

OOP 第四周作业文档

2019010175 孔瑞阳 土木 92

一、项目信息

1、功能说明

先询问是否使用系统当前时间，如果不使用输入一个合法的年月日（公历）。

再输入一个整数 n 。

输出输入的日期经过 n 天后的日期。

2、软件构件介绍

文件	功能介绍
CP_Integer.h/cpp	实现整数的输入和操作
CP_Calendar.h/cpp	日历类，实现日历内部的操作
CP_CalendarApplication.h/cpp	日历的交互操作
CP_CalendarTest.h/cpp	日历类的测试程序
calendarMain.cpp	主程序

3、测试环境

CPU	Intel(R) Core(TM)i7-9750H CPU @ 2.6Ghz 6 核 12 线程
GPU	NVIDIA GeForce RTX2070
RAM	DDR4 16G+16G
Operating System	Microsoft Windows 版本 1809 (OS 内部版本 17763.1098)
Compiler	MSVC++ 14.24

二、模型

1、输入与输出

输入和输出采用 C++ 标准库中的输入输出流 `cin` 和 `cout` 进行。

在输入过程中，

- 1、没有公历 0 年，若年份输入 0 则重新输入。
- 2、月份的范围是 1-12，如果输入的月份不在这个范围内重新输入。
- 3、根据之前输入的年份和月份，可以推出这个月的天数，得出日期的范围。如果输入的日期不在计算的范围内则重新输入。

对于输出，

- 1、如果年份 x 是正数，则输出公元 x 年
- 1、如果年份 x 是负数，则输出公元前 $|x|$ 年

2、公元历法

参考链接: https://en.wikipedia.org/wiki/Gregorian_calendar

月份	1	2	3	4	5	6	7	8	9	10	11	12
天数	31	28(闰年 29)	31	30	31	30	31	31	30	31	30	31

闰年规则: 如果年份是 4 的倍数是闰年, 但是如果是 100 的倍数就不是闰年, 其中又如果是 400 的倍数是闰年。

3、日期计算

首先, 将每一个日期和一个整数一一对应, 这个整数代表公元前 1 年 12 月 31 日到这个日期的天数。比如, 公元 1 年 1 月 1 日对应 1, 公元前 1 年 12 月 31 日对应 0, 公元前 1 年 12 月 30 日对应 -1。

那么我们只要实现两个操作:

- 1、把一个日期转换为一个整数。
- 2、把一个整数转换成一个日期。

把输入的日期转换为整数之后, 进行“n 天后”的这个操作只要直接做加法就可以了, 再把 +n 之后的整数转换为日期。

具体来说:

(1) 判断闰年

对于公元后 x 年, 则根据闰年规则计算。

对于公元前 x 年, 将 x-1 进行上述操作即可。(如 B. C. 1 是闰年, B. C. 101 不是, B. C. 401 是闰年)

(2) 日期转化为整数

第一步: 计算本年份之前的年份有多少天

由于公历严格按照 400 年为一个周期, 每 400 年根据容斥原理有 $100 - 4 + 1 = 97$ 个闰年, 所以 400 年的总天数永远是 $365 * 400 + 97 = 146097$ 天。

那么我们可以先计算 $1 \sim \left\lfloor \frac{y-1}{400} \right\rfloor * 400$ 年的天数 (公元前直接转化为绝对值即可), 总

天数就是 $\left\lfloor \frac{y-1}{400} \right\rfloor * 146097$, 再从第 $\left\lfloor \frac{y-1}{400} \right\rfloor * 400 + 1$ 年循环到第 y-1 年, 每遇到一年通过判

断这一年是否是闰年来计算当年的天数。可以看出这个循环不超过 400 次。

第二步: 计算本月份之前的月份有多少天

如果是公元后, 就从 1 月循环到 m-1 月, 否则从 m+1 月循环到 12 月。如果遇到 2 月, 通过判断当年是不是闰年来决定天数。

第三步: 计算本月份有多少天

如果是公元后, 那么日期 d 直接是天数。

如果是公元前, 根据年份和月份计算出当月的天数, 那么结果就是这个月剩下的天数。

(3) 整数转化为日期

第一步：计算年份

首先，还是按照 400 年的周期先计算自己应该处于哪一个 400 年，初始化年份为

$$\left\lfloor \frac{D-1}{146097} \right\rfloor * 400 + 1 \quad (\text{如果是公元前则取负}), \text{并减去相应的天数。其次进入一个循环, 如}$$

果当前剩余的天数比现在的年份还要大的话，就减去当前年份的天数，并把当前年份+1（如果是公元前则-1），直到当前剩余的天数小于本年度的天数。可以看出这个循环不会超过 400 次。

