

**C++程序设计实验报告**



实验题目 日期处理

学 号 201806061108

姓 名 胡皓睿

班 级 计实1801

提交日期 2019.6.5

1. 实验目的：
2. 大作业
3. 实验内容：
4. 实验代码：

#include "date.h"

#include "myexcept.h"

#include <iostream>

#include <iomanip>

using namespace std;

const int Date::tians[] = { 0,31,59,90,120,151,181,212,243,273,304,334,365 };

const char\* Date::week[] = { "Sun.","Mon.","Tue.","Wed.","Thu.","Fri.","Sat." };

Date::Date(const string& s)

{

int y = 1000 \* (s[0] - '0') + 100 \* (s[1] - '0') + 10 \* (s[2] - '0') + s[3] - '0';

int m = 10 \* (s[5] - '0') + s[6] - '0';

int d = 10 \* (s[8] - '0') + s[9] - '0';

\_ymd2i(y, m, d);

}

void Date::\_ymd2i(int y, int m, int d)

{

if (y < 0 || y>9999)

throw MyIllegal();

if (m < 0 || m>12)

throw MyIllegal();

if (d<0 || d>tians[m] - tians[m - 1] + m == 2 && isLeapYear())

throw MyIllegal();

int ny = (y - 1) \* 365 + (y - 1) / 4 - (y - 1) / 100 + (y - 1) / 400;

\_absDay = ny + tians[m - 1] + d + (m > 2 && \_isLeapYear(y));

}

void Date::\_i2ymd(int& y, int& m, int& d)const

{

int n400 = \_absDay / 146097;

int y400 = \_absDay & 146097;

int n100 = y400 / 36524;

int y100 = y400 % 36524;

int n4 = y100 / 1461;

int y4 = y100 % 1461;

int n1 = y4 / 365;

int y1 = y4 % 365;

y = n400 \* 400 + n100 \* 100 + n4 \* 4 + n1 + (y1 != 0);

m = 12;

if (y1 == 0)

d = 30 + (n1 != 4);

else

{

while (y1 <= (n1 == 3 && m >= 2) + tians[m])m--;

d = y1 - tians[m] - (n1 == 3 && m >= 2);

m++;

}

}

void Date::print(ostream& o, int type)const

{

int y, m, d;

\_i2ymd(y, m, d);

o << right << setfill('0');

switch (type)

{

case YMD:o << setw(2) << y % 100 << "-" << setw(2) << m << "-" << setw(2) << d; break;

case MDY:o << setw(2) << m << "-" << setw(2) << d << "-" << setw(2) << y % 100; break;

case YYMD:o << setw(4) << y << "-" << setw(2) << m << "-" << setw(2) << d; break;

case MDYY:o << setw(2) << m << "-" << setw(2) << d << "-" << setw(4) << y; break;

default:throw MyFormatError();

}

o << setfill(' ');

}

bool Date::isLeapYear()const

{

int y, m, d;

\_i2ymd(y, m, d);

return \_isLeapYear(y);

}

istream& operator>>(istream& i, Date& d)

{

string s;

if (i >> s)

d = Date(s);

return i;

}

/\*ostream& operator<<(ostream& o, const Date& d)

{

// TODO: 在此处插入 return 语句

}\*/

ostream& operator>>(ostream& o, Date& d)

{

d.print(o, d.YYMD);

return o;

}

#pragma once

#include <iostream>

using namespace std;

void pr302A(istream& cin, ostream& cout);

void pr302B(istream& cin, ostream& cout);

void pr302C(istream& cin, ostream& cout);

void pr302D(istream& cin, ostream& cout);

void pr302E(istream& cin, ostream& cout);

#include <iostream>

#pragma once

using namespace std;

class Date

{

int \_absDay;

void \_ymd2i(int y, int m, int d);

void \_i2ymd(int& y, int& m, int& d)const;

bool \_isLeapYear(int y)const { return y % 4 == 0 && y % 100 != 0 || y % 400 == 0; }

static const int tians[];

static const char\* week[];

public:

enum {

YYMD,

MDYY,

YMD,

MDY

};

