1.

#include <iostream>

#include <iomanip>

using namespace std;

class Pascal

{

private:

int\*\* triangle;

int \_row;

static ostream& itv(ostream& o) { return o << " "; }; //间隔

static ostream& half\_itv(ostream& o) { return o << " "; } //半个间隔

static ostream& blk(ostream& o) { return o << " "; } //每个数字占位

static ostream& half\_blk(ostream& o) { return o << " "; } //半个数字占位

public:

Pascal(int row);

void print();

~Pascal();

};

Pascal::Pascal(int row) :\_row(row)

{

triangle = new int\* [row];

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

{

triangle[i] = new int[i + 1];

triangle[i][0] = triangle[i][i] = 1;

for (int j = 1; j < i; ++j)

triangle[i][j] = triangle[i - 1][j - 1] + triangle[i - 1][j];

}

}

void Pascal::print()

{

for (int i = 0; i < \_row; ++i)

{

int tmp = (\_row - 1 - i) / 2;

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

cout << itv << blk;

if ((\_row - 1 - i) % 2)

cout << half\_itv << half\_blk;

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

cout << setw(4) << triangle[i][j] << itv;

cout << endl;

}

}

Pascal::~Pascal()

{

for (int i = 0; i < \_row; ++i)

delete[] triangle[i];

delete[] triangle;

}

int main()

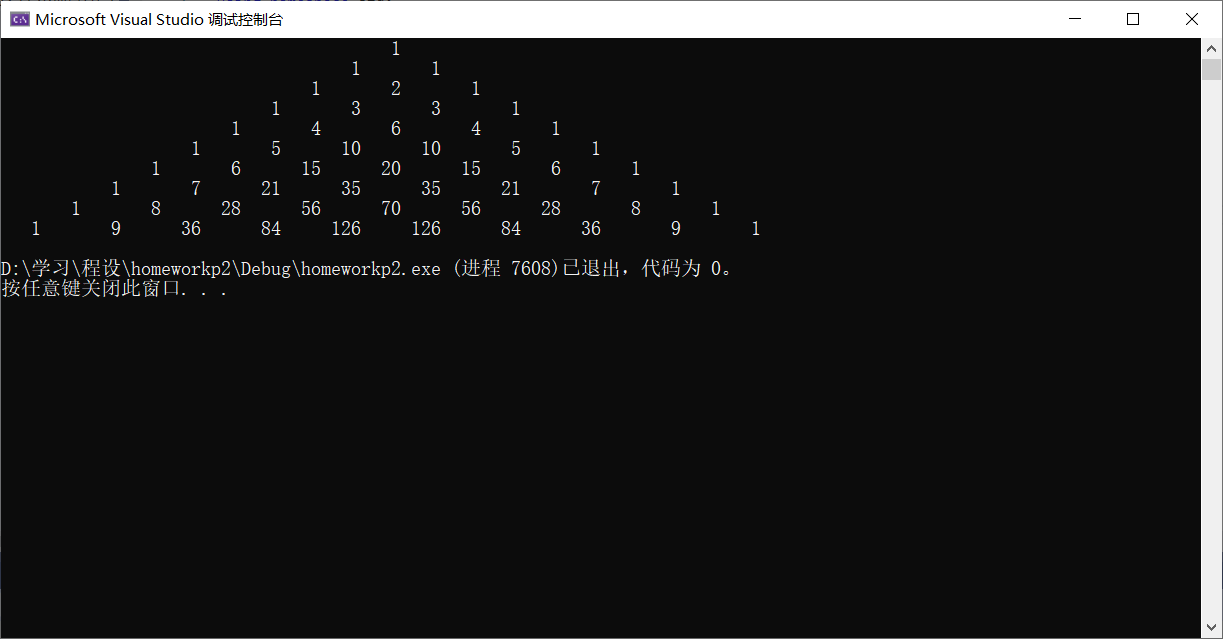
{

Pascal p(10);

p.print();

return 0;

}



2.