第二步：计算月份

如果是公元后则从 1 月开始向后循环，如果是公元前则从 12 月开始向前循环。

每循环到一个月份，计算这个月份有多少天，如果当前剩余的天数大于这个月的天数，则减去这个月的天数继续循环，否则当前循环到的月份就是最终的月份。

第三步：计算日期

如果是公元前，则当前剩余的天数 D 就是日期。

如果是公元后，那么从这个月的最后一天往前推 D-1 天即可。

三、单元测试

1、设置系统时间的测试

测试时间：2020 年 3 月 15 日

是否选择系统时间选择 yes，输出为 日期是：公元 2020 年 3 月 15 日

2、输入单元的测试

对于输入年份

(1) 等价类划分

- ① 输入的不是整数
- ② 输入为 0
- ③ 输入一个不为 0 的整数

(2) 测试：

等价类	选取案例	输出结果（下一步操作）
等价类①	HelloWorld!	您输入的不是整数，请输入一个整数：
	1919.810	您输入的不是整数，请输入一个整数：
等价类②	0	年份不能为 0，请重新输入
等价类③	1919810	请输入月份
	-114514	请输入月份
	1	请输入月份
	-1	请输入月份

对于输入月份

(1) 等价类划分

- ① 输入的不是 1-12 的正整数。
- ② 输入的是 1-12 的正整数。

(2) 测试:

等价类	选取案例	输出结果(下一步操作)
等价类①	HappyWorld!	您输入的不是整数, 请输入一个整数:
	114.514	您输入的不是整数, 请输入一个整数:
	-10	您输入的月份不在 1-12 的范围内, 请重新输入
	100	您输入的月份不在 1-12 的范围内, 请重新输入
等价类②	1	请输入日期(用 1-31 的整数表示)
	4	请输入日期(用 1-30 的整数表示)
	9	请输入日期(用 1-30 的整数表示)
	10	请输入日期(用 1-31 的整数表示)
	2 (2020 年)	请输入日期(用 1-29 的整数表示)
	2 (-2021 年)	请输入日期(用 1-29 的整数表示)
	2 (1 年)	请输入日期(用 1-28 的整数表示)

对于输入日期

(1) 等价类划分

- ① 输入的不是整数
- ② 输入的日期不在指定范围内
- ③ 输入的日期合法

(2) 测试:

等价类	选取案例	输出结果(下一步操作)
等价类①	HappyWorld!	您输入的不是整数, 请输入一个整数:
	114.514	您输入的不是整数, 请输入一个整数:
等价类②	31 (4 月)	您输入的日期不在 1-30 的范围内, 请重新输入
	-1 (4 月)	您输入的日期不在 1-30 的范围内, 请重新输入
	29 (1 年 2 月)	您输入的日期不在 1-28 的范围内, 请重新输入
	0 (7 月)	您输入的日期不在 1-31 的范围内, 请重新输入
等价类③	29 (-1 年 2 月)	请输入一个整数表示 n 天后
	31 (5 月)	请输入一个整数表示 n 天后
	1 (3 月)	请输入一个整数表示 n 天后
	31 (8 月)	请输入一个整数表示 n 天后

对于输入 n 天后

由于只涉及到整数的输入, 在上述测试中已经得到验证。

3、设置、输出时间；获取年月日单元的测试

输入时间	showYear	showMonth	showDay	outputCalendar
2020/3/15	2020	3	15	日期是：公元 2020 年 3 月 15 日
-1/6/1	-1	6	1	日期是：公元前 1 年 6 月 1 日
2017/3/14	2017	3	14	日期是：公元 2017 年 3 月 14 日
-124353/2/29	-124353	2	29	日期是：公元前 124353 年 2 月 29 日

4、日期与天数转换单元的测试

测试单元：CP_CalendarTest.h/cpp

第一步：利用在线日历网站进行日期转换为天数的测试（公元后）。

日历网站：<http://tools.jb51.net/static/api/datecalc/index.html>

将起始日期调成 0001-01-01，将结果+1 即可。

测试结果：

日期	日历网站	程序运行结果
9999-12-31	3652059	3652059
0001-01-01	1	1
0114-05-14	41406	41406
1919-08-10	700756	700756
2020-03-14	734798	734798

两者相同，初步测试成功。

第二步：利用相互转换自动测试。

1、不断地随机一个天数，将其转换为日期，再转换为天数，判断是否相同。

2、不断地随机一个日期，将其转换为天数，再转换为日期，判断是否相同。

每分钟可以进行上千万次运算，在经过 10 分钟上亿次运算之后依然没有出现错误，可以判断基本正确。

5、n 天后日期的测试

具体的实现正确性已经在 4、日期与天数转换单元测试 中得到验证。

所以仅验证特殊/边界情况。

输入日期	n	程序运行结果
1/1/1	-2	n 天后的日期是：公元前 1 年 12 月 30 日
-1/12/31	10	n 天后的日期是：公元 1 年 1 月 10 日
-1/12/31	2147483647	n 天后的日期是：公元 5879611 年 7 月 11 日
-1/12/31	-2147483647	n 天后的日期是：公元前 5879611 年 6 月 23 日
-1/12/31	-2147483648	n 天后的日期是：公元 5879599 年 12 月 3876 日
1/1/1	2147483647	n 天后的日期是：公元 5879599 年 12 月 3876 日
2/1/1	2147483647	n 天后的日期是：公元前 5879610 年 6 月 22 日

通过测试，特殊情况（公元前后转换）无误，但是由于 int totalDays 范围的限制，能表示的日期为：公元前 5879611 年 6 月 23 日 - 公元 5879611 年 7 月 11 日。