

SJTU-SE

# 测试报告

第六组

李 珊 516030910175

王梦瑶 516030910177

陈 诺 516030910199

胡雨奇 516030910257

## 目录

测试计划.....	2
1. 测试目的.....	2
2. 测试方法.....	2
3. 测试环境.....	2
4. 不同的功能与对应的测试计划.....	2
3.1 日期计算器.....	2
3.1.1 功能测试.....	2
3.1.2 等价类测试.....	3
3.1.3 边界值测试.....	5
3.1.4 小结.....	8
3.2 程序员计算器.....	9
3.2.1 功能测试.....	9
3.2.2 等价类测试.....	9
3.2.3 边界值测试.....	10
3.2.4 小结.....	16
3.3 科学计算器.....	16
3.3.1 功能测试.....	16
3.3.2 等价类测试.....	17
3.3.3 边界值测试.....	20
3.3.4 小结.....	24
3.4 标准计算器.....	25
3.4.1 功能测试.....	25
3.4.2 等价类测试.....	25
3.4.3 边界值测试.....	29
3.4.4 小结.....	32
4. 总结.....	32
5. 参考.....	32

# 测试计划

## 1. 测试目的

1. 联系和掌握黑盒测试的一般过程与步骤
2. 通过测试检验 Windows 内嵌计算器的可靠性

## 2. 测试方法

利用黑盒测试中的等价类测试法、边界值检查法、因果图法等进行测试，检验计算器是否实现了软件需求规格说明书中规定的功能。

## 3. 测试环境

系统：Windows 10

位数：64 位

## 4. 不同的功能与对应的测试计划

### 3.1 日期计算器

#### 3.1.1 功能测试

##### 1. GUI 测试

通过拖拽、伸缩 GUI 界面，证明了 GUI 具有一定的自适应性。在测试环境（像素大小为 2000x3000）下最小大小为 644x1002 像素，小于这个范围时计算器不能再缩小。最大大小为占满屏幕。计算器在最大、最小和其中的大小时，都能保持 GUI 组件正常显示。

通过点击最小化、最大化、关闭按钮、输入组件，证明计算器 GUI 的按钮、组件具有正常的功能。

## 2. 日期计算器功能

1. 计算日期间的相隔时间;
2. 添加天数;
3. 减去天数。

### 3.1.2 等价类测试

#### 1. 日期之间的间隔时间

编号	输入 1	输入 2	有效等价类	期望输出	实际输出
1	2019 年 2 月 28 日	2019 年 3 月 1 日	包含 2 月尾的一般非闰年	1 天	1 天
2	2020 年 2 月 28 日	2020 年 3 月 1 日	包含 2 月尾的一般闰年	2 天	2 天
3	1900 年 2 月 28 日	1900 年 3 月 1 日	包含 2 月尾, 年份为整除 4 且整除 100 的非闰年	1 天	1 天
4	2000 年 2 月 28 日	2000 年 3 月 1 日	包含 2 月尾, 年份为被 4 整除被 100 整除且被 400 整除的世纪闰年	2 天	2 天
5	2018 年 1 月 1 日	2019 年 1 月 1 日	输入 1 和输入 2 跨年	1 年或 365 天	1 年 356 天
6	2019 年 4 月 14 日	2019 年 5 月 14 日	输入 1 和输入 2 跨月	1 月或 30 天	1 月 30 天
7	2020 年 4 月 14 日	2019 年 4 月 14 日	输入 1 小于输入 2	提示错误或显示正确间隔	1 年 366 天
8	2019 年 4 月 14 日	2019 年 4 月 14 日	相同日期	0 天	相同日期
9	2020 年 5 月 01 日	2019 年 4 月 14 日	普通输入	1 年 0 月 13 天或 383 天	1 年, 2 周, 3 天 383 天

可以看到, 输出结果全部是正确的。虽然有一些测试用例的输出结果与期望输出有一些不同, 但是具有相同的语义信息 (例如第 9 条测试用例), 因此可以认为日期之间的相隔时间功能通过理论等价类测试。

