//2D ARRAY DYNAMIC MEMORY ALLOCATION:

#include<iostream>

using namespace std;

int main(){

int r,c;

cout<<"no of rows:";

cin>>r;

cout<<"no of columns:";

cin>>c;

int a[r][c];

int \*\*p=new int\*[r];

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

p[i]=new int[c];

}

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

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

cin>>a[i][j];

}

}

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

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

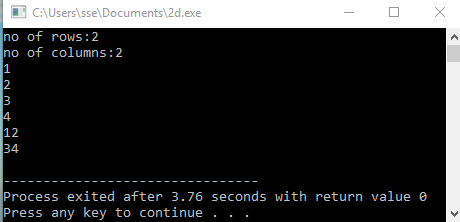
cout<<a[i][j];

}

}

}

OUTPUT:



//STATIC DATA AND MEMBER FUNCTION:

#include<iostream>

using namespace std;

class st{

public:

int a;

static int b;

void get(){

cout<<"value for a:";

cin>>a;

cout<<"value for b:";

cin>>b;

}

static void far(){

cout<<b;

// cout<<a;

}

};

int st::b;

int main(){

st obj;

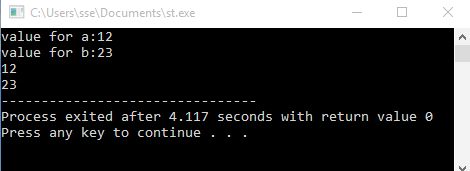
obj.get();

cout<<obj.a<<endl;

st::far();

}

OUTPUT:



//USER DEFINED EXCEPTION:

#include<iostream>

#include<exception>

using namespace std;

class name:public exception{

public:

const char\* what() const throw(){

cout<<"CANNOT DIVISIBLE BY ZERO!!";

}

};

int main(){

int a,b;

try{

cout<<"enter the value of a:";

cin>>a;

cout<<"enter the value of b:";

cin>>b;

if(b==0){

name obj;

throw obj;

}

else

cout<<"division of two num:"<<a/b;

}

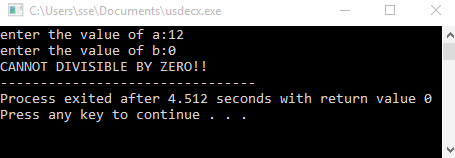
catch(exception &e){

e.what();

}

}

OUTPUT:



//VIRTUAL BASE CLASS

#include<iostream>

using namespace std;

class A{

public:

void display(){

cout<<" I AM VIRTUAL BASE CLASS!";

}

};

class B:virtual public A{

};

class C:virtual public A{

};

class D:public B,public C{

};

int main(){

D obj;

obj.display();

}

OUTPUT:

