***NAME : Himanshu Dixit***

***ENROLL NO. : 21103262***

***BATCH : B10***

***Software Development Lab – II [15B17CI271]***

***Assignment Sheet***

***Week 1***

***Q1.*** *WAP using C++ to read three numbers from user and display their sum and average on  output screen.*

*Sample input:*

*Enter any three numbers: 3 5 7*

*Sample output:*

*Sum is = 15*

*Average is = 5*

***Solution :***

#include <iostream>

using namespace std;

int main()

{

int a,b,c,sum;

float avg;

cout<<"Enter any three numbers :";

cin>>a>>b>>c;

sum = a+b+c;

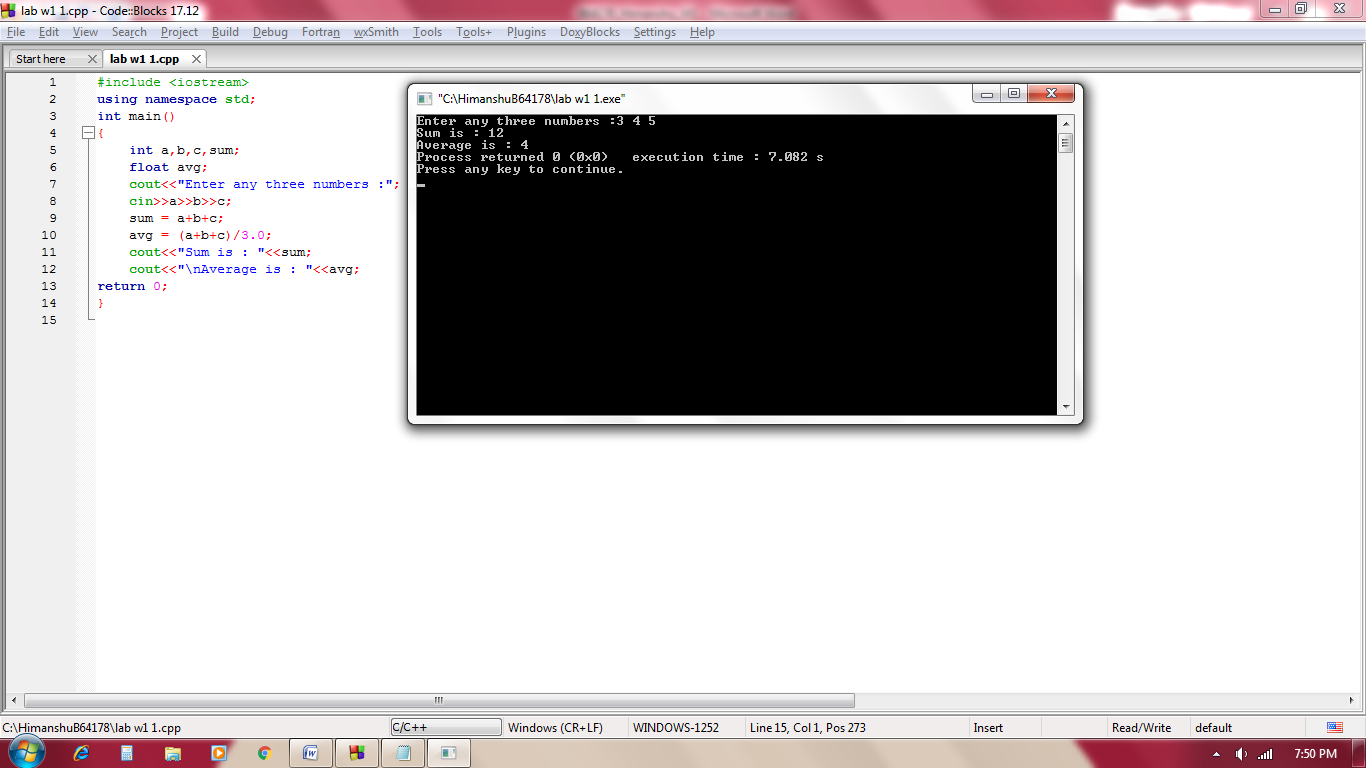
avg = (a+b+c)/3.0;

cout<<"Sum is : "<<sum;

cout<<"\nAverage is : "<<avg;

return 0;

}



***Q2.*** *Write a C++ program to read two numbers from the user and display the larger value  on the output screen.*

*Sample input: Enter any two numbers: 27 60*

*Sample output: larger number is = 60*

***Solution :***

#include <iostream>

using namespace std;

int main()

{

int a,b;

cout<<"Enter two numbers :";