## 2. 添加天数

编号	输入 1	输入 2	有效等价类	期望输出	实际输出
1	2019 年 2 月 28 日	1	包含 2 月尾, 年数为一般非闰年	2019 年 3 月 1 日	2019 年 3 月 1 日
1	2020 年 2 月 28 日	1	包含 2 月尾, 年数为一般闰年	2020 年 2 月 29 日	2020 年 2 月 29 日
2	1900 年 2 月 28 日	1	包含 2 月尾, 年数为整除 4 且整除 100 的非闰年	1900 年 3 月 1 日	1900 年 3 月 1 日
3	2000 年 2 月 28 日	1	包含 2 月尾, 年数为整除 4 且整除 100 且被 400 整除的世纪闰年	2000 年 2 月 29 日	2000 年 2 月 29 日
4	2018 年 12 月 31 日	1	添加天数跨年	2019 年 1 月 1 日	2019 年 1 月 1 日
5	2019 年 4 月 30 日	1	添加天数跨月	2019 年 5 月 1 日	2019 年 5 月 1 日
6	2019 年 4 月 30 日	0	未添加天数	提示未添加天数或 2019 年 4 月 30 日	2019 年 4 月 30 日

## 3. 减去天数

编号	输入 1	输入 2	有效等价类	期望输出	实际输出
1	2019 年 3 月 1 日	1	包含 2 月尾, 年数为一般非闰年	2019 年 2 月 28 日	2019 年 2 月 28 日
2	2020 年 3 月 1 日	1	包含 2 月尾, 年数为一般闰年	2020 年 2 月 29 日	2020 年 2 月 29 日
3	1900 年 3 月 1 日	1	包含 2 月尾, 年数为整除 4 且整除 100 的非闰年	1900 年 2 月 28 日	1900 年 2 月 28 日
4	2000 年 3 月 1 日	1	包含 2 月尾, 年数为整除 4 且整除 100 且被 400 整除的世纪闰年	2000 年 2 月 29 日	2000 年 2 月 29 日
5	2019 年 1 月 1 日	1	减去天数跨年	2018 年 12 月 31 日	2018 年 12 月 31 日
6	2019 年 5 月 1 日	1	减去天数跨月	2019 年 4 月 30 日	2019 年 4 月 30 日

7	2019 年 4 月 30 日	0	未减去天数	提示未减去天数或 2019 年 4 月 30 日	2019 年 4 月 30 日
---	-----------------	---	-------	--------------------------	-----------------

### 3.1.3 边界值测试

#### 1. 日期之间的相隔时间

- 输入的最小值：1601 年 1 月 1 日
- 输入的最大值：2550 年 12 月 31 日

因为 GUI 输入不支持输入 max+ 和 min-，所以这里只基于单故障假设做测试，不做健壮性测试。

编号	输入 1	输入 2	与边界值关系	期望输出	实际输出
1	2019 年 04 月 14 日	1601 年 01 月 01 日	<1nom, 2min>	152774 天	418 年, 3 月, 1 周, 6 天 152774 天
2	2019 年 04 月 14 日	1601 年 01 月 02 日	<1nom, 2min+>	152773 天	418 年, 3 月, 1 周, 5 天 152773 天
3	2019 年 04 月 14 日	2019 年 04 月 14 日	<1nom, 2nom>	0 天	相同日期
4	2019 年 04 月 14 日	2550 年 12 月 30 日	<1nom, 2max->	194204 天	531 年, 8 月, 2 周, 2 天 194204 天
5	2019 年 04 月 14 日	2550 年 12 月 31 日	<1nom, 2max>	194205 天	531 年, 8 月, 2 周, 3 天 194205 天
6	1601 年 01 月 01 日	2019 年 04 月 14 日	<1min, 2nom>	152774 天	418 年, 3 月, 1 周, 6 天 152774 天
7	1601 年 01 月 02 日	2019 年 04 月 14 日	<1min+, 2nom>	152773 天	418 年, 3 月, 1 周, 5 天 152773 天
8	2550 年 12 月 30 日	2019 年 04 月 14 日	<max-, 2nom>	194204 天	531 年, 8 月, 2 周, 2 天 194204 天
9	2550 年 12 月 31 日	2019 年 04 月 14 日	<1max, 2nom>	194205 天	531 年, 8 月, 