HW4

1.

請寫一程式，判斷 (2,1)，(2,-1)，(3,3)，(5,0)，(1,-7)，(0,-2) 中，有哪些點在圓心為(0,0)且半徑為3的圓內(上)，並輸出在圓內(上)座標

(圓方程式為：)

//圓內

#include <iostream>

#include <cstdlib>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{

int c[6][2] = {{2,1},{2,-1},{3,3},

{5,0},{1,-7},{0,-2}};

//c = {{2,1},{2,-1},{3,3},{5,0},{1,-7},{0,-2}};

int ox = 0,oy = 0; //圓心

cout<<"在圓內(上)的座標點 :";

for (int x=0;x<=5;x++){

if (sqrt(c[x][0]\*c[x][0]+c[x][1]\*c[x][1])<=sqrt((3-0)\*(3-0)+(3-0)\*(3-0))){

cout<<setw(5)<<"("<<c[x][0]<<","<<c[x][1]<<")";

}

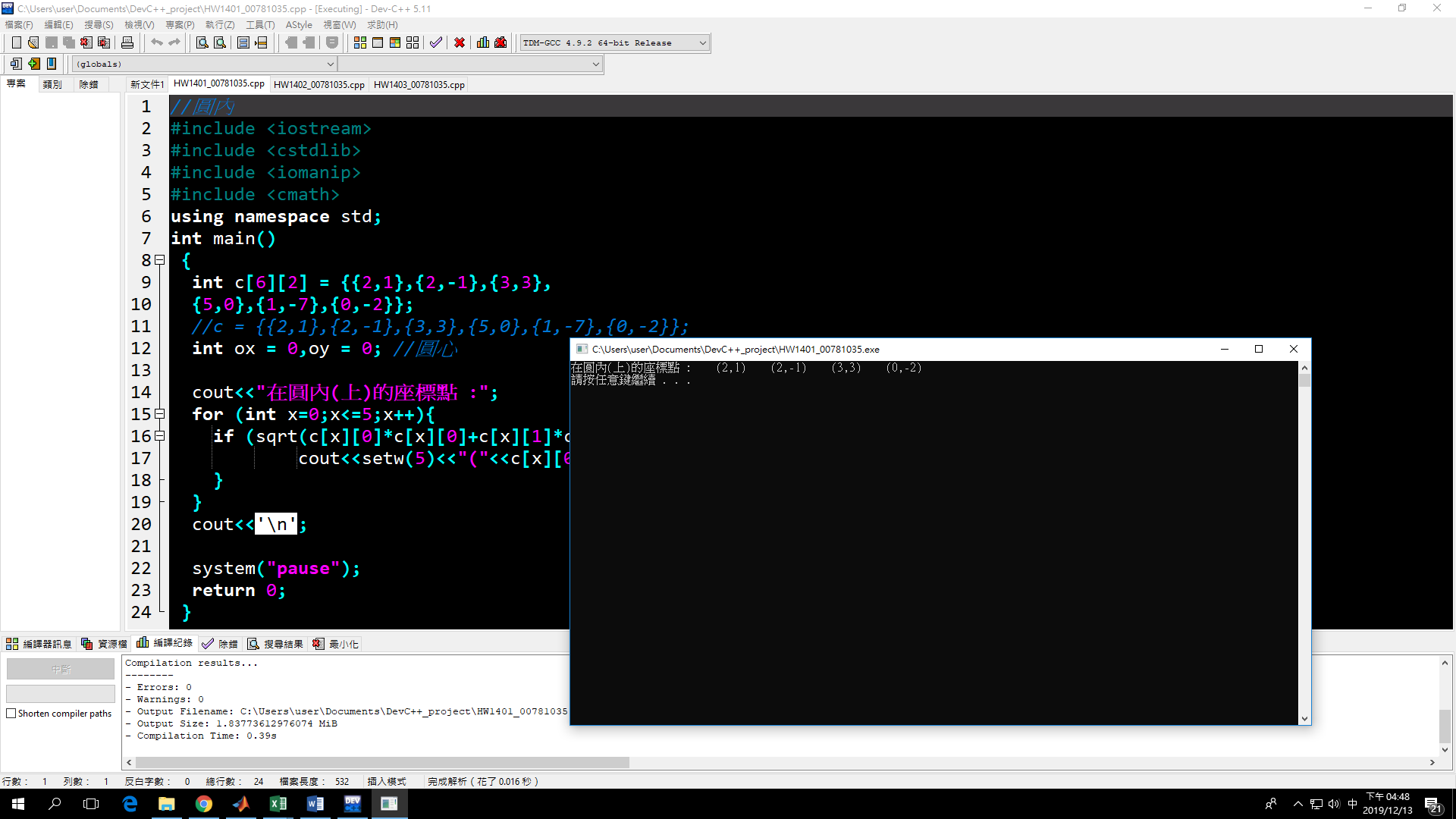
}

cout<<'\n';

system("pause");

return 0;

}



2.

