西安交通大学实验报告

课程 计算机程序设计 实验名称 类与对象 第 1 页 共 17 页

系 别 钱 学 森 学 院 实 验 日 期 2017年 4 月 27 日

专业班级 钱62 组别\_\_\_\_\_\_\_\_\_\_\_ 实 验 报 告 日 期 2017 年 4 月27日

姓 名 周宇晨 学号 2160405046 报 告 退 发 ( 订正 、 重做 )

同 组 人\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 教 师 审 批 签 字

一 实验目的

1.理解类与对象的概念。

2.掌握类的定义方法和使用方法。

二 实验题目

题目一 小车类

1.要点分析

在小车类中增加两个输入形状与倒车的函数。

2.程序源码

#include<iostream>

#include<cstring>

#include<windows.h>

using namespace std;

class automobile

{

private:

char type[20]; char color[20];

float price;

int carry\_weight;

int carry\_customer;

char shape[20];

public:

void set\_data(char \*t, char \*c, float pri, int cw, int cc, char \*p);

void movecar(int l);

void horming(int num);

void downcar(int l);

char \*show\_type() { return type; }

void daochecar(int l);

void xiexiacar(int l);

};

void automobile::set\_data(char \*t, char \*c, float pri, int cw, int cc, char \*p)

{

strcpy(type, t);

strcpy(color, c);

price = pri;

carry\_weight = cw;

carry\_customer = cc;

strcpy(shape, p);

}

void automobile::movecar(int l)

{

cout << "\n" << type << "水平直线运动:" << endl;

int n = strlen(shape);

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

{

cout << ' ' << shape;

Sleep(500);

for (int j = 0; j<n; j++)cout << '\b';

}

}

void automobile::downcar(int l)

{

cout << "\n" << type << "垂直下降运动:" << endl;

int n = strlen(shape);

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

{

cout << shape;

Sleep(500);

for (int j = 0; j<n; j++)cout << '\b';

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

cout << endl;

}

}

void automobile::horming(int num)

{

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

{

cout << type;

cout << '\007' << " dang!" << endl;

Sleep(1000);

}

}

void automobile::daochecar(int l)

{

int i, j;

cout << "\n" << type << "倒车:" << endl;

int n = strlen(shape);

for (i = 0; i<10; i++) cout << " ";

for (i = 0; i<l; i++)

{

cout << shape;

Sleep(500);

for (j = 0; j<n; j++)cout << '\b';

for (j = 0; j<n; j++)cout << " ";

for (j = 0; j<n + 1; j++)cout << '\b';

}

}

void automobile::xiexiacar(int l)

{

int i, j, k;

cout << "\n" << type << "斜向下:" << endl;

int n = strlen(shape);

for (i = 0; i<l; i++)

{

cout << shape;

Sleep(500);

for (j = 0; j<n; j++)cout << '\b';

for (j = 0; j<n; j++)cout << " ";

cout << '\n';

for (k = 0; k <= i; k++)cout << " ";

}

}

int main()

{

cout << "默认汽车型号为yyida，颜色为white，价格，载重量t，载客量人" << endl;

cout << "请输入汽车形状" << endl;

automobile nison;

char tp[] = "yyida";

char ys[] = "white";

char a[20];

cin >> a;

nison.set\_data(tp, ys, 200000, 5, 3, a); //测试设置初始数据函数

nison.horming(5); //测试鸣笛函数

nison.movecar(5); //测试水平直线运动

nison.downcar(5); //测试垂直下降运动

nison.daochecar(10); //测试倒车

nison.xiexiacar(10); //测试斜向下运动

return 0;

}

题目二 定义一个Dog类

1.要点分析

定义Dog类，键入属性。

2.程序源码

#include <iostream>

using namespace std;

class Dog

{

private:

int age;

float weight;

char name[20], sex[5];

public:

void set(char n[], int a, char s[], float w)

{

age = a;

weight = w;

strcpy(name, n);

strcpy(sex, s);

}

void show();

};

void Dog::show()

{

cout << "姓名：" << name << endl;

cout << "年龄：" << age << endl;

cout << "性别：" << sex << endl;

cout << "体重:" << weight << endl;

}

int main()