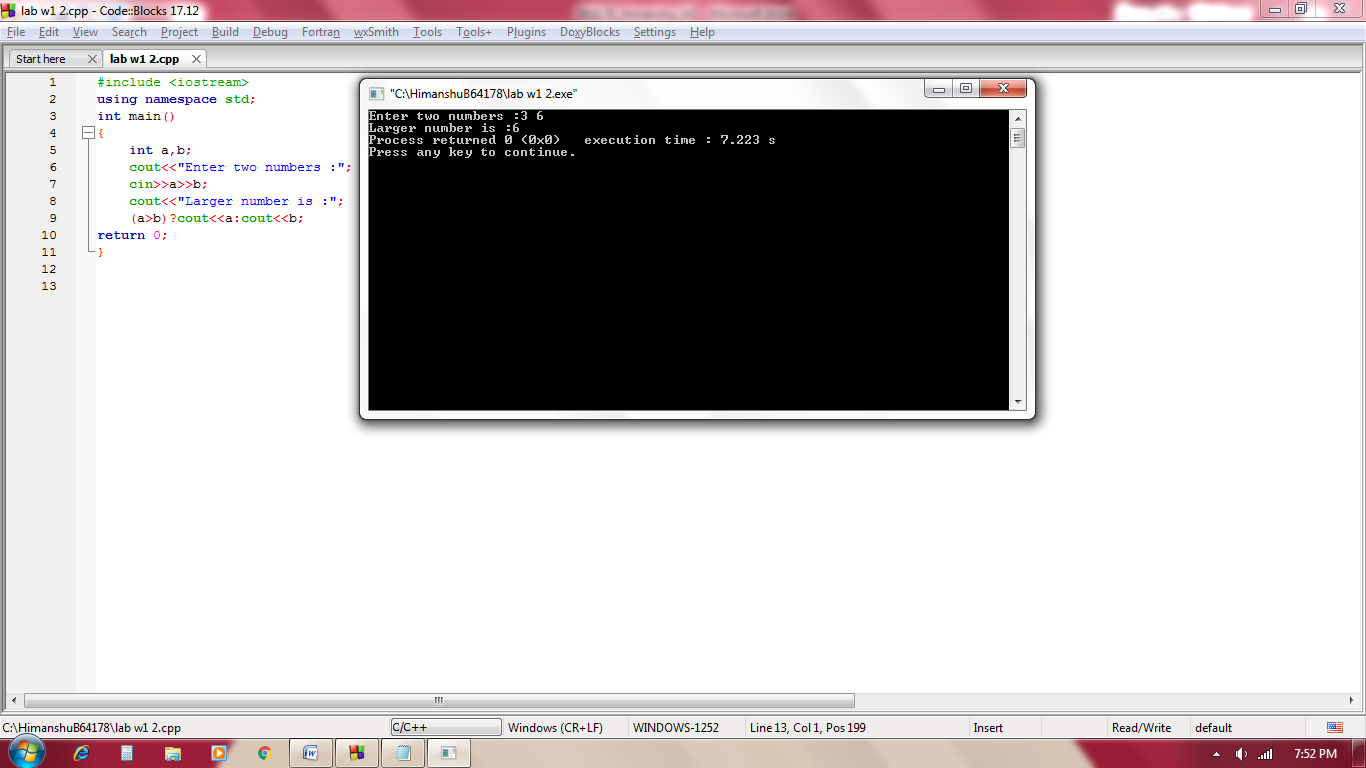
cin>>a>>b;

cout<<"Larger number is :";

(a>b)?cout<<a:cout<<b;

return 0;

}



***Q3.*** *Write a C++ program to read the values of variables: a, b c and d from user and display  the value of x on output screen, where*

*x = a / (b -c) + d*

*Sample input1: Enter the value of a,b,c& d :*

*250 85 25 2*

*Sample output1: x=a/(b-c) +d = 6.16667*

*Sample input2: Enter the value of a,b,c& d :*

*300 60 60 5*

*Sample output2: x= Cannot divide by zero Error. Retry with different input*

***Solution :***

#include <iostream>

using namespace std;

int main()

{

float a,b,c,d,x;

cout<<"Enter the value of a,b,c,d :";

cin>>a>>b>>c>>d;

x=a/(b-c)+d;

if(b==c)

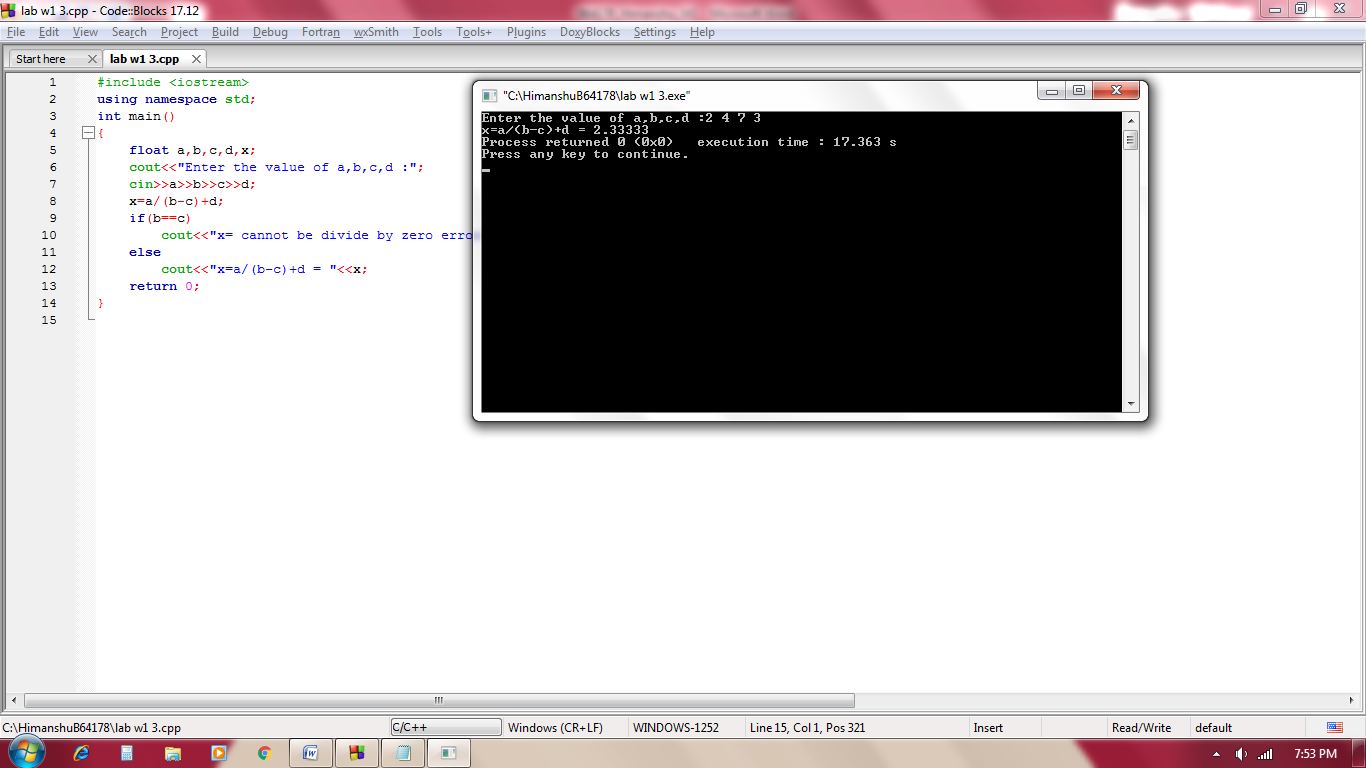
cout<<"x= cannot be divide by zero error. retry with different input";

