs/2;
47     rs = rs%2;
48     cout<<"2 x "<<num<<endl;
49 }
50 if(rs>=1){
51     num= rs/1;
52     rs = rs%1;
53     cout<<"1 x "<<num<<endl;
54 }
55 };
56
57 int main(){
58     int num;
59     minNotes(5678);
60
61     return 0;
62 }
63
64

```

#include<iostream>

using namespace std;

void minNotes(int rs){

int num;

// rs is amount

if(rs>=5000){

num= rs/5000;

rs = rs%5000 ;

cout<<"5000 x "<<num<<endl;

}

if(rs>=1000){

num= rs/1000;

rs = rs%1000 ;

cout<<"1000 x "<<num<<endl;

```
}
```

```
if(rs>=500){  
    num= rs/500;  
    rs = rs%500;  
    cout<<"500 x "<<num<<endl;
```

```
}
```

```
if(rs>=100){  
    num= rs/100;  
    rs = rs%100;  
    cout<<"100 x "<<num<<endl;
```

```
}
```

```
if(rs>=20){  
    num= rs/20;  
    rs = rs%20;  
    cout<<"20 x "<<num<<endl;
```

```
}
```

```
if(rs>=10){  
    num= rs/10;  
    rs = rs%10;  
    cout<<"10 x "<<num<<endl;
```

```
}
```

```
if(rs>=5){  
    num= rs/5;  
    rs = rs%5;  
    cout<<"5 x "<<num<<endl;
```

```
}
```

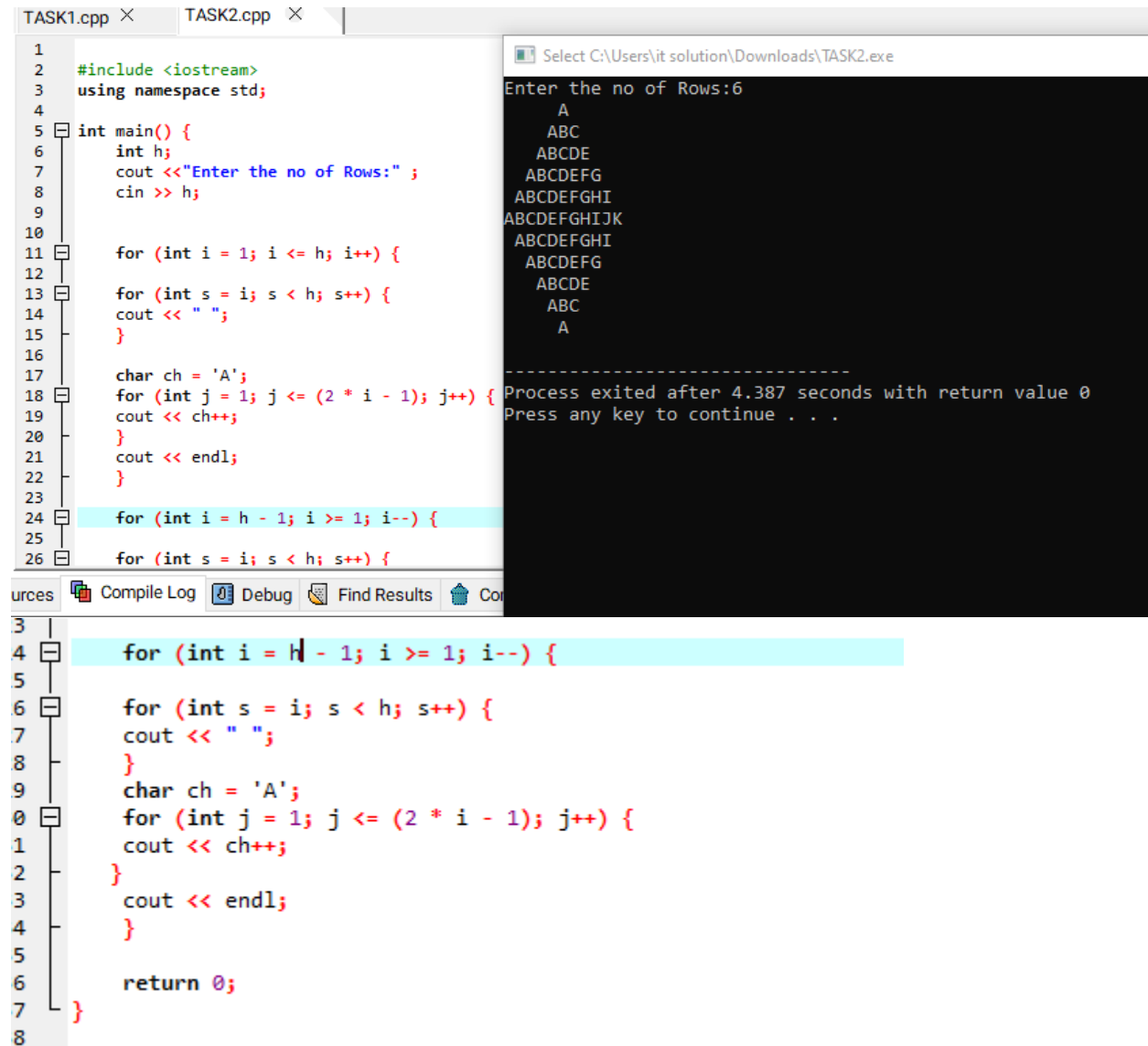
```
if(rs>=2){  
    num= rs/2;  
    rs = rs%2;  
    cout<<"2 x "<<num<<endl;
```

