```cpp
#include <iostream>
#include <iomanip>
#include <cstring>
using namespace std ;

const int NUM_MEDAL = 3 ;
const int NUM_COUNTRY = 4 ;


void readMedal(int [][NUM_MEDAL] );
int totalMedals (int [][NUM_MEDAL]);
int largestNum (int [][NUM_MEDAL]);
int smallestNum (int [][NUM_MEDAL]);
int highestGold (int [][NUM_MEDAL]);
int totalBronze (int [][NUM_MEDAL]);

int main()
{
    int medals[NUM_COUNTRY][NUM_MEDAL];
    readMedal(medals);

    cout << "----------------------------------------" << endl;
    cout << "Total medals won by country 3 : " << totalMedals(medals) << endl;
    cout << "Largest number of medals won : " << largestNum(medals) << endl;
    cout << "Smallest number of medals won : " << smallestNum(medals) << endl;
    cout << "Highest number of gold medals won : " << highestGold(medals) << endl;
    cout << "Total number of bronze medals won : " << totalBronze(medals) << endl;

    return 0;

}
```

```cpp
int main()

}

void readMedal(int medals[][NUM_MEDAL] )
{
    cout << "Enter number of medals for each country : " << endl;
    for (int a=0 ; a < NUM_COUNTRY ; a++)
    {
        cout << "----------------------------------------" << endl;
        cout << "Country " << (a + 1) << " : " << endl;
        cout << "Enter number of gold medals: ";
        cin >> medals[a][0];
        cout << "Enter number of silver medals: ";
        cin >> medals[a][1];
        cout << "Enter number of bronze medals: ";
        cin >> medals[a][2];
        cout << "----------------------------------------" << endl;
    }
}

int totalMedals (int medals[][NUM_MEDAL])
{
    int total = 0;
    for (int b = 0 ; b < NUM_MEDAL ; b++)
    {
        total += medals[2][b];
    }
    return total;
}

int largestNum (int medals [][NUM_MEDAL])
```
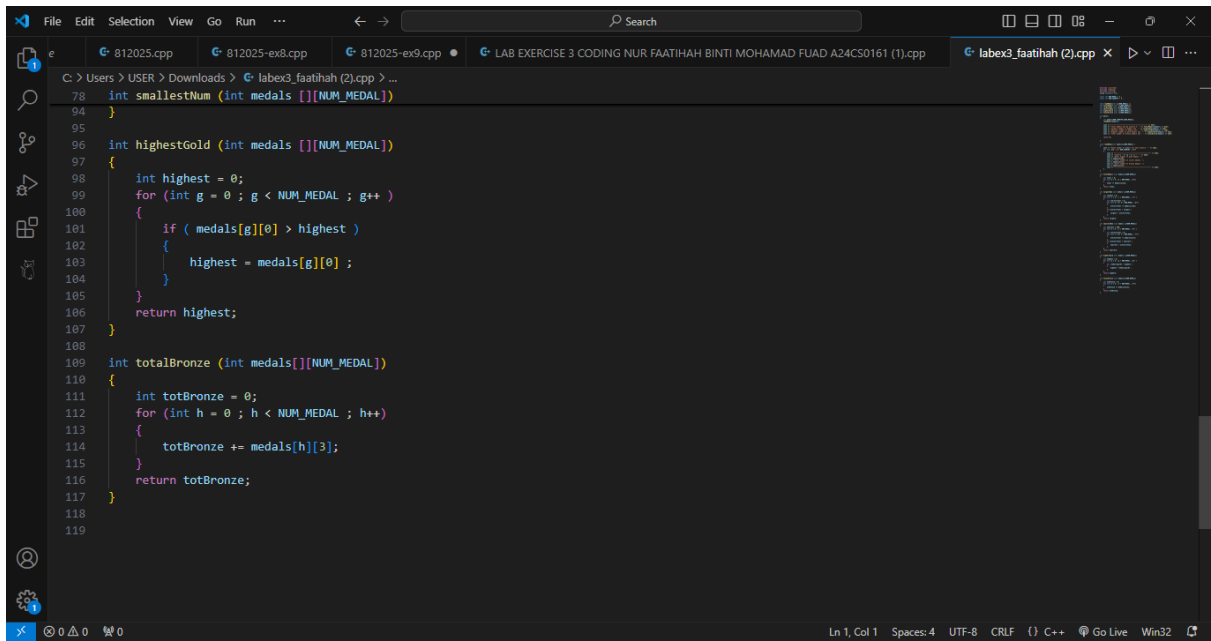
File Edit Selection View Go Run ...

812025.cpp   812025-ex8.cpp   812025-ex9.cpp   LAB EXERCISE 3 CODING NUR FAATIHAH BINTI MOHAMAD FUAD A24CS0161 (1).cpp   labex3_faatihah (2).cpp

C: > Users > USER > Downloads > labex3_faatihah (2).cpp > ...

```cpp
int largestNum (int medals [][NUM_MEDAL])
{
    int largest = 0;
    for (int c = 0 ; c < NUM_MEDAL ; c++ )
    {
        int overallTotal = 0;
        for (int d = 0; d < NUM_MEDAL ; d++)
        {
            overallTotal += medals[c][d];
        }
        if (overallTotal > largest )
        {
            largest = overallTotal;
        }
    }
    return largest;
}

int smallestNum (int medals [][NUM_MEDAL])
{
    int smallest = 999;
    for (int e = 0 ; e < NUM_MEDAL ; e++ )
    {
        int overallTotal = 0;
        for (int f = 0; f < NUM_MEDAL ; f++)
        {
            overallTotal += medals[e][f];
        }
        if (overallTotal < smallest )
        {
            smallest = overallTotal;
```

```cpp
int smallestNum (int medals [][NUM_MEDAL])
{
    int smallest = 999;
    for (int e = 0 ; e < NUM_MEDAL ; e++ )
    {
        int overallTotal = 0;
        for (int f = 0; f < NUM_MEDAL ; f++)
        {
            overallTotal += medals[e][f];
        }
        if (overallTotal < smallest )
        {
            smallest = overallTotal;
        }
    }
    return smallest;
}

int highestGold (int medals [][NUM_MEDAL])
{
    int highest = 0;
    for (int g = 0 ; g < NUM_MEDAL ; g++ )
    {
        if ( medals[g][0] > highest )
        {
            highest = medals[g][0] ;
        }
    }
    return highest;
}
```

```cpp
int smallestNum (int medals [][NUM_MEDAL])
}

int highestGold (int medals [][NUM_MEDAL])
{
    int highest = 0;
    for (int g = 0 ; g < NUM_MEDAL ; g++ )
    {
        if ( medals[g][0] > highest )
        {
            highest = medals[g][0] ;
        }
    }
    return highest;
}

int totalBronze (int medals[][NUM_MEDAL])
{
    int totBronze = 0;
    for (int h = 0 ; h < NUM_MEDAL ; h++)
    {
        totBronze += medals[h][3];
    }
    return totBronze;
}
```