/\*\*\*

练习1.3 练习1.4 练习1.5

2019年3月22日22:26:13

by PK

\*\*\*\*/

#include <iostream>

#include <stdio.h>

int main(void)

{

/\*练习1.3

输出hello world

\*/

printf("hello C hello world! \r\n"); //C

std::cout << "hello C++ hello singledog! \r\n"<<std::endl;

//以上为练习1.3

/\*\* 练习1.4

计算两个输入数的加和乘

\*\*/

int num1,num2,res;

std::cout<<"please input two number \r\n"<<std::endl;

std::cin>>num1>>num2;

res = num1+num2;

std::cout<<"num1+num2 = "<<res<<std::endl;

res = num1\*num2;

std::cout<<"num1\*num2 = "<<res<<std::endl;

printf("\r\n\r\n");

/\*\*练习1.5

将所有的输出操作放在一条语句中

\*\*/

std::cout<<"the sum of num1 + num2 is "<<num1+num2<<" and result of num1\*num2 is "<< num1\*num2<<"\r\n"<<std::endl;

getchar();

while(1);

return 0;

}

练习1.6

程序段是不合法的，因为<<运算符的用法错误，<<左侧为流对象，右侧为流输出内容，流输出可以有多个对象，但是不能用分号将其隔开，否则编译器会认为语句已经结束

修改：

代码修改为

Std::cout<<”The sum of ”<<v1<<”and”<<v2<<”is”<<v1+v2<<std::endl;

程序输出v1 和v2的和及对应字符串

/\*\*\*\*\*\*\*\*\*\*88

\*\*\* 2019年3月24日14:55:32

1.4.1节练习

while语句实现循环控制

练习1.9

练习1.0

练习1.11

\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

int main(void)

{

//练习1.9 求50到100的循环加和

int num = 50,sum = 0;

while(num<101)

{

sum += num;

num++;

}

std::cout<<"sum = "<<sum<<"\r\n"<<std::endl;//结果为3825

//练习1.10 递减

num = 10;

while(num>=0)

{

std::cout<<" "<<num<<" "<<std::endl;

num--;

}

//练习1.11 输入2个整数，输出二者之间的所有整数

int innum1,innum2;

std::cout<<" please input two increasing integer"<<std::endl;

std::cin>>innum1>>innum2;

if(innum1 > innum2)

{std::cout<<" input error! \r\n please input two increasing integer!"<<std::endl;}

else

{

num = innum1;

while(num<innum2)

{

std::cout<<num<<std::endl;

num++;

}

}

while(1);

return 0;

}

/\*\*\*\*\*\*\*\*\*\*88

\*\*\*

2019年3月24日15:23:18

1.4.2节练习

for循环控制流练习

练习1.12

练习1.13

练习1.14

\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

int main(void)

{

/\*\*\*

练习1.12

描述for循环的功能，并求sum值

\*\*\*/

int sum = 0;

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

sum+=i;

std::cout<<"sum = "<<sum<<"\r\n"<<std::endl;

/\*\*\*\*

\*\* 功能：循环求整数-100到100的和 sum输出值 0

\*\*\*\*/

//练习1.13

//练习1.13.1 求50到100的循环加和 for实现

sum = 0;

for(int i=50;i<101;i++)

{

sum += i;

}

std::cout<<"sum of 50 to 100 inclusive is "<<sum<<"\r\n"<<std::endl;//结果为3825

//练习1.13.2 递减

int num = 10;

for(int i =10;i>=0;i--)

{

std::cout<<i<<std::endl;

}

//练习1.13.2 输入2个整数，输出二者之间的所有整数

int innum1,innum2;

std::cout<<" input two increasing integer"<<std::endl;

std::cin>>innum1>>innum2;

if(innum1 > innum2)

{std::cout<<" input error! u idiot! \r\n please input two increasing integer!"<<std::endl;}

else

{

for(int i = innum1;i<=innum2;i++)

std::cout<<i<<std::endl;

}

//

/\*\*\*\*\*\*\*\*\*\*\*\*\*

//练习1.14

在for循环中，循环控制变量的初始化和修改都放在语句头部分，形式较简洁，且特别适用于循环次数已知的情况

在while循环中，循环控制变量一般在循环前初始化，形式没有for语句简洁，但是适用于循环次数未知的情况

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

while(1);

return 0;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* 2019年3月24日16:03:34

1.4.4 练习

\*\* 要点：if语句

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

//程序功能：记录连续输入整数中重复出现的数并输出

int main(void)

{

int currVal =0,val = 0;

if(std::cin>>currVal) //输入不合法就不执行

{

int cnt = 1; //计数

while(std::cin>>val)

{

if(val == currVal)++cnt; //记录重复的值

else

{

std::cout<<currVal<<" occurs "<< cnt << " times " <<std::endl;

currVal = val;

cnt = 1;

}

}

std::cout<<currVal<<" occurs "<< cnt << " times " <<std::endl;

}

while(1);

return 0;

}

1.17：输入值都相等，程序会只输出一次 xx occurs x times 就结束；若值全不相等，程序输出xx occurs 1 times ，次数与输入的数的个数相同

/\*\*\*\*\*\*\*\*\*\*88

\*\*\*

2019年3月24日16:18:50

1.4.4节练习

练习1.19

\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

int main(void)

{

//练习1.19 输入2个整数，输出二者之间的所有整数

//修正逻辑以使输入不必升序

int innum1,innum2;

std::cout<<" input two integer"<<std::endl;

std::cin>>innum1>>innum2;

if(innum1 > innum2)

{

int temp;

temp = innum2;

innum2 = innum1;

innum1 = temp;