#include <iostream>

using namespace std;

int main()

{

int\* a = new int[100];

int(\*b)[25] = new int[20][25];

int(\*c)[8][5] = new int[9][8][5];

int m, n;

cin >> m >> n;

int sum = 0;

for (int i = 0; i < 100; ++i)

sum += (a[i] = i);

cout << "The sum of a is: " << sum << endl;

sum = 0;

for (int i = 0; i < 20; ++i)

for (int j = 0; j < 25; ++j)

sum += (b[i][j] = i \* 25 + j);

cout << "The sum of b is: " << sum << endl;

sum = 0;

for (int i = 0; i < 9; ++i)

for (int j = 0; j < 8; ++j)

for (int k = 0; k < 5; ++k)

sum += (c[i][j][k] = i \* 100 + j \* 10 + k);

cout << " The sum of c is: " << sum << endl;

sum = 0;

int\*\* d = new int\* [m];

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

{

d[i] = new int[n];

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

sum += (d[i][j] = i \* n + j);

}

cout << "The sum of d is: " << sum << endl;

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

delete[] d[i];

delete[] d;

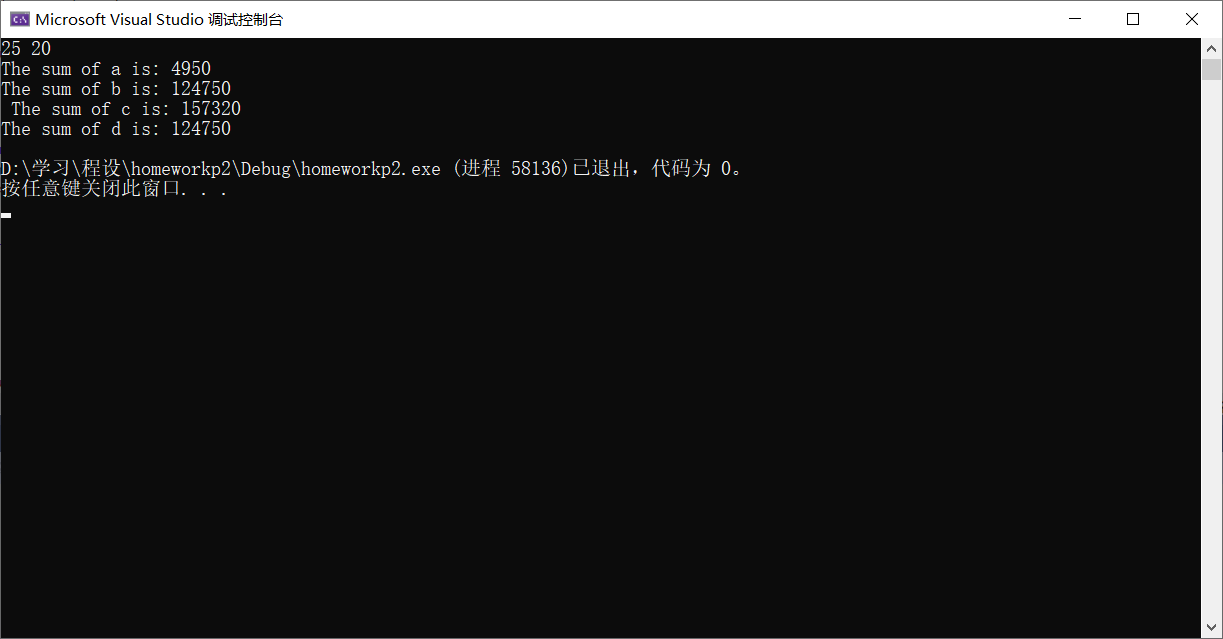
delete[] c;

delete[] b;

delete[] a;

return 0;

}



3（探究题）：

#include <iostream>

#include <string>

using namespace std;

class Ball

{

public:

enum Color //Ball类内的枚举类型

{

Red,

Yellow,

Blue,

White,

Black

};

Ball(Color color) :\_color(color) {} //初始化\_color

string GetColor() //得到颜色字符串

{

switch (\_color)

{

case Red:

return static\_cast<string>("红"); //将C风格字符串转换为C++string对象

case Yellow:

return static\_cast<string>("黄");

case Blue:

return static\_cast<string>("蓝");

case White:

return static\_cast<string>("白");

case Black:

return static\_cast<string>("黑");

default:

cout << "No such Color!" << endl;

return static\_cast<string>("No such Color!");

}

}

private:

Color \_color;

};

int main()

{

Ball ball[5] = { Ball::Red, Ball::Yellow, Ball::Blue, Ball::White, Ball::Black };

int sum = 0;

for(int i = 0; i < 3; ++i)

for(int j = i + 1; j < 4; ++j)

for (int k = j + 1; k < 5; ++k)

{

++sum;

cout << ball[i].GetColor() << ", " << ball[j].GetColor() << ", " << ball[k].GetColor() << endl;

}

cout << "总取法数是：" << sum;

return 0;

}

