目标：

编写C++程序来达到：设计日历类，它的数据包括年、月、日。要求通过面向对象技术保证年可以是任意的整数，月只能介于1和12之间，日只能1和31之间且应当符合公元历法（也称为公历或阳历）。要求通过日历类的实例对象可以设置/获取/输出其所记录的年月日，并且可以计算给定n天之后的年月日。这里n是整数，即可能是正整数，也可能是负整数，甚至是0。并且用正整数年份表示公元后的年份，负整数的绝对值表示公元前多少年，求解n天之后的年月日。

原理：

构建Calendar类如下：

class Calendar

{

private:

long long year;

int month;

int day;

long long All\_Days;

//这是在计算中使用的代表日期的变量，在这里用0代表1年1月1日，用-1代表公元前1年12月31日

void Calculate\_From\_All\_Days();

void Calculate\_To\_All\_Days();

public:

Calendar();

~Calendar(){}

int \* SetDate(long long year, int month, int day);

//通过调用该函数实现输入设定日期的功能（返回值是为了在自动测试模块中使用测试，手动测试时只需当做void处理即可）。其中year可以为正数、负数，正数代表公元XX年，负数代表公元前XX年，不可输入0；如果输入的日期不存在会报错。

int \* SetDate(long long All\_Days);

void PrintDate();

//通过调用PrintDate函数来实现输出当前日期的功能。

int \* GetDesiredDate(int n);

//通过调用GetDesiredDate函数来实现输出当前日期后n天的日期的功能。其中n可以为正、负或者0。

static bool If\_Leap\_Year(long long This\_Year);

static int Total\_Days\_In\_Year(long long year);

static int Get\_Feb\_Plus\_Day(long long year);

static int Days\_In\_The\_Month(long long year, int month);

};

在构造函数中调用系统函数获取当前的日期并转化为北京时间（东八区），赋值给构造的对象。用SetDate函数进行对日期的赋值操作，用GetDesiredDate函数进行计算n天后的日期。在计算的过程中为了减少越界的可能性，采用往复计算的方法，通过对数据往返处理后与原数据对比来减少越界风险。

同时将判断闰年、获取每个月份的天数设定为静态函数，便于调用。

在手动测试过程中实现了循环输入数据测试，测试过程中可以检测输入的合法性。

在自动测试中，用到了系统库函数gmtime来进行时间到日期的转换，并将结果与程序运行结果比较。

**如何编程和运行：**

* 软件构件库部分：

构建所需的类和最基本的计算函数。Calendar类及其中的函数已经在上面说明。

* 构件库测试部分：

编写所需要的测试函数。

* 构件库应用部分：

编写main函数，在函数中提供选择自动和手动测试的功能。

结果：

以下测试数据均在64位操作系统下用VS 2017 测试得到。

自动测试程序运行结果：

------------------------------------------------------------

公元 1992年 8月 3日

正确！

公元 1992年 8月 8日

正确！

公元 1992年 8月 13日

正确！

公元 1992年 8月 18日

正确！

公元 1992年 8月 23日

正确！

……（省略中间部分的结果）

公元 2044年 7月 16日

正确！

公元 2044年 7月 21日

正确！

公元 2044年 7月 26日

正确！

公元 2044年 7月 31日

正确！

公元 2044年 8月 5日

正确！

共测试3800组数据，其中0组错误！

请按任意键继续. . .

------------------------------------------------------------

自动测试程序验证结果报告：

* 对于64位的计算机

在测试的过程中发现对于测试的数据没有发现错误的计算结果。

手动测试部分某次测试结果：

------------------------------------------------------------

输入 1 进行手动测试，输入 2 进行自动测试！

------------------------------------------------------------

1

公元 2017年 3月 25日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

31

公元 2017年 4月 25日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

3650

公元 2027年 3月 23日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

-31

公元 2017年 2月 22日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

3650000

公元 12010年 8月 7日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

-3650000

公元前 7978年 11月 12日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

4000

公元 2028年 3月 7日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

0

公元 2017年 3月 25日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

1

公元 2017年 3月 26日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

-1

公元 2017年 3月 24日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

2

请输入一个日期！按照 年 月 日 的顺序，中间用 空格 隔开！

2017 2 29

输入错误，请重新输入！

请输入一个日期！按照 年 月 日 的顺序，中间用 空格 隔开！

2016 2 29

公元 2016年 2月 29日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

2

请输入一个日期！按照 年 月 日 的顺序，中间用 空格 隔开！

2017 2 28

公元 2017年 2月 28日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

2

请输入一个日期！按照 年 月 日 的顺序，中间用 空格 隔开！

0 1 1

输入错误，请重新输入！

请输入一个日期！按照 年 月 日 的顺序，中间用 空格 隔开！

2

13 31

输入错误，请重新输入！

请输入一个日期！按照 年 月 日 的顺序，中间用 空格 隔开！

2

2 29

输入错误，请重新输入！

请输入一个日期！按照 年 月 日 的顺序，中间用 空格 隔开！

-1 2 29

公元前 1年 2月 29日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

2

请输入一个日期！按照 年 月 日 的顺序，中间用 空格 隔开！

3200 2 29

输入错误，请重新输入！

请输入一个日期！按照 年 月 日 的顺序，中间用 空格 隔开！

172800 2 29

公元 172800年 2月 29日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

31

公元 172800年 3月 31日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

-29

公元 172800年 1月 31日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

0

公元 172800年 2月 29日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

2

请输入一个日期！按照 年 月 日 的顺序，中间用 空格 隔开！

-32 1 1

公元前 32年 1月 1日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

1

公元前 32年 1月 2日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

1

输入n！

-1

公元前 33年 12月 31日

输入 1 计算n天之后的日期，输入 2 重新输入日期，输入其他字符退出！

程序验证结果报告：

* 对于64位的计算机（使用的是long long）

不正确的输入并不会导致程序出错，而是会提示输入错误，要求重新输入。

程序在判断闰年方面并未发现错误，在对负整数的测试中未发现错误。

结论：

程序能较好完成设计目的要求，在使用的过程中未发现错误。

MFC部分：

使用两种方法编写MFC日历：

1. 使用VS中内置的Calendar对象；
2. 使用前面设计的Calendar类直接制作MFC程序。其中可以输入指定年份、月份并且对错误输入进行屏蔽处理。同时可以用按钮对年份、月份进行选择，达到获取日历的效果。如果用输入的方式，在输入数据的同时就会改变日历的显示。