else

cout<<"x=a/(b-c)+d = "<<x;

return 0;

}



***Q4.*** *Given an unsorted array with both positive and negative elements. Find the smallest  positive number missing from the array in O(n) time using constant extra space. It is  allowed to modify the original array.*

*Examples:*

*Input: {2, 3, 7, 6, 8, -1, -10, 15}*

*Output: 1*

*Input: { 2, 3, -7, 6, 8, 1, -10, 15 }*

*Output: 4*

*Input: {1, 1, 0, -1, -2}*

*Output: 2*

***Solution :***

#include <iostream>

using namespace std;

int main()

{

int a[100],n,k=1,min,l;

cout<<"Enter the size : ";

cin>>n;

cout<<"Enter the elements : ";

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

cin>>a[i];

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

{

for(int j=i;j<n;j++)

{

if(a[i]>a[j])

{

int temp=a[j];

a[j]=a[i];

a[i]=temp;

}

}

}

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

{

if(a[i]>0)

{

l=i;

break;

}

}

if(a[l]!=1)

cout<<"1";

else

{

for(int i=l; i<n; i++)

{

if(a[i]!=k)

{

cout<<k;

break;

}

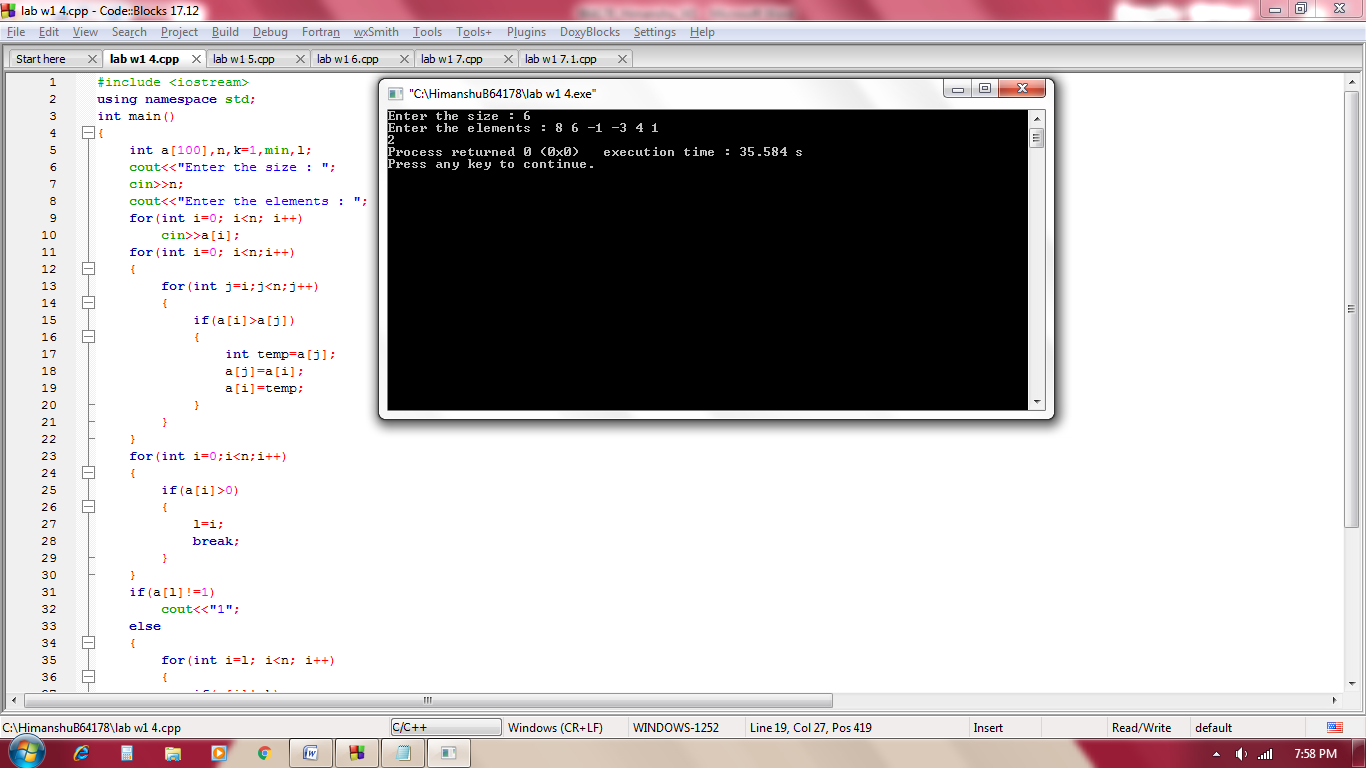
k++;