```
}  
  
if(rs>=1){  
    num= rs/1;  
    rs = rs%1;  
    cout<<"1 x "<<num<<endl;  
}  
};
```

```
int main(){  
    int num;  
    minNotes(5678);  
  
    return 0;  
  
};
```

[TASK #02 by HASNAIN RAZA – 24K-645]

TASK #02: Write a C++ program to input a number from user and print a diamond pattern of alphabet characters according to input.



The image shows a C++ IDE with two tabs: TASK1.cpp and TASK2.cpp. The TASK2.cpp tab is active, displaying the following code:

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int h;
6     cout << "Enter the no of Rows:" ;
7     cin >> h;
8
9     for (int i = 1; i <= h; i++) {
10
11         for (int s = i; s < h; s++) {
12             cout << " ";
13         }
14
15         char ch = 'A';
16         for (int j = 1; j <= (2 * i - 1); j++) {
17             cout << ch++;
18         }
19         cout << endl;
20
21         for (int i = h - 1; i >= 1; i--) {
22
23             for (int s = i; s < h; s++) {
24                 cout << " ";
25             }
26
27             char ch = 'A';
28             for (int j = 1; j <= (2 * i - 1); j++) {
29                 cout << ch++;
30             }
31             cout << endl;
32         }
33
34         return 0;
35     }
```

The output window shows the execution of the program. It prompts the user to enter the number of rows (6) and displays the resulting diamond pattern of alphabet characters:

```
Enter the no of Rows:6
A
 ABC
  ABCDE
   ABCDEFG
  ABCDEFGHI
 ABCDEFGHIJK
  ABCDEFGHI
   ABCDEFG
    ABCDE
     ABC
      A
-----
Process exited after 4.387 seconds with return value 0
Press any key to continue . . .
```