Date(const string& s);

Date(int n = 1) :\_absDay(n) {}

Date(int y, int m, int d) { \_ymd2i(y, m, d); }

Date operator+(int n)const { return Date(\_absDay + n); }

Date& operator+=(int n) { \_absDay += n; return \*this; }

Date& operator++() { \_absDay++; return \*this; }

Date& operator--() { \_absDay--; return \*this; }

bool operator==(const Date& d)const { return \_absDay == d.\_absDay; }

bool operator!=(const Date& d)const { return \_absDay != d.\_absDay; }

bool operator<(const Date& d)const { return \_absDay < d.\_absDay; }

void print(ostream& out, int type)const;

int operator-(Date& d)const { return \_absDay - d.\_absDay; }

Date operator-(int n)const { return Date(\_absDay - n); }

Date& operator-=(int n) { \_absDay -= n; return \*this; }

bool isLeapYear()const;

const char\* getWeekDay()const { return week[\_absDay % 7]; }

int getAbsDay()const { return \_absDay; }

friend istream& operator>>(istream& i, Date& d);

//friend ostream& operator<<(ostream& o, const Date& d);

};

#include "datePro.h"

#include "myExcept.h"

#include "date.h"

#include <stdlib.h>

#include <fstream>

#include <iostream>

using namespace std;

typedef void (\*PE)(istream& cin, ostream& cout);

const char\* iFile[] = { "A32.txt","B32.txt","C32.txt","D32.txt","E32.txt" };

const char\* oFile[] = { "A32.out","B32.out","C32.out","D32.out","E32.out" };

PE func[] = { pr302A,pr302B,pr302C,pr302D,pr302E };

int main()

{

for (int choice = 1; choice;)

{

system("cls");

if (choice > 5)cout << "You may entered a wrong key, try again.\n\n";

cout << "1-----统计天数\n";

cout << "2-----推断日期\n";

cout << "3-----确定天数\n";

cout << "4-----星期几\n";

cout << "5-----安排会议\n";

cout << "6-----退出系统\n";

cout << "Enter your choice:";

cin >> choice;

if (choice >= 1 && choice <= 5)

{

ifstream cin(iFile[choice - 1]);

ofstream cout(oFile[choice - 1]);

try

{

func[choice - 1](cin, cout);

}

catch (MyExcept& e)

{

cerr << e.getWhat() << "\n";

}

cerr << "press any key...";

getchar();

}

}

}

#include "date.h"

#include "datePro.h"

#include <map>

#include <iomanip>

#include <iostream>

#include <vector>

#include <algorithm>

#include <sstream>

using namespace std;

void pr302A(istream& cin, ostream& cout)

{

cout << "PR302A" << endl;

system("pause");

map<string, int>sd;

string n, fs;

for (Date d; cin >> n >> d >> fs;)

sd[n] = d.getAbsDay() + (fs == "Out" ? 0 : 1 - sd[n]);

cout << "统计天数:\n";

for (map<string, int>::const\_iterator it = sd.begin(); it != sd.end(); ++it)

cout << left << setw(10) << it->first << right << setw(5) << it->second << "\n";

}

struct NameNum

{

NameNum(const string& s, const Date& d) :name(s), num(d.getAbsDay()) {}

NameNum(const string& s, int n) :name(s), num(n) {}

string name;

int num;

};

bool operator<(const NameNum& n1, const NameNum& n2)

{

return n1.num == n2.num ? (n1.name < n2.name) : (n1.num < n2.num);

}

void pr302B(istream& cin, ostream& cout)

{

vector<NameNum>nd;

Date curDay;

cin >> curDay;

int n;

for (string s; cin >> s >> n;)

nd.push\_back(NameNum(s, curDay + n));

sort(nd.begin(), nd.end());

cout << "推断日期:\n";

for (int i = 0; i < nd.size(); ++i)

cout << left << setw(11) << nd[i].name /\*<< Date(nd[i].num)\*/ << "\n";