}

}

return 0;

}



***Q5.*** *Smallest prime number missing in an array: Given an array containing n distinct  numbers. The task is to find the smallest prime which is not present in the array.*

*Note: If there is no prime number missing up to the maximum element of the array then  print “No prime number missing”.*

*Examples:*

*Input:arr[] = {9, 11, 4, 2, 3, 7, 0, 1}*

*Output: 5*

*5 is the smallest prime, which is not present in array.*

*Input:arr[] = {3, 0, 2, 5}*

*Output: No prime number missing*

*As 5 is the maximum element and all prime numbers upto 5*

*are present in the array.*

***Solution :***

#include <iostream>

using namespace std;

int main()

{

int a[100],n,max=0,i=2;

cout<<"Enter the size : ";

cin>>n;

cout<<"Enter the elements : ";

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

{

cin>>a[i];

if(max<a[i])

max=a[i];

}

while(i<=max)

{

int flag=0;

for(int k=2; k<=i/2; k++)

{

if(i%k==0)

{

flag=1;

break;

}

}

if(flag==0)

{

int t=1;

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

{

if(i==a[j])

{

t=0;

break;

}

}

if(t==1)

{

cout<<i;

break;

}

}

i++;

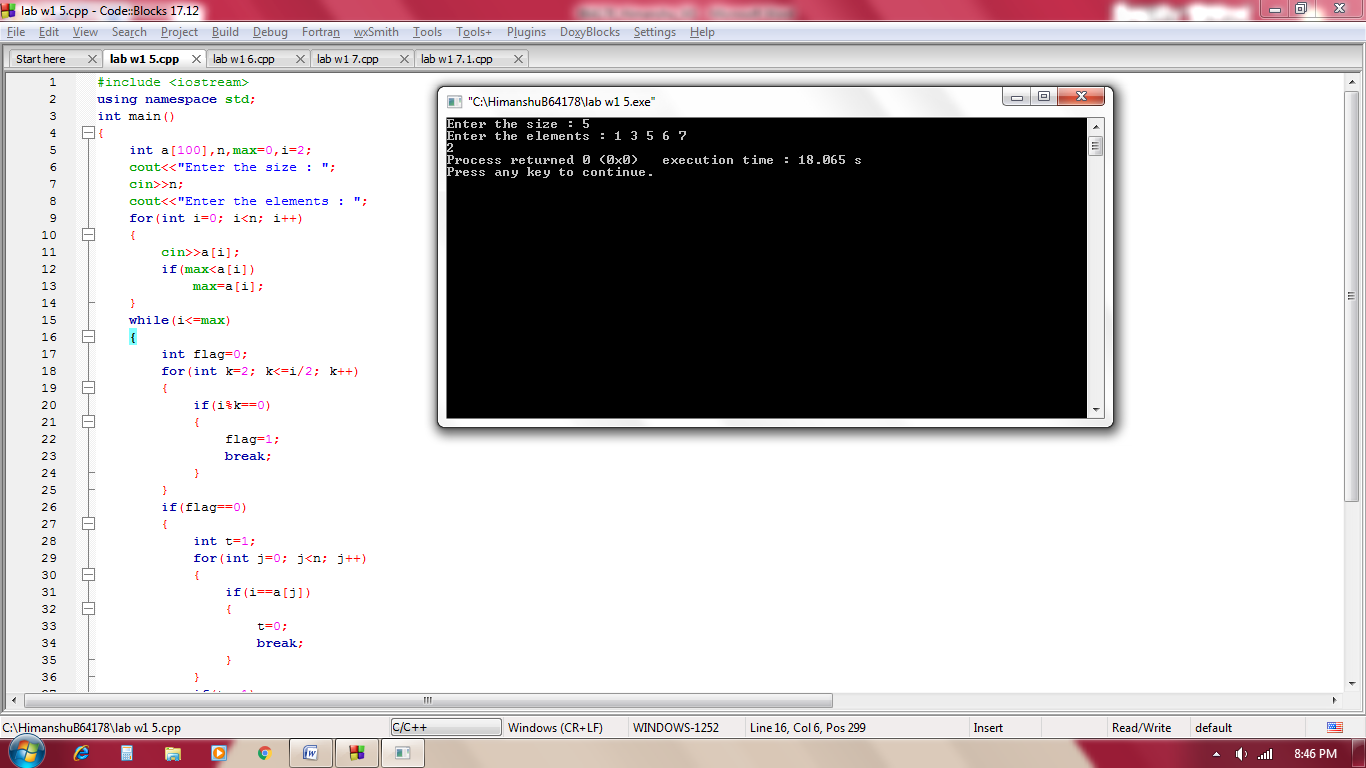
}

if(i>max)

cout<<"no prime number missing";

return 0;

}



***Q6.*** *Given a boolean matrix mat[M][N] of size M X N, modify it such that if a matrix cell  mat[i][j] is 1 (or true) then make all the cells of ith row and jth column as 1.*

*Example 1*

*The matrix*

*1 0*

*0 0*

*should be changed to following*

*1 1*

*1 0*

*Example 2*

*The matrix*

*0 0 0*

*0 0 1*

*should be changed to following*

*0 0 1*

*1 1 1*

*Example 3*

*The matrix*

*1 0 0 1*

*0 0 1 0*

*0 0 0 0*

*should be changed to following*

*1 1 1 1*

*1 1 1 1*

*1 0 1 1*

***Solution :***

#include <iostream>

using namespace std;

int main()

{

int a[100][100],b[100][100],r,c;

cout<<"Enter the row and coloumn of matrix : ";

cin>>r>>c;

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

{

for(int j=0; j<c; j++){

cin>>a[i][j];

b[i][j]=a[i][j];

}

}

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

{

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

{

if(b[i][j]==1)

{

for(int k=0; k<r; k++)

a[k][j]=1;

for(int k=0; k<c; k++)

a[i][k]=1;

}

}

}

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

{

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

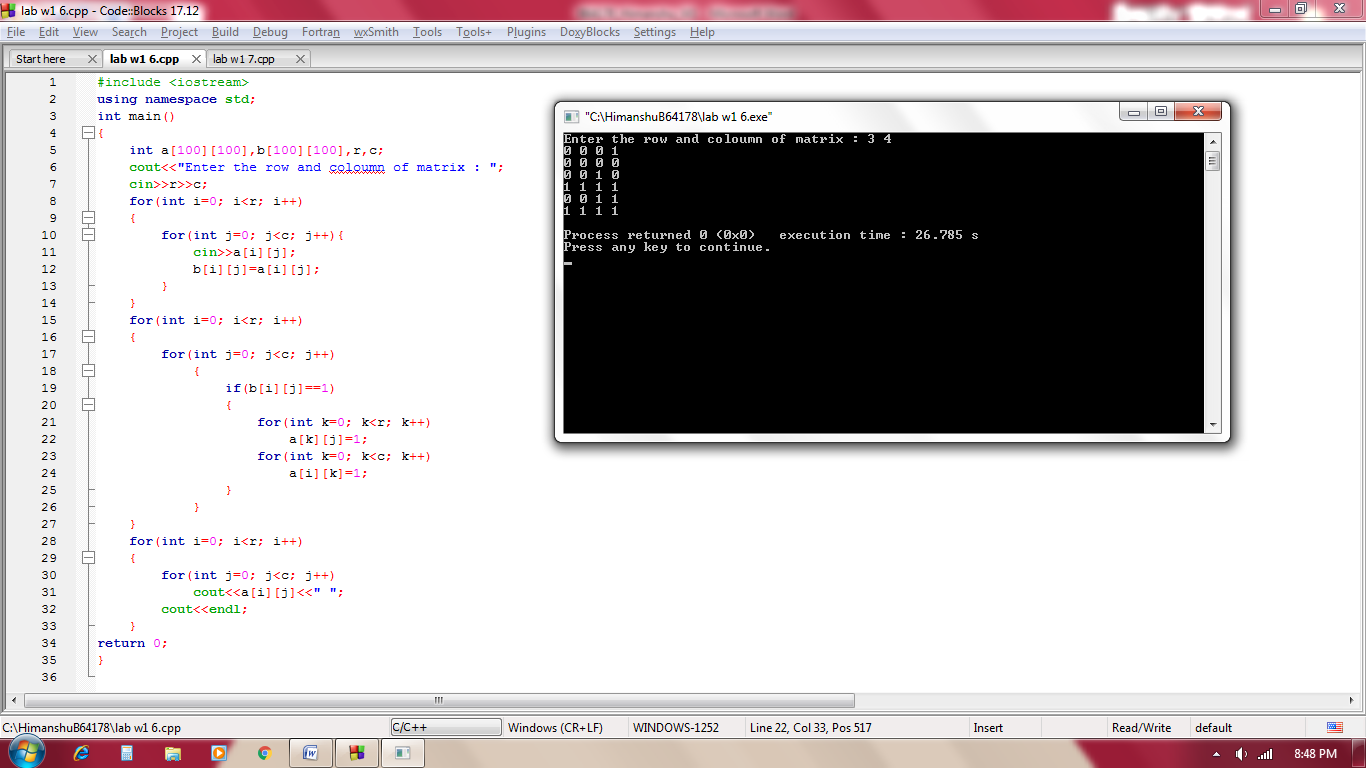
cout<<a[i][j]<<" ";

cout<<endl;

}

return 0;