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int h;
```

```
    cout << "Enter the no of Rows:" ;
```

```
    cin >> h;
```

```
    for (int i = 1; i <= h; i++) {
```

```
        for (int s = i; s < h; s++) {
```

```
            cout << " ";
```

```
        }
```

```
        char ch = 'A';
```

```
        for (int j = 1; j <= (2 * i - 1); j++) {
```

```
            cout << ch++;
```

```
        }
```

```
        cout << endl;
```

```
    }
```

```
    for (int i = h - 1; i >= 1; i--) {
```

```
        for (int s = i; s < h; s++) {
```

```
            cout << " ";
```

```
        }
```

```
        char ch = 'A';
```

```
        for (int j = 1; j <= (2 * i - 1); j++) {
```

```
            cout << ch++;
```

```
        }
```

```
cout << endl;
```

```
}
```

```
return 0;
```

```
}
```


[TASK #03 by HASNAIN RAZA – 24K-645]

TASK #03: Write a C++ program that defines a function named `findPrimesInRange`, which takes two numbers

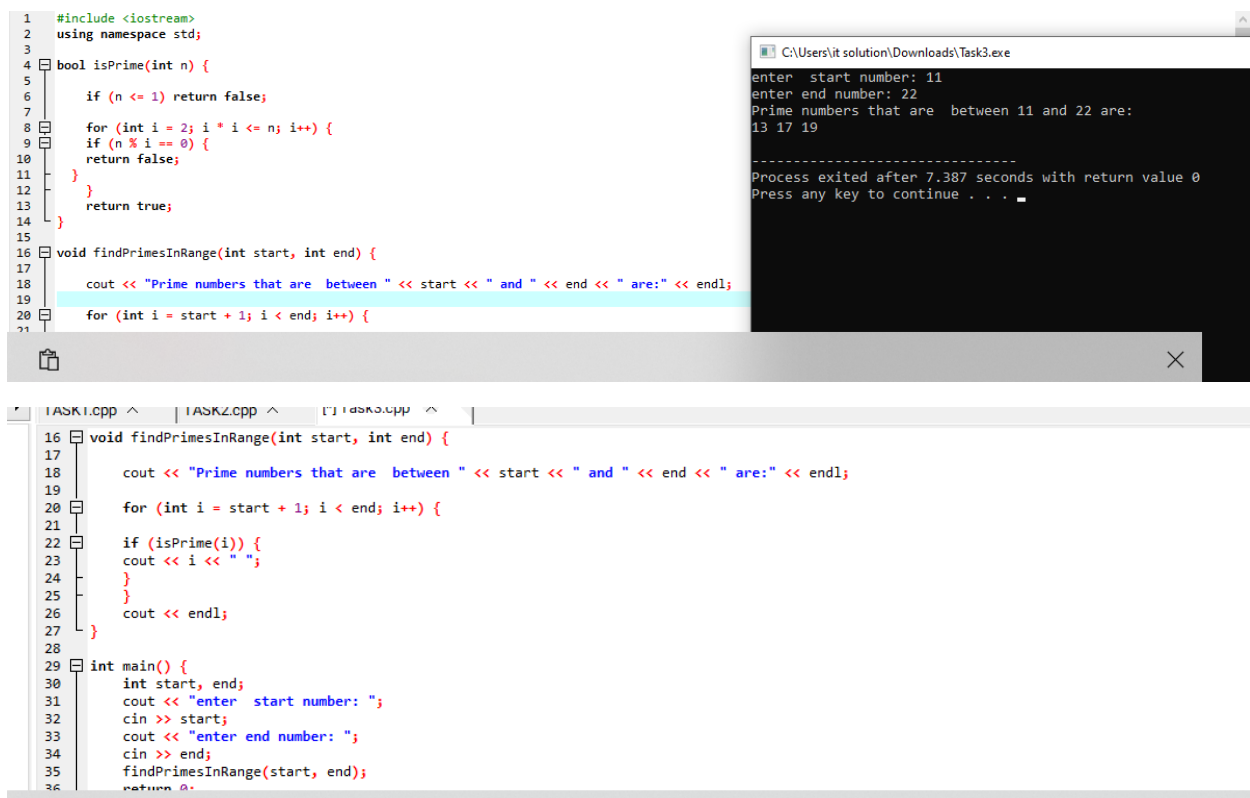
as input and calculates all the prime numbers between them (exclusive). The program should print all the prime numbers found within the given range.

Input:

Two integers, start and end, where the program will calculate and display all prime numbers between start and end (excluding start and end).

Output:

A list of prime numbers between the given start and end.



The image shows a C++ program in a code editor and its execution output in a terminal window. The code defines a function `isPrime` to check if a number is prime, and a function `findPrimesInRange` to find all prime numbers between two given numbers (exclusive). The `main` function takes input for start and end numbers and calls `findPrimesInRange`.