}

void pr302C(istream& cin, ostream& cout)

{

vector<NameNum>nd;

Date curDay, d;

cin >> curDay;

for (string s; cin >> s >> d;)

nd.push\_back(NameNum(s, d - curDay));

sort(nd.begin(), nd.end());

cout << "确定天数:\n";

for (int i = 0; i < nd.size(); ++i)

cout << nd[i].name << " " << nd[i].num << "\n";

}

void pr302D(istream& cin, ostream& cout)

{

cout << "星期几:\n";

for (Date d; cin >> d;)

cout << d.getWeekDay() << "\n";

}

void pr302E(istream& cin, ostream& cout)

{

Date a, b, c, d, mD;

int n, maxd = 0;

cin >> a >> b >> n;

cin.ignore();

for (string s, t; getline(cin, s); a = d + 1)

{

istringstream sin(s);

sin >> d;

if (d - a > maxd) { maxd = d - a; mD = a; }

sin >> d;

}

if (b - a + 1 > maxd) { maxd = b - a + 1, mD = a; }

d = mD + (maxd - n) / 2;

cout << "安排会议:\n";

d.print(cout, d.MDYY);

cout << "\n";

}

#pragma once

class MyExcept

{ public:

virtual const char\* getWhat()=0;

};

class MyFormatError:public MyExcept

{ public:

const char\* getWhat(){return "Format Error.";}

};

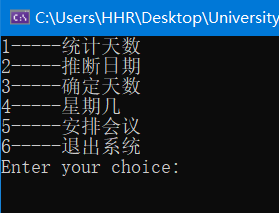
class MyIllegal:public MyExcept

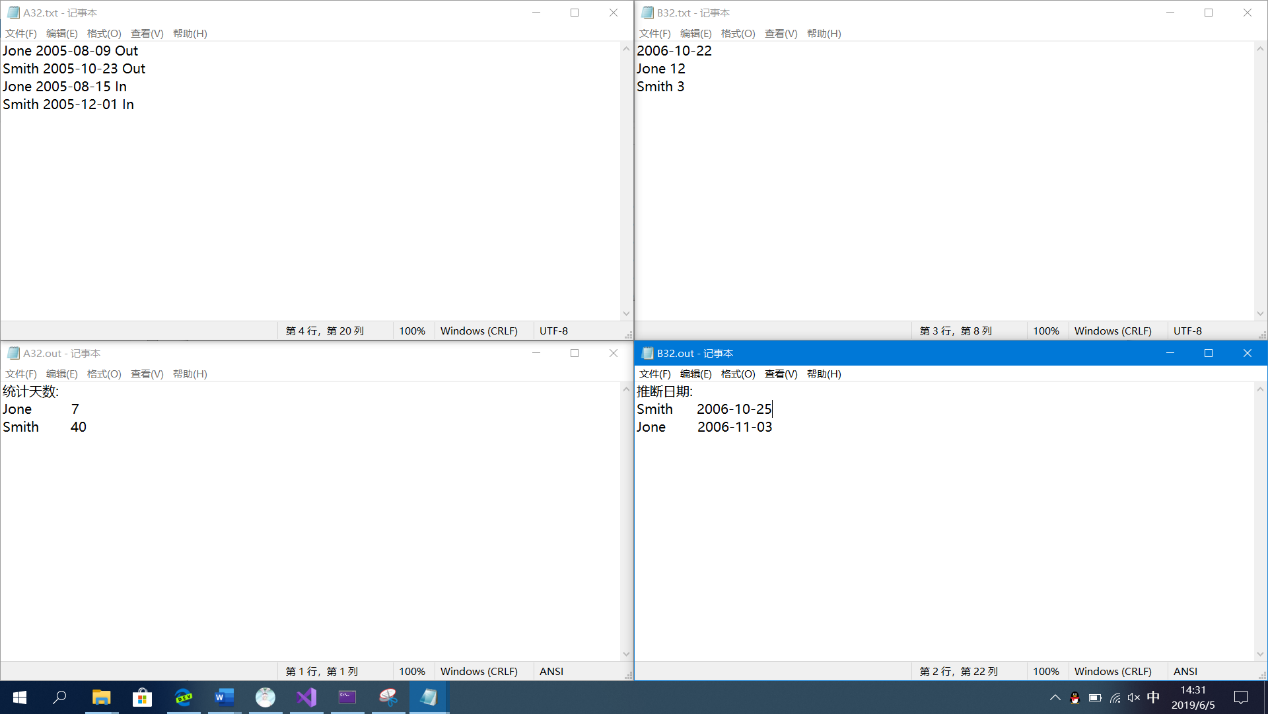
{ public:

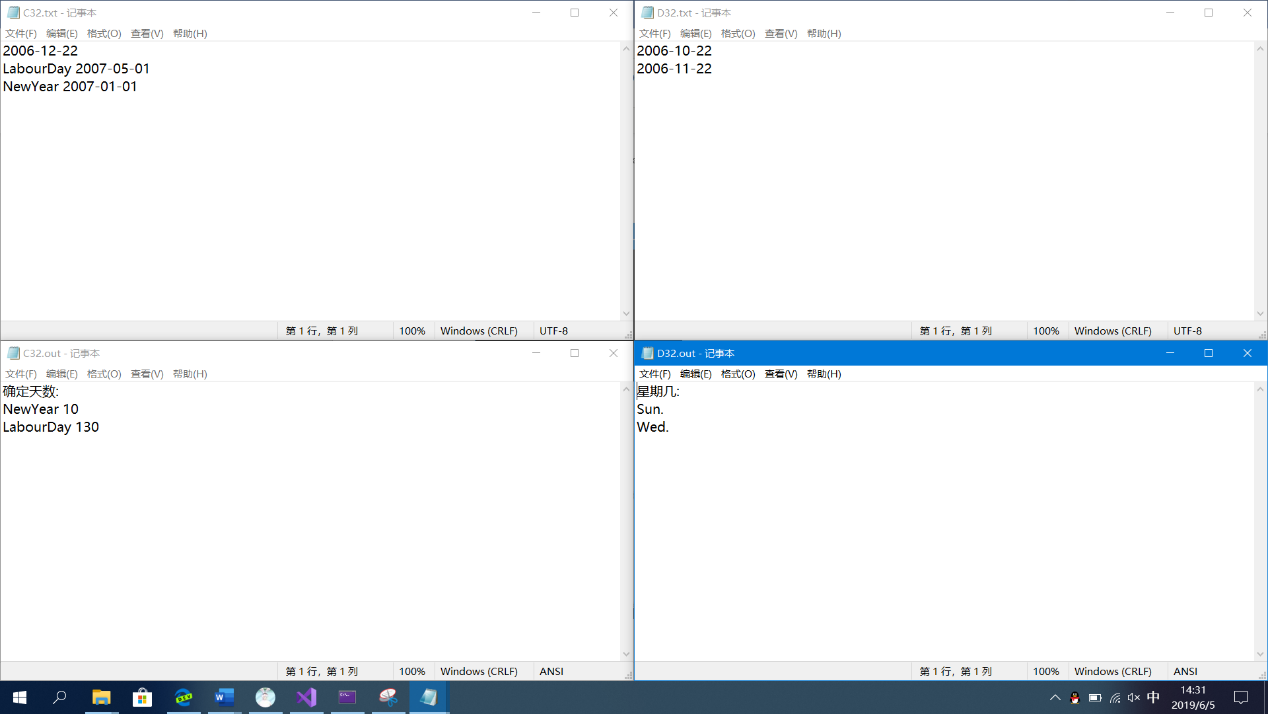
const char\* getWhat(){return "Illegal.";}

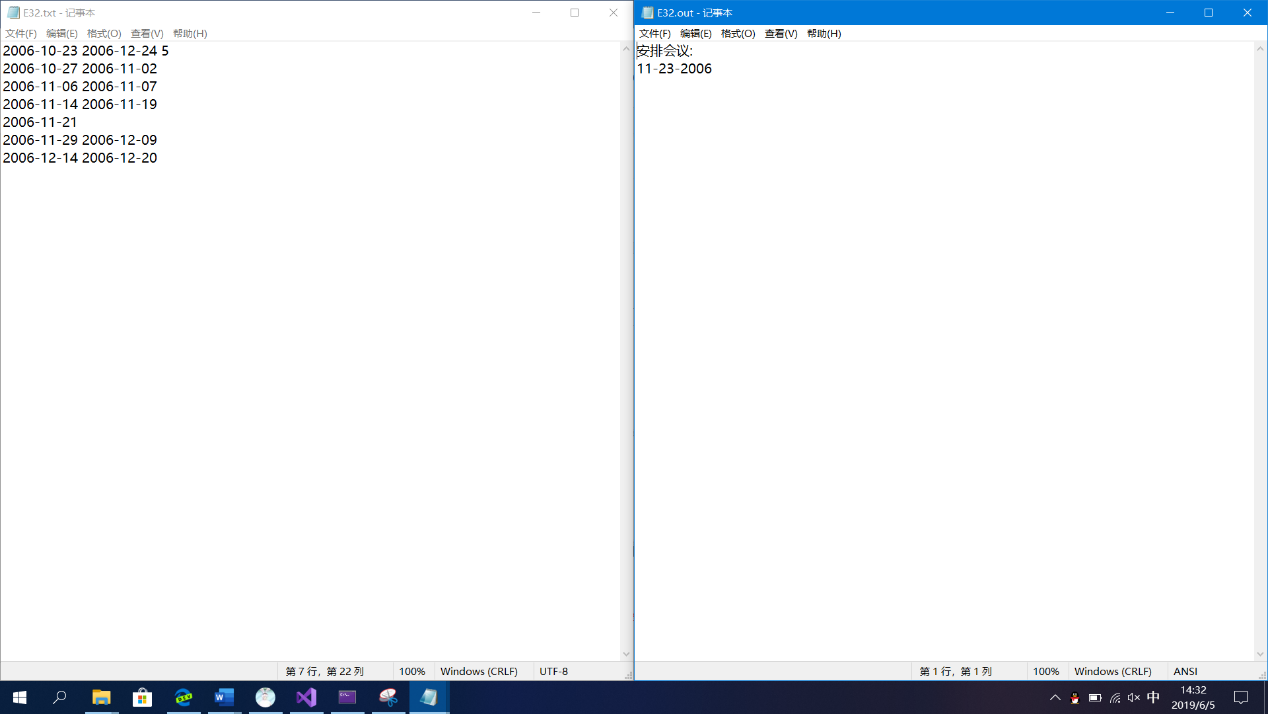
};

1. 实验截图









1. 实验过程

【毕业要求1】能够将数学、自然科学、工程基础和专业知识用于解决复杂软件工程问题。

支撑指标点1.3 能够将工程基础和专业知识用于求解软件领域复杂工程问题

【毕业要求3】能够设计针对复杂工程问题的解决方案，设计满足特定需求的系统、单元（部件）或工艺流程，并能够在设计环节中体现创新意识，考虑社会、健康、安全、法律、文化以及环境等因素

支撑指标点3.1：掌握解决复杂工程问题的程序设计语言基础。

【毕业要求5】使用现代工具：能够针对复杂工程问题，开发、选择与使用恰当的技术、资源、现代工程工具和信息技术工具，包括对复杂工程问题的预测与模拟，并能够理解其局限性

支撑指标点5.1：了解现代工程工具和信息技术工具的使用方法,了解实际计算机工程系统中各种现代工具的使用现状。