}



***Q7.*** *C++ Program to assign data to members of a structure variable and display it. Example:*

*Enter Full name: Magdalena Dankova*

*Enter age: 27*

*Enter salary: 1024.4*

*Displaying Information.*

*Name: Magdalena Dankova*

*Age: 27*

*Salary: 1024.4*

***Solution :***

#include<iostream>

#include<stdio.h>

using namespace std;

struct employee

{

char name[100];

int age;

float salary;

} emp;

int main()

{

cout<<"Enter Full name:";

gets(emp.name);

cout<<"Enter age:";

cin>>emp.age;

cout<<"Enter salary:";

cin>>emp.salary;

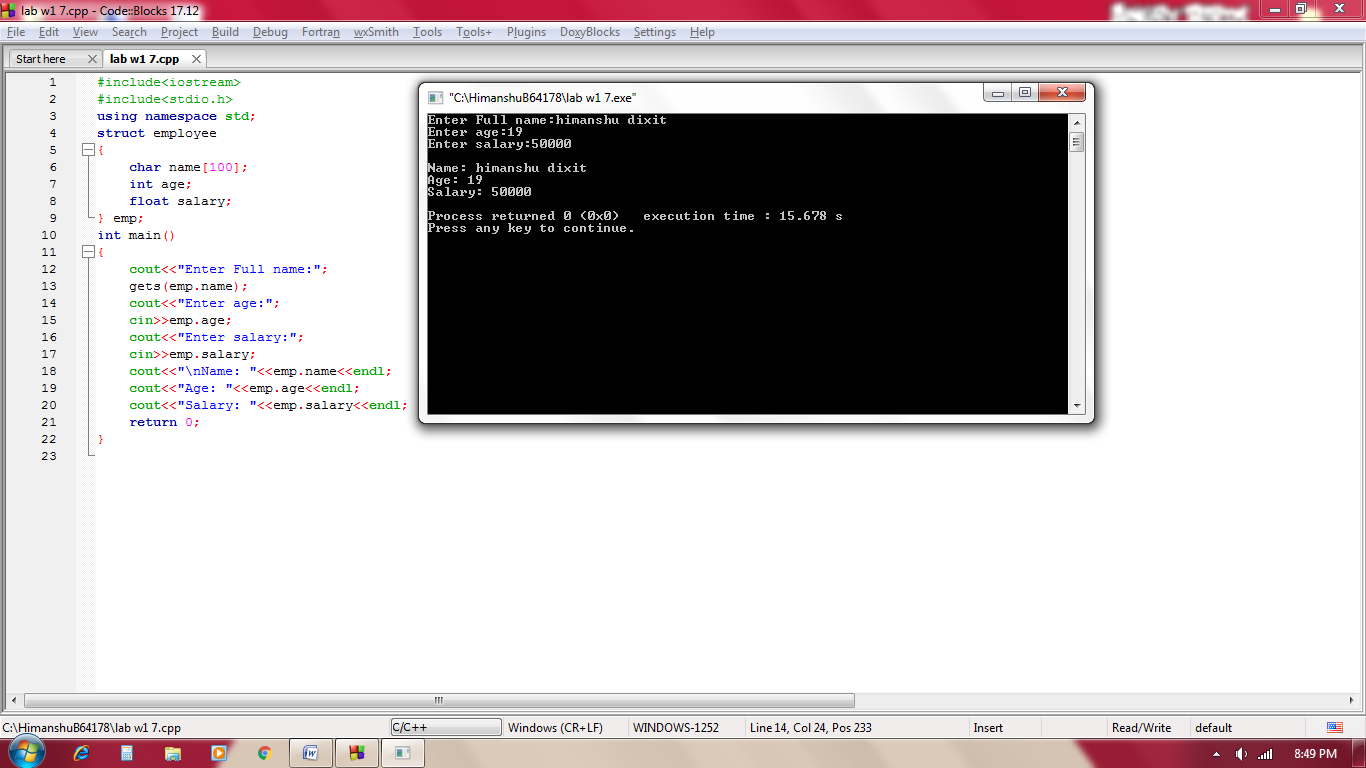
cout<<"\nName: "<<emp.name<<endl;

cout<<"Age: "<<emp.age<<endl;

cout<<"Salary: "<<emp.salary<<endl;

return 0;

}

**

***Q7.1*** *Modify the program in the following way*

*Enter Full name: Magdalena Dankova*

*Enter age: 27*

*Enter salary components*

*Basic:\_\_\_\_*

*HRA:\_\_\_\_\_\_*

*BOOK Allowance:\_\_\_\_\_*

*Furniture Allowance: \_\_\_\_\_\_\_*

*Special allowance:\_\_\_\_\_\_\_\_\_\_\_*

*Total salary = sum(basic+HRA+BOOK Allowance+Furniture Allowance+Special allowance)*

***Solution :***

#include<iostream>

#include<stdio.h>

using namespace std;

struct employee

{

char name[100];

int age;

float salary;

struct modify

{

float basic;

float hra;

float bookallow;

float furniallow;

float specialallow;

} m;

} emp;

int main()

{

cout<<"Enter Full name: ";

gets(emp.name);

cout<<"Enter age: ";

cin>>emp.age;

cout<<"Enter salary components\nBasic: ";

cin>>emp.m.basic;

cout<<"HRA: ";

cin>>emp.m.hra;

cout<<"BOOK Allowance: ";

cin>>emp.m.bookallow;

cout<<"Furniture Allowance: ";

cin>>emp.m.furniallow;

cout<<"Special allowance: ";

cin>>emp.m.specialallow;