```
1 #include <iostream>
2 using namespace std;
3
4 bool isPrime(int n) {
5     if (n <= 1) return false;
6     for (int i = 2; i * i <= n; i++) {
7         if (n % i == 0) {
8             return false;
9         }
10    }
11    return true;
12 }
13
14 void findPrimesInRange(int start, int end) {
15     cout << "Prime numbers that are between " << start << " and " << end << " are:" << endl;
16     for (int i = start + 1; i < end; i++) {
17         if (isPrime(i)) {
18             cout << i << " ";
19         }
20     }
21     cout << endl;
22 }
23
24 int main() {
25     int start, end;
26     cout << "enter start number: ";
27     cin >> start;
28     cout << "enter end number: ";
29     cin >> end;
30     findPrimesInRange(start, end);
31     return 0;
32 }
```

The terminal output shows the program running with start number 11 and end number 22. It displays the prime numbers 13, 17, and 19. The process exited after 7.387 seconds with return value 0.

```
C:\Users\it solution\Downloads\Task3.exe
enter start number: 11
enter end number: 22
Prime numbers that are between 11 and 22 are:
13 17 19

-----
Process exited after 7.387 seconds with return value 0
Press any key to continue . . .
```

[TASK #04 by HASNAIN RAZA – 24K-645]

TASK #04: Write a C++ program that performs the following tasks:

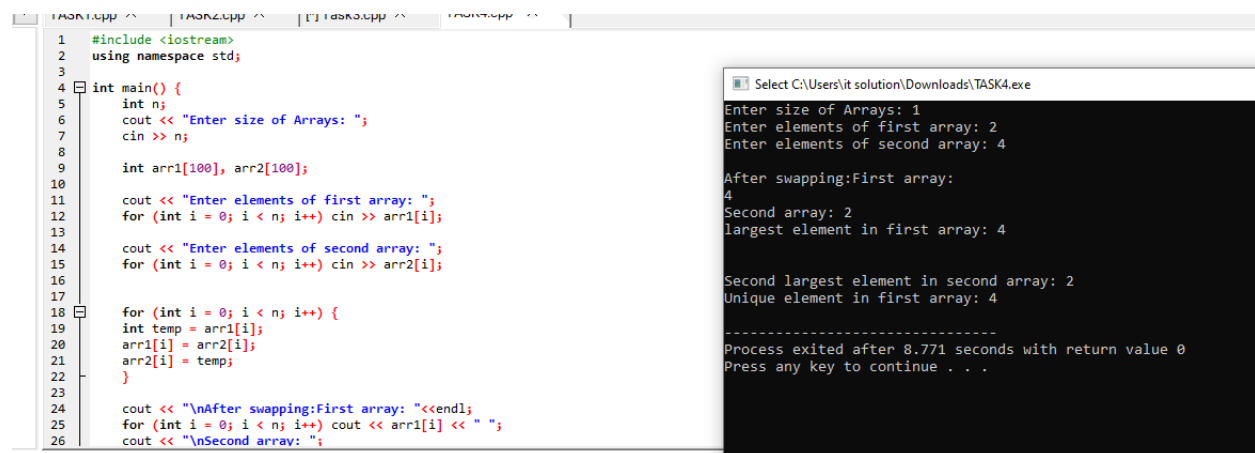
1. Asks the user to input size and values for two arrays.
2. Swaps the values of both arrays.
3. Finds the largest element from the first array (after swapping).
4. Finds the second largest element from the second array (after swapping).
5. Finds the unique element (element that appears only once) in the first array (after swapping) and if all values in the array are unique, return the value at index 0.

Input:

- Two arrays of integers (1D) provided by the user. The program ensures that both arrays have the same size.

Output:

- The new swapped arrays.
- The largest element from the first swapped array.
- The second largest element from the second swapped array.
- The unique element in the first swapped array.



The screenshot shows a C++ program in a code editor and its execution output in a console window. The program defines two arrays, arr1 and arr2, of size 100. It prompts the user to enter the size of the arrays (1) and then the elements of each array (2 for arr1, 4 for arr2). After swapping the arrays, it displays the new contents: arr1 contains [4] and arr2 contains [2]. It then finds the largest element in arr1 (4) and the second largest element in arr2 (2). Finally, it identifies the unique element in arr1 (4) and prints the result. The console output matches the program's logic and the task requirements.

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n;
6     cout << "Enter size of Arrays: ";
7     cin >> n;
8
9     int arr1[100], arr2[100];
10
11     cout << "Enter elements of first array: ";
12     for (int i = 0; i < n; i++) cin >> arr1[i];
13
14     cout << "Enter elements of second array: ";
15     for (int i = 0; i < n; i++) cin >> arr2[i];
16
17     for (int i = 0; i < n; i++) {
18         int temp = arr1[i];
19         arr1[i] = arr2[i];
20         arr2[i] = temp;
21     }
22
23     cout << "\nAfter swapping:First array: ";
24     for (int i = 0; i < n; i++) cout << arr1[i] << " ";
25     cout << "\nSecond array: ";
26     for (int i = 0; i < n; i++) cout << arr2[i] << " ";
27 }
```