{

Dog jb;

char name[20], sex[5];

int age;

float weight;

cout << "请输入狗狗的姓名：";

cin >> name;

cout << "请输入狗狗的年龄:";

cin >> age;

cout << "请输入狗狗的性别：";

cin >> sex;

cout << "请输入狗狗的体重（kg）：";

cin >> weight;

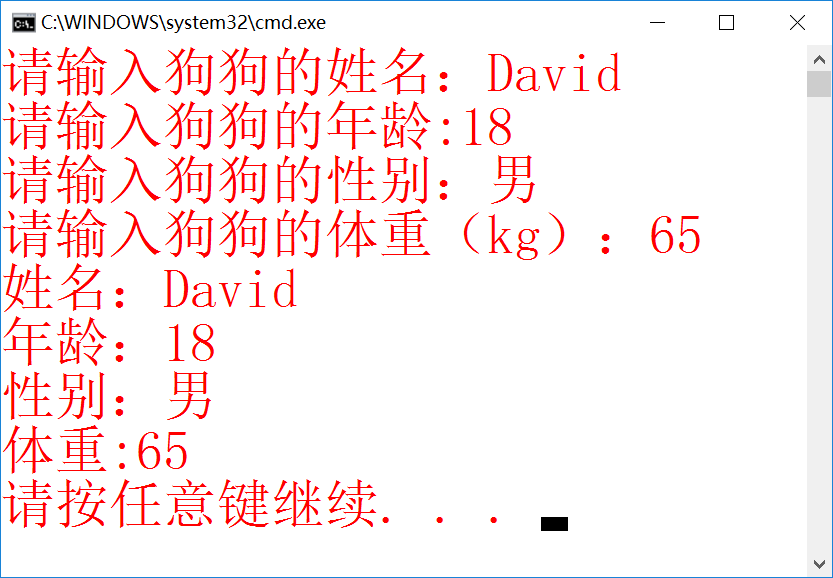
jb.set(name, age, sex, weight);

jb.show();

return 0;

}

3.实验结果



题目三 椭圆类

1.要点分析

定义椭圆类，定义area函数计算面积。

2.程序源码

#include <iostream>

using namespace std;

class Ellipse

{

private:

float xa, ya;

float xb, yb;

public:

void set(float x1,float x2,float y1,float y2)

{

xa = x1;

ya = y1;

xb = x2;

yb = y2;

}

float area()

{

return 3.1415926\*abs(xa - xb)\*abs(ya - yb) / 4;

}

};

int main()

{

float cxa, cxb, cya, cyb;

Ellipse el;

cout << "请输入椭圆外界矩形左上角坐标：";

cin >> cxa >> cya;

cout << "请输入椭圆外界矩形右下角坐标：";

cin >> cxb >> cyb;

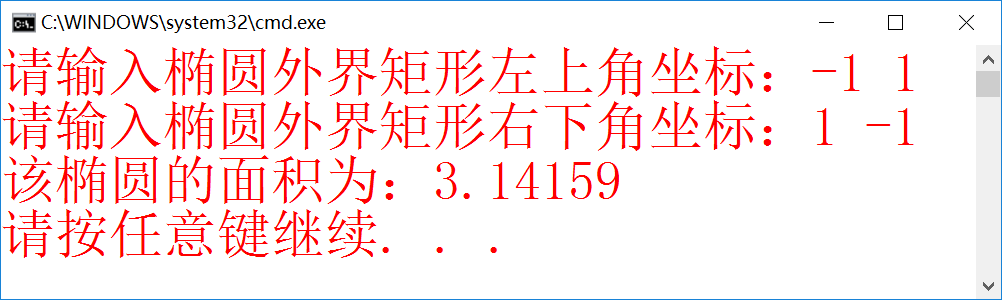
el.set(cxa, cxb, cya, cyb);

cout << "该椭圆的面积为：" << el.area() << endl;

return 0;

}

3.实验结果



题目四 Time类

1.要点分析

输入时间，当输入时间不符合时返回系统时间。并按三种格式输出。

2.程序源码

#include<iostream>

#include<time.h>

using namespace std;

class Time

{

private:

int hour, minute, second;

void SetSystemTime(); //取得系统时间

public:

void init(int, int, int);

void print\_1();

void print\_2();

void print\_3();

int get\_hour() { return hour; }

int get\_minute() { return minute; }

int get\_second() { return second; }

};

void Time::SetSystemTime()

{

//取得系统时间

tm \*gm;

time\_t t = time(NULL);

gm = gmtime(&t);

hour = gm->tm\_hour;

minute = gm->tm\_min;

second = gm->tm\_sec;

}

void Time::init(int hh, int mm, int ss)

{

hour = hh;

if (mm < 60)

minute = mm;

else

{

SetSystemTime();

return;

}

if (ss < 60)

second = ss;

else

{

SetSystemTime();

return;

}

};

void Time::print\_1()

{

cout << hour << "'" << minute << "''" << second << endl;

}

void Time::print\_2()

{

cout << hour << "时" << minute << "分" << second << "秒" << endl;

}

void Time::print\_3()

{

cout << hour << ":" << minute << ":" << second << endl;

}

int main()

{

Time demo;

int h, m, s;

cin >> h >> m >> s;

demo.init(h,m,s);

demo.print\_1();

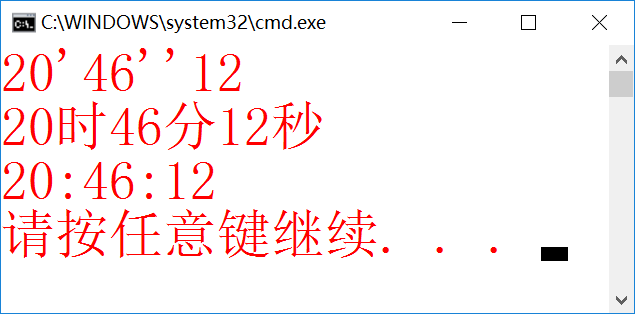
demo.print\_2();

demo.print\_3();

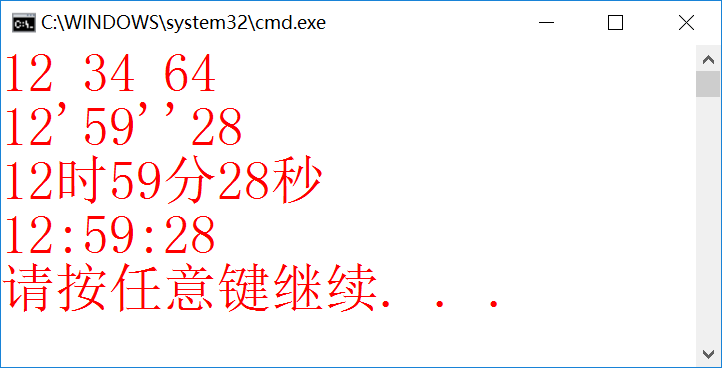
return 0;

}

3.实验结果



输入时间错误：



题目五 定义一个三角形类

1.要点分析

定义三角形类，定义area函数用海伦公式计算面积。

2.程序源码

#include<iostream>

using namespace std;

class triangle

{

private:

int a, b, c;

public:

void init()

{

cout << "请输入三角形三边长:";

cin >> a >> b >> c;

}

float area()

{

int p = (a + b + c) / 2;

return sqrt(p\*(p - a)\*(p - b)\*(p - c));

}

void printa()

{

cout << "三角形面积是：" << area() << endl;

}

};

int main()

{

triangle demo;

demo.init();

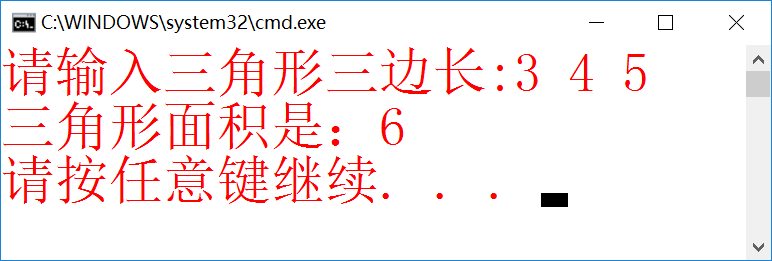
demo.area();

demo.printa();

return 0;

}

3.实验结果



题目六 二元一次方程组求解

1.要点分析

求根公式；输出格式的控制。

2.程序源码

#include<iostream>

using namespace std;

class EGroup

{

private:

double a, b, c, d, e, f, g;

double x1, x2;

public:

void init()

{

cout << "请输入系数：";

cin >> a >> b >> c >> d >> e >> f;

}

void slove()

{

x1 = (c\*e - b\*f) / (a\*e - b\*d);

x2 = (c\*d - a\*f) / (b\*d - a\*e);

}

void prints()

{

cout << "方程根:\nx=" << x1 << endl;

cout << "y=" << x2 << endl;

}

void printe()

{

cout << "方程为：" << endl;

if (a < 0 && a != -1)cout << a << "x+";

else if (a == -1)cout << "-x+";

else if (a > 0 && a != 1)cout << a << "x+";

else if (a == 1)cout << "x+";

if (b < 0 && b != -1)cout << "\b" << b << "y=";

else if (b == -1)cout << "\b-y=";

else if (b == 0)cout << "\b=";

else if (b > 0 && b != 1)cout << b << "y=";

else if (b == 1)cout << "y=";

cout << c << endl;

if (d < 0 && d != -1)cout << d << "x+";

else if (d == -1)cout << "-x+";

else if (d > 0 && d != 1)cout << d << "x+";

else if (d == 1)cout << "x+";

if (e < 0 && e != -1)cout << "\b" << e << "y=";

else if (e == -1)cout << "\b-y=";

else if (e == 0)cout << "\b=";

else if (e > 0 && e != 1)cout << e << "y=";

else if (e == 1)cout << "y=";

cout << f << endl;

}

};

int main()

{

EGroup equal;

equal.init();

equal.printe();

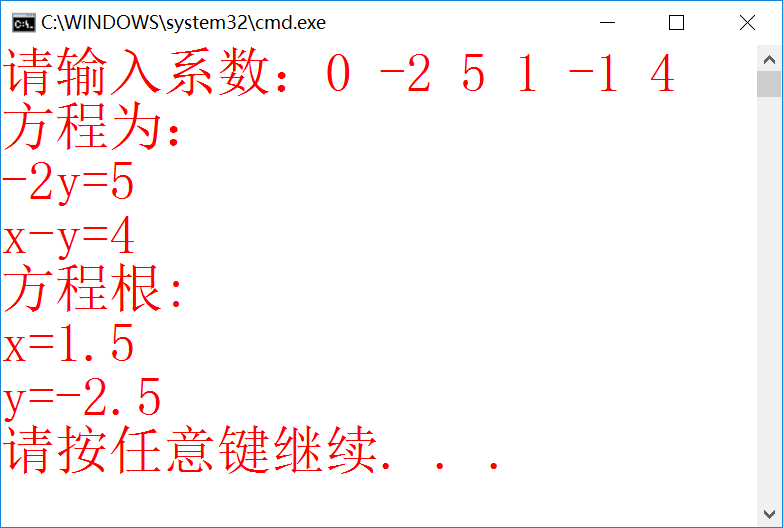
equal.slove();

equal.prints();

return 0;

}

3.实验结果



题目七 机器人

1.要点分析

定义机器人类；使用judge函数判定调用翻译周、月、整数函数中的哪几个。

2.程序源码

#include<iostream>

using namespace std;

class robot

{

private:

char robname[20];

int year;

int perform;

int week, month, x;

public:

void set()

{

cout << "请输入机器人名字：";

cin >> robname;

cout << "请输入机器人生产年份：";

cin >> year;

cout << "请输入机器人性能指数（0-3）：";

cin >> perform;

while (perform < 0 || perform > 3)

{

cout << "性能指数输入错误，请重新输入：";

cin >> perform;

}

}

void input()

{

if (perform == 1)

{

cout << "请输入周：";

cin >> week;

}

else if (perform == 2)

{

cout << "请输入周：";

cin >> week;

cout << "请输入月：";

cin >> month;

}

else if (perform == 3)

{

cout << "请输入周：";

cin >> week;

cout << "请输入月：";

cin >> month;

cout << "请输入整数：";

cin >> x;

}

}

int get\_f()

{

return perform;

}

void judge()

{

if (perform == 1)