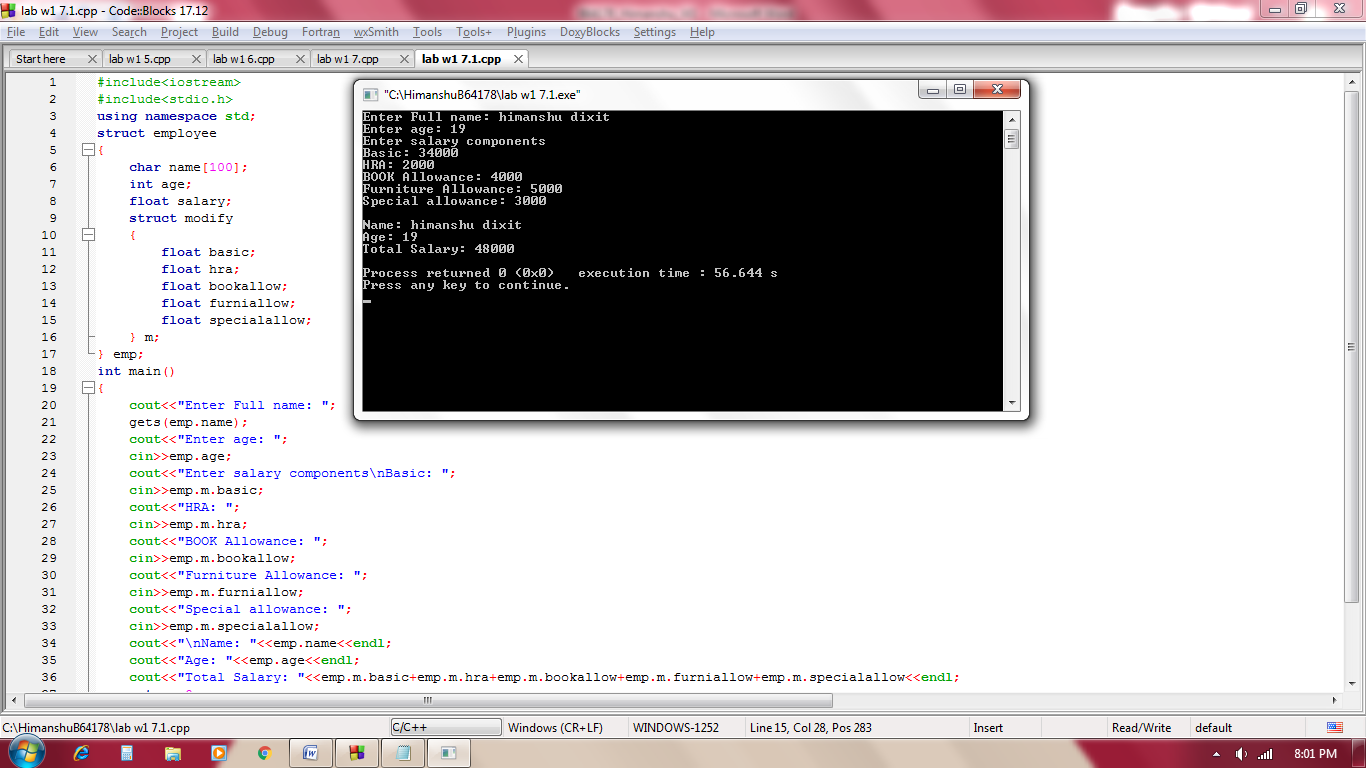
cout<<"\nName: "<<emp.name<<endl;

cout<<"Age: "<<emp.age<<endl;

cout<<"Total Salary: "<<emp.m.basic+emp.m.hra+emp.m.bookallow+emp.m.furniallow+emp.m.specialallow<<endl;

return 0;

}



***Q8.*** *Write a program in C++ to compare individual members of structures.*

***a)*** *Create ‘phone’ structure with Price, Battery Power (In mAH)and Rating (between  0-5) as member variables.*

***b)*** *Create two structure variables (Two phone).*

***c)*** *Compare both phones and display the better phone w.r.t to each criterion (A  better phone is low in price with better Battery Power and high rating).*

*Sample input:*

*Enter Phone1 details:*

*Enter Price:*

*12000*

*Enter Battery Power(In mAh):*

*4000*

*Enter Rating(between 0-5):*

*3.7*

*Enter Phone2 details:*

*Enter Price:*

*15000*

*Enter Battery Power(In mAh):*

*4500*

*Enter Rating(between 0-5):*

*4*

*Sample output:*

*Phone better w.r.t Price is Phone1*

*Phone better w.r.t Battery Power is Phone2*

*Phone better w.r.t Rating is Phone2*

***Solution :***

#include <iostream>

using namespace std;

struct phone

{

float price;

int batpow;

int rating;

} p1,p2;

int main()

{

cout<<"Enter phone 1 details\n";

cout<<"enter price = ";

cin>>p1.price;

cout<<"enter battery power = ";

cin>>p1.batpow;

cout<<"enter rating = ";

cin>>p1.rating;

cout<<"Enter phone 2 details\n";

cout<<"enter price = ";

cin>>p2.price;

cout<<"enter price = ";

cin>>p2.price;

cout<<"enter price = ";

cin>>p2.price;

if(p1.price < p2.price)

cout<<"phone better wrt price is phone1";

else

cout<<"phone better wrt price is phone2";

if(p1.batpow < p2.batpow)

cout<<"phone better wrt battery power is phone2";

else

cout<<"phone better wrt battery power is phone1";

if(p1.rating < p2.rating)

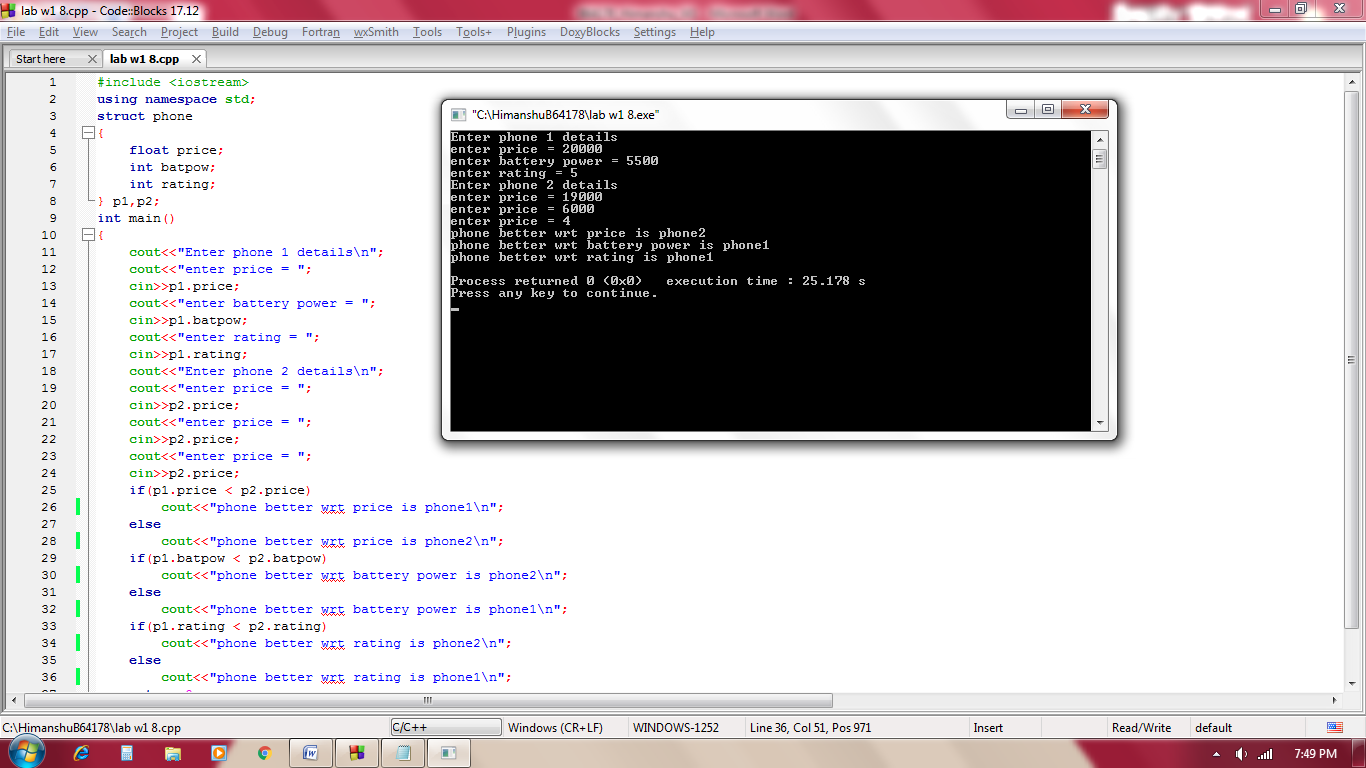
cout<<"phone better wrt rating is phone2";

else

cout<<"phone better wrt rating is phone1";

return 0;

}



***Q9.*** *Write a C++ program to read a 2-D matrix from the user and display it in spiral form.*

*Sample input:*

*Elements of 3 by 6 matrix are: { { 1, 2, 3, 4, 5, 6 },{ 7, 8, 9, 10, 11, 12 },{ 13, 14,  15, 16, 17, 18 } }*

*Sample output:*

*Elements of matrix in spiral form: 1 2 3 4 5 6 12 18 17 16 15 14 13 7 8 9 10  11*

***Solution :***

#include <iostream>

using namespace std;

int main()

{

int a[100][100],r,c;

cout<<"Enter row and column of matrix : ";

cin>>r>>c;

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

{

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

cin>>a[i][j];

}

int tot=r\*c,k=0;

int x=0,y=0,b=0,d=c-1,e=r-1;

while(1)

{

if(k<tot)

{

for(int j=x; j<c; j++)

{

cout<<a[b][j]<<" ";

k++;

}

}

else

break;

if(k<tot)

{

for(int i=y+1; i<r; i++)

{

cout<<a[i][d]<<" ";

k++;

}

}

else

break;

if(k<tot)

{

for(int j=d-1; j>=0; j--)

{

cout<<a[e][j]<<" ";

k++;

}

}

else

break;

if(k<tot)

{

for(int i=e-1; i>0; i--)

{

cout<<a[i][x]<<" ";

k++;

}

}

else

break;

x++;

c--;

r--;

b++;

y++;

d--;

e--;

}

return 0;

}