Output:

```
Select C:\Users\it solution\Downloads\TASK4.exe
Enter size of Arrays: 1
Enter elements of first array: 2
Enter elements of second array: 4

After swapping:First array:
4
Second array: 2
largest element in first array: 4

Second largest element in second array: 2
Unique element in first array: 4

-----
Process exited after 8.771 seconds with return value 0
Press any key to continue . . .
```

```

    cout << "\nSecond array: ";
    for (int i = 0; i < n; i++) cout << arr2[i] << " ";

    int largest = arr1[0];
    for (int i = 1; i < n; i++) {
        if (arr1[i] > largest) largest = arr1[i];
    }

    int first = arr2[0], second = -1;
    for (int i = 1; i < n; i++) {
        if (arr2[i] > first) {
            second = first;
            first = arr2[i];
        } else if (arr2[i] > second && arr2[i] != first) {
            second = arr2[i];
        }
    }

    if (second == -1) second = first;

43     }
44
45     if (second == -1) second = first;
46
47
48     int unique = arr1[0];
49     bool found = false;
50     for (int i = 0; i < n; i++) {
51         int count = 0;
52         for (int j = 0; j < n; j++) {
53             if (arr1[i] == arr1[j]) count++;
54         }
55         if (count == 1) {
56             unique = arr1[i];
57             found = true;
58             break;
59         }
60     }
61     if (!found) unique = arr1[0];
62
63     cout << "\nlargest element in first array: " << largest<<endl;
64     cout << "\n\nSecond largest element in second array: " << second<<endl;
65     cout << "Unique element in first array: " << unique << endl;
66
67     return 0;
68 }

```

#include <iostream>

using namespace std;

int main() {

int n;

cout << "Enter size of Arrays: ";

cin >> n;

int arr1[100], arr2[100];

```
cout << "Enter elements of first array: ";
```

```
for (int i = 0; i < n; i++) cin >> arr1[i];
```

```
cout << "Enter elements of second array: ";
```

```
for (int i = 0; i < n; i++) cin >> arr2[i];
```

```
for (int i = 0; i < n; i++) {
```

```
int temp = arr1[i];
```

```
arr1[i] = arr2[i];
```

```
arr2[i] = temp;
```

```
}
```

```
cout << "\nAfter swapping:First array: "<<endl;
```

```
for (int i = 0; i < n; i++) cout << arr1[i] << " ";
```

```
cout << "\nSecond array: ";
```

```
for (int i = 0; i < n; i++) cout << arr2[i] << " ";
```

```
int largest = arr1[0];
```

```
for (int i = 1; i < n; i++) {
```

```
if (arr1[i] > largest) largest = arr1[i];
```

```
}
```

```
int first = arr2[0], second = -1;
```

```
for (int i = 1; i < n; i++) {
```

```

    if (arr2[i] > first) {
        second = first;
        first = arr2[i];
    } else if (arr2[i] > second && arr2[i] != first) {
        second = arr2[i];
    }
}

if (second == -1) second = first;

```

```

int unique = arr1[0];
bool found = false;
for (int i = 0; i < n; i++) {
    int count = 0;
    for (int j = 0; j < n; j++) {
        if (arr1[i] == arr1[j]) count++;
    }
    if (count == 1) {
        unique = arr1[i];
        found = true;
        break;
    }
}

if (!found) unique = arr1[0];

