1.

#include "pch.h"

#include <iostream>

using namespace std;

class CPU {

public:

CPU(int f, double v) :frequency(f), voltage(v) {cout << "调用函数" << endl; }

~CPU() {cout << "调用析构函数" << endl; }

void run();

void stop();

private:

enum CPU\_Rank{P1=1,P2,P3,P4,P5,P6,P7};

int frequency;

double voltage;

};

void CPU::run() {

cout << "程序开始执行" << endl;

CPU\_Rank rank=P1;

cout << "等级：" << P1<<endl;

cout << "频率：" << frequency<<"MHz"<< endl;

cout << "电压：" << voltage << endl;

}

void CPU::stop() {

cout << "程序停止调用" << endl;

}

int main() {

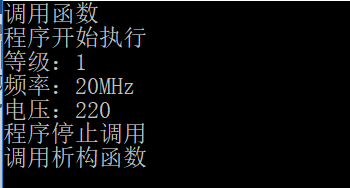
CPU cpu(20, 220);

cpu.run();

cpu.stop();

return 0;

}



2

. #include "pch.h"

#include <iostream>

using namespace std;

enum CPU\_Rank{P1=1,P2,P3,P4,P5,P6,P7};

class CPU {

private:

CPU\_Rank rank;

int frequency;

double voltage;

public:

CPU() {};

CPU(CPU\_Rank r, int f, double v) {

rank = r;

frequency = f;

voltage = v;

cout << "构造了CPU！" << endl;

}

CPU(CPU &CPU1) { cout << "复制构造了一个CPU！" << endl; }

~CPU() { cout << "析构了一个CPU!" << endl; }

};

class Ram {

private:

int ram;

public:

Ram() {};

Ram(int m) {

ram = m;

cout << "构造了Ram！" << endl;

}

Ram(Ram &Ram) { cout << "复制构造了一个Ram！" << endl; }

~Ram() { cout << "析构了一个Ram！" << endl; }

};

class CDROM {

private:

int cdrom;

public:

CDROM() {};

CDROM(int c) {

cdrom = c;

cout << "构造了一个CDROM!" << endl;

}

CDROM(CDROM &CDROM) { cout << "复制构造了一个CDROM！" << endl; }

~CDROM() { cout << "析构了一个CDROM！" << endl; }

};

class Computer {

private:

CPU A;

Ram B;

CDROM C;

public:

Computer(CPU\_Rank r, int f, double v, int m, int c) { cout << "调用Computer！" << endl; }

void run() { cout << "Computer 开始运行！" << endl; };

void stop() { cout << "Computer停止运行！" << endl; };

~Computer() { cout << "析构了一个Computer！" << endl; }

};

int main() {

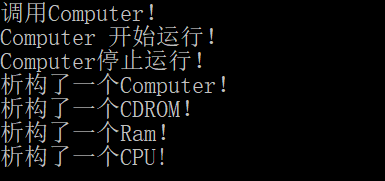
Computer computer(P1, 20, 220, 6, 7);

computer.run();

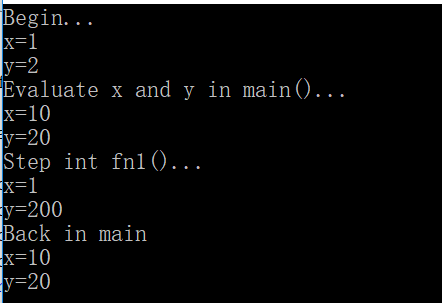
computer.stop();

return 0;

}



3. （1）



（2）

#include "pch.h"

##include <string.h>

using namespace std;

class CLIENT {

private:

static char SERVERNAME;

static int CLIENTNUM;

public:

CLIENT() {};

~CLIENT() {};

static int GETCLIENTNUM();

static void CHANGESERVERNAME(char name) {};

static char GETSERVERNAME();

};

CLIENT::CLIENT() {

CLIENTNUM--;

}

int CLIENT::GETCLIENTNUM() {

return CLIENTNUM;

}

void CLIENT::CHANGESERVERNAME(char name) {

SEVERNAME = name;

}

char CLIENT::GETSERVERNAME() {

return SERVERNAME;

}

#include <iostream.h>

int CLIENT::CLIENTNUM = 0;

char CLIENT::SERVERNAME = 'A';

int main() {

CLIENT a;

CLIENT b;

cout << "服务器名称为：" << CLIENT::GETSERVERNAME() << endl;

CLIENT::CHANGESERVERNAME('G');

cout << "客户机数量为：" << CLIENT::GETCLIENTNUM() << endl;

cout << "服务器名称为：" << CLIENT::GETSERVERNAME() << endl;

}