{

tran\_weekday(week);

}

else if (perform == 2)

{

tran\_weekday(week);

tran\_month(month);

tran\_int(x);

}

else if (perform == 3)

{

tran\_weekday(week);

tran\_month(month);

tran\_int(x);

}

}

void tran\_weekday(int weekday);

void tran\_month(int month);

void tran\_int(int k);

};

void robot::tran\_month(int month)

{

cout << "月份翻译结果：";

switch (month)

{

case 1:cout << "Janurary" << endl; break;

case 2:cout << "Feburary" << endl; break;

case 3:cout << "March" << endl; break;

case 4:cout << "April" << endl; break;

case 5:cout << "May" << endl; break;

case 6:cout << "June" << endl; break;

case 7:cout << "July" << endl; break;

case 8:cout << "August" << endl; break;

case 9:cout << "September" << endl; break;

case 10:cout << "October" << endl; break;

case 11:cout << "November" << endl; break;

case 12:cout << "December" << endl; break;

default:cout << "Month Error!" << endl;

}

}

void robot::tran\_weekday(int weekday)

{

cout << "星期翻译结果：";

switch (weekday)

{

case 1:cout << "Monday" << endl; break;

case 2:cout << "Tuesday" << endl; break;

case 3:cout << "Wednesday" << endl; break;

case 4:cout << "Thursday" << endl; break;

case 5:cout << "Friday" << endl; break;

case 6:cout << "Saturday" << endl; break;

case 7:cout << "Sunday" << endl; break;

}

}

void trge(int a)//将一位数转化为相应的英文

{

if (a == 1) cout << "one" << ' ';

else if (a == 2) cout << "two" << ' ';

else if (a == 3) cout << "three" << ' ';

else if (a == 4) cout << "four" << ' ';

else if (a == 5) cout << "five" << ' ';

else if (a == 6) cout << "six" << ' ';

else if (a == 7) cout << "seven" << ' ';

else if (a == 8) cout << "eight" << ' ';

else if (a == 9) cout << "nine" << ' ';

}

void trshi(int x)//将2位数转化为相应的英文（几十几）

{

int a = x / 10, b = x % 10;

if (a == 0)

{

if (b != 0) trge(b);//若为个位数则按上一个函数输出

}

else if (a == 1)//十位数为1的情况

{

if (b == 0) cout << "ten" << ' ';

if (b == 1) cout << "eleven" << ' ';

if (b == 2) cout << "twelve" << ' ';

if (b == 3) cout << "thirteen" << ' ';

if (b == 4) cout << "fourteen" << ' ';

if (b == 5) cout << "fifteen" << ' ';

if (b == 6) cout << "sixteen" << ' ';

if (b == 7) cout << "seventeen" << ' ';

if (b == 8) cout << "eighteen" << ' ';

if (b == 9) cout << "nighteen" << ' ';

}

else if (a == 2)

{

cout << "twenty" << ' '; trge(b);

}

else if (a == 3)

{

cout << "thirty" << ' '; trge(b);

}

else if (a == 4)

{

cout << "forty" << ' '; trge(b);

}

else if (a == 5)

{

cout << "fifty" << ' '; trge(b);

}

else if (a == 6)

{

cout << "sixty" << ' '; trge(b);

}

else if (a == 7)

{

cout << "seventy" << ' '; trge(b);

}

else if (a == 8)

{

cout << "eighty" << ' '; trge(b);

}

else if (a == 9)

{

cout << "ninety" << ' '; trge(b);

}

}

void trbai(int y)//将三位数转化为英文

{

int a, b, c;

a = y / 100;

b = y - 100 \* a;//分离百位和后两位

if (a)//若a不为0

{

trge(a);

cout << "hundred" << ' '; trshi(b);

}

else trshi(b);

}

void robot::tran\_int(int x)

{

cout << "整数翻译结果：";

int a, b, c, d;

a = x / 1000000000; b = (x - a \* 1000000000) / 1000000; d = x % 1000; c = (x / 1000) % 1000;//按分节符分离各三位数

if (a != 0) { trbai(a); cout << "billion" << ' '; }

if (b != 0) { trbai(b); cout << "million" << ' '; }

if (c != 0) { trbai(c); cout << "thousand" << ' '; }

trbai(d);//输出结果

cout << endl;

}

int main()

{

robot rob;

rob.set();

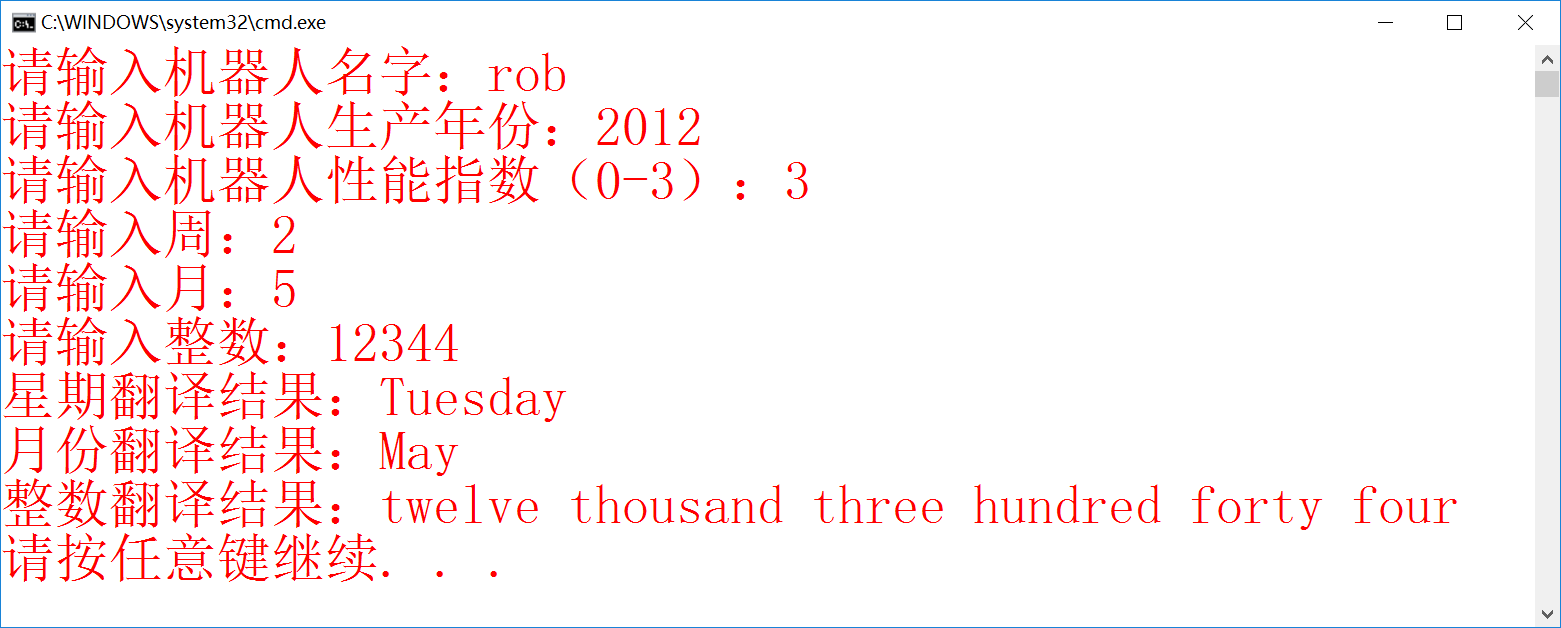
rob.input();

rob.judge();

return 0;

}

3.实验结果



三 个人小结

本周学习了类与对象的相关知识。从我个人的体验来看，就目前的使用范围而言类和结构体并没有本质上的区别，理解起来也完全可以用类比的思路，难度并不大。本周的题目也比较基础，突出了实用性，越来越贴近现实，能用来解决实际问题。虽然这周体验很舒适，但考虑到对象两字凶名在外，不得不小心，下周的课一定要认真听讲。

另拜读了李雨桉上周的实验报告。必须要突出指出的是排版问题。这方面我主观上就不愿意花时间，然而在呈现上这一点的效果确实是十分显著的。对于书写者可能并无区别，但对阅读者来说要抓住主要信息排版的提示就非常重要了（来自周末为完成数学建模阅读大量文献的哀叹）。