```

```

cout << "\n\nlargest element in first array: " << largest<<endl;

cout << "\n\nSecond largest element in second array: " << second<<endl;

cout << "Unique element in first array: " << unique << endl;

```

```
    return 0;
}
```

[TASK #05 by HASNAIN RAZA – 24K-645]

TASK #05:

Write a C++ program to multiply two matrices using a function named multiplyMatrices. The program should take two 2-dimensional arrays (matrices) as input, validate the multiplication rule, and calculate the product of the matrices. Use the following instructions:

Rules for Matrix Multiplication:

1. The number of columns in the first matrix must equal the number of rows in the second matrix.

Input:

1. Dimensions of the first matrix A (m rows and n columns).
2. Dimensions of the second matrix B (n rows and p columns).
3. Elements of both matrices A and B.

Output:

1. The resulting matrix C after multiplication.

```
1  #include <iostream>
2  using namespace std;
3
4  #include <iostream>
5  using namespace std;
6
7  int main() {
8      int s, n, n4, p;
9      int A[30][30], B[30][30], C[30][30];
10
11      cout << "Enter rows and columns of first matrix: ";
12      cin >> s >> n;
13      cout << "Enter rows and columns of second matrix: ";
14      cin >> n4 >> p;
15
16      if (n != n4) {
17          cout << "Matrix multiplication not possible!!!";
18          return 0;
19      }
20
21      cout << "Enter first matrix:\n";
```

```
Select C:\Users\it solution\Downloads\TASK5.exe
Enter rows and columns of first matrix: 2 3
Enter rows and columns of second matrix: 3 2
Enter first matrix:
3 5 7
7 6 5
Enter second matrix:
3 7 8
6 3 2
Resultant Matrix!!!:
70 65
84 95

-----
Process exited after 22.06 seconds with return value 0
Press any key to continue . . .
```

```

26     cout << "Enter second matrix:\n";
27     for (int i = 0; i < n4; i++)
28     for (int j = 0; j < p; j++)
29     cin >> B[i][j];
30
31
32     for (int i = 0; i < s; i++) {
33     for (int j = 0; j < p; j++) {
34     C[i][j] = 0;
35     for (int k = 0; k < n; k++) {
36     C[i][j] += A[i][k] * B[k][j];
37     }
38     }
39     }
40
41
42     cout << "Resultant Matrix!!!:\n";
43     for (int i = 0; i < s; i++) {
44     for (int j = 0; j < p; j++)
45     cout << C[i][j] << " ";
46     cout << endl;
47     }
48
49     return 0;
50 }
51

```

```
#include <iostream>
```

```
using namespace std;
```

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int s, n, n4, p;
```

```
    int A[30][30], B[30][30], C[30][30];
```

```
    cout << "Enter rows and columns of first matrix: ";
```

```
    cin >> s >> n;
```

```
    cout << "Enter rows and columns of second matrix: ";
```

```
cin >> n4 >> p;
```

```
if (n != n4) {
```

```
cout << "Matrix multiplication not possible!!!";
```

```
return 0;
```

```
}
```

```
cout << "Enter first matrix:\n";
```

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

```
for (int j = 0; j < n; j++)
```

```
cin >> A[i][j];
```

```
cout << "Enter second matrix:\n";
```

```
for (int i = 0; i < n4; i++)
```

```
for (int j = 0; j < p; j++)
```

```
cin >> B[i][j];
```

```
for (int i = 0; i < s; i++) {
```

```
for (int j = 0; j < p; j++) {
```

```
C[i][j] = 0;
```

```
for (int k = 0; k < n; k++) {
```

```
C[i][j] += A[i][k] * B[k][j];
```

```
}
```

```
}
```

```
}
```



```
cout << "Resultant Matrix!!!:\n";  
  
for (int i = 0; i < s; i++) {  
    for (int j = 0; j < p; j++)  
        cout << C[i][j] << " ";  
    cout << endl;  
}  
  
return 0;  
}
```