}

for(int i = innum1;i<=innum2;i++)

std::cout<<i<<std::endl;

while(1);

return 0;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*

2019年3月24日16:33:47

1.5节练习

书店程序

练习1.21 1.22 1.23 1.24

关键：类

\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

#include "Sales\_item.h"

#define work1\_23 //编译开关

int main(void)

{

//课本例程 & 练习1.21

//功能：计算两个类对应成员的和

#ifdef work1\_21

Sales\_item item1,item2;

std::cin>> item1>>item2;

if(item1.isbn() == item2.isbn())

{

std::cout<<item1+item2<<std::endl;

//return 0;

}

else

{

std::cerr << "Data must refer to same ISBN"<<std::endl;

//return -1;

}

//以上代码对应练习1.21 实际为课本例程

#endif

//练习1.22 读取多个具有相同isbn的销售记录，输出所有记录和

#ifdef work1\_22

Sales\_item item1,item2;

if (std::cin>> item1)

{

int cnt = 1; //计数

while(std::cin>> item2)

{

if(item1.isbn() == item2.isbn())

{

cnt ++;

item1 += item2;

}

else

std::cerr << "Data must refer to same ISBN"<<std::endl;

}

std::cout<<"total record is "<<cnt<<"\r\n"<<item1<<std::endl;

}

#endif

#ifdef work1\_23

//练习2.23与2.24

//此程序其实有bug，必须连续输入相同的条目才可以统计到相同的书本信息，而且输入和输出混在一起非常难看

Sales\_item item1,item2;

if (std::cin>> item1)

{

int cnt = 1; //计数

while(std::cin>> item2)

{

if(item1.isbn() == item2.isbn())

{

cnt ++;

item1 += item2;

}

else

{

std::cout<<"total record of "<< item1.isbn()<<" is "<<cnt<<"\r\n"<<item1<<std::endl;

item1 = item2;

cnt = 1;

}

}

std::cout<<"total record of "<< item1.isbn()<<" is "<<cnt<<"\r\n"<<item1<<std::endl;

}

#endif

while(1);

}

/\*\*\*

\*\* 2019年3月26日19:32:03

练习2.3

\*

\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

int main(void)

{

unsigned u = 10,u2 = 42; //输出

std::cout << u2 - u << std::endl; //30

std::cout << u-u2 << std::endl; //很大的一个数

int i = 10,i2 = 42;

std::cout<<i2-i<<std::endl; //30

std::cout<<i-i2<<std::endl; //-30

std::cout<<i-u<<std::endl; //0

std::cout<<u-i<<std::endl; //0

while(1);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* 2019年3月26日20:02:38

\*\* 练习2.40 练习2.41 练习2.42

\*\* 自行编写一个头文件，定义类，重写输出书本销售记录的程序

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

#include "Sales\_data.h"

using namespace std;

#define work1\_23

int main(void)

{

#ifdef work1\_21

Sales\_data item1,item2;

double avg;

cin >> item1.bookNo >> item1.units\_sold >>item1.price ;

item1.revenue = item1.units\_sold\*item1.price ;

cin >> item2.bookNo >> item2.units\_sold >> item2.price;

item2.revenue = item2.units\_sold\*item2.price ;

if(item1.bookNo == item2.bookNo)

{

item2.units\_sold += item1.units\_sold;

item2.revenue += item1.revenue;

avg = item2.revenue / item2.units\_sold;

std::cout<<item2.bookNo<< item2.units\_sold <<item2.revenue<<avg<<std::endl;

//return 0;

}

else

{

std::cerr << "Data must refer to same ISBN"<<std::endl;

//return -1;

}

//以上代码对应练习1.21 实际为课本例程

#endif

//练习1.22 读取多个具有相同isbn的销售记录，输出所有记录和

#ifdef work1\_22

Sales\_data item1,item2;

double avg;

if (cin >> item1.bookNo >> item1.units\_sold >>item1.price)

{

int cnt = 1; //计数

item1.revenue = item1.units\_sold\*item1.price ;

while(cin >> item2.bookNo >> item2.units\_sold >>item2.price)

{

if(item1.bookNo == item2.bookNo)

{

cnt ++;

item2.revenue = item2.units\_sold\*item2.price ;

item1.units\_sold += item2.units\_sold;

item1.revenue += item2.revenue;

}

else

std::cerr << "Data must refer to same ISBN"<<std::endl;

}

avg = item1.revenue /item1.units\_sold;

std::cout<<"total record is "<<cnt<<"\r\n"<<item1.bookNo<<item1.units\_sold<<item1.revenue<<avg<<std::endl;

}

#endif

#ifdef work1\_23

//练习2.23与2.24

//此程序其实有bug，必须连续输入相同的条目才可以统计到相同的书本信息,而且输入和输出混在一起非常难看

Sales\_data item1,item2;

if (cin >> item1.bookNo >> item1.units\_sold >>item1.price)

{

int cnt = 1; //计数

item1.revenue = item1.units\_sold\*item1.price ;

while(cin >> item2.bookNo >> item2.units\_sold >>item2.price)

{

if(item1.bookNo == item2.bookNo)

{

cnt ++;

item2.revenue = item2.units\_sold\*item2.price ;

item1.units\_sold += item2.units\_sold;

item1.revenue += item2.revenue;

}

else

{

std::cout<<"total record is "<<cnt<<"\r\n"<<item1.bookNo<<item1.units\_sold<<item1.revenue<<std::endl;

item1 = item2;

cnt = 1;

}

}

std::cout<<"total record of "<< item1.bookNo<<" is "<<cnt<<"\r\n"<<item1.bookNo<<item1.units\_sold<<item1.revenue<<std::endl;

}

#endif

while(1);

}

程序段使用了条件编译，还是比较习惯叫结构体而不是类；