請寫一程式，宣告一個二維陣列int data[3][2]={1,2,3,4,5,6} ;利用指標的方式，輸出data的每一個元素之內容及所在之位址。

//一重指標、二維陣列

#include <iostream>

#include <cstdlib>

#include <iomanip>

using namespace std;

int main()

{

int num[3][2] = {1,2,3,4,5,6};

int \*ptr;

ptr = num[0];

//取得num陣列第一個元素所在的記憶體位址給ptr儲存

//\*ptr 間接存取num陣列第一個元素的記憶體位址的內容

cout<<"num陣列第一個元素的記憶體位址的內容:"<<\*ptr

<<'\n'<<"num陣列第一個元素所在的記憶體位址"<<ptr<<"\n\n";

for (int i=0;i<6;i++){

cout<<setw(10)<< \*ptr;

//cout<<setw(10)<< num[i];

ptr++;

}

//cout<<"\nnum陣列最後一個元素的記憶體位址的內容:"<<\*ptr

//<<'\n'<<"num陣列最後一個元素所在的記憶體位址"<<ptr<<'\n';

cout << '\n';

for (int i=0;i<6;i++){

cout<<setw(10)<< num[i];

ptr++;

}

cout << '\n';

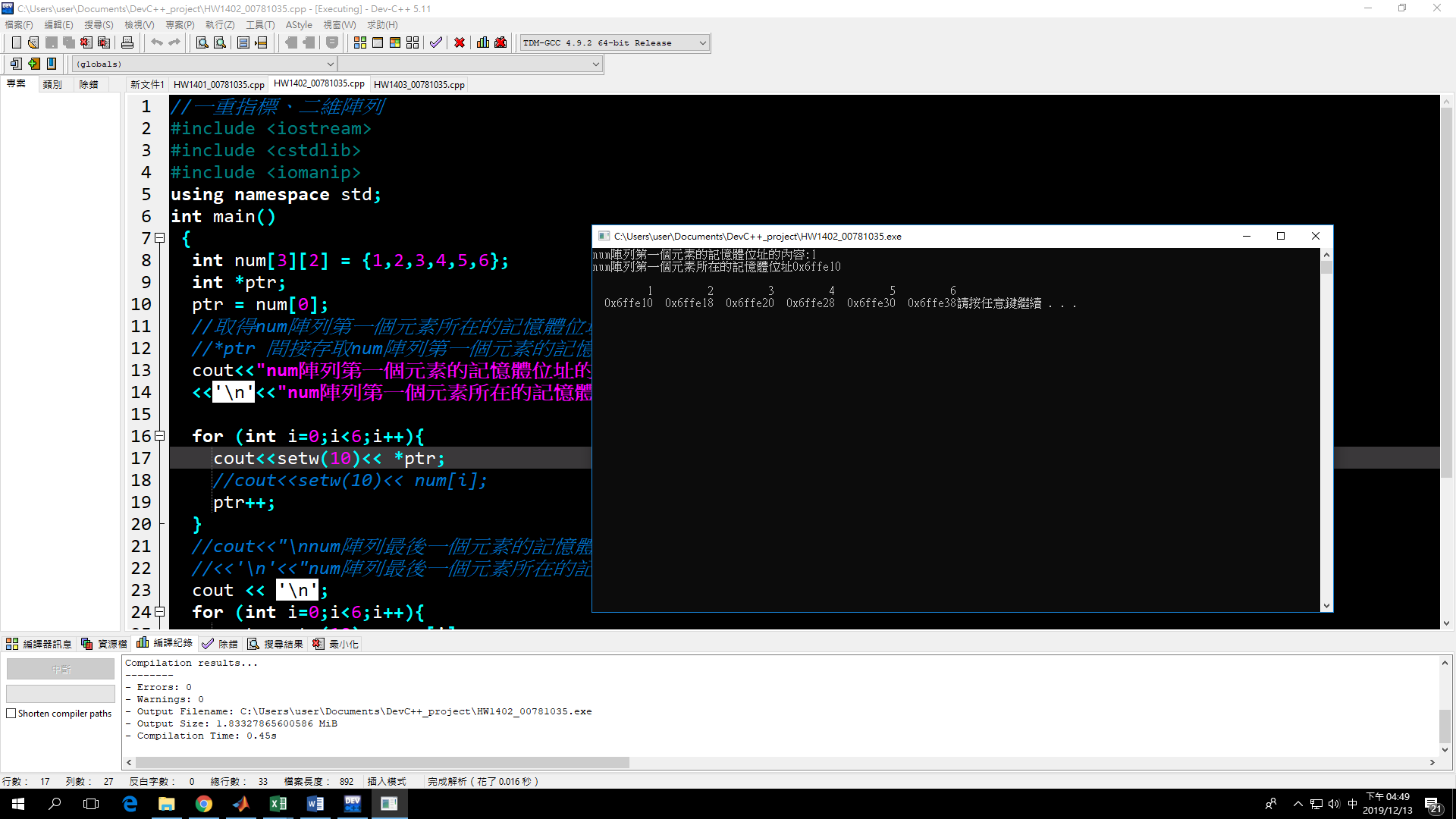
//cout<<"\nnum陣列最後一個元素的記憶體位址的內容:"<<\*ptr

//<<'\n'<<"num陣列最後一個元素所在的記憶體位址"<<ptr<<'\n';

system("pause");

return 0;

}



3.

請寫一程式，印出以下電影院座位表(顯示號碼即可)

[10][11][12][13][14]

[15][16][17][18][19][20][21]

[22][23][24][25][26][27][28][29][30]

(回想金字塔圖形如何表示)

#include <iostream>

#include <cstdlib>

using namespace std;

int main(){

int num[25] = {6,7,8,9,10,11,12,13,14,

15,16,17,18,19,20,21,22,23,24,25,26,27,

28,29,30};

int i,j,k;

int layer,n;

cout << "輸入電影座位金字塔層數: "; //7

cin >> layer;

cout << "第幾排開始? : "; //3

cin>>n;

cout<<"\n";

k = n-1; //3-1

// k 用在第二個 j 迴圈

for (i = (n-1)\*(n-1)+1;i <= layer;i++){

for (j=1;j<=(layer-i)\*2;j++){

cout << ' ';

}

//2平方~4平方

for (j=k\*k+1;j<=(k+1)\*(k+1);j++){

cout << num[j-1];

}

cout << '\n';

k++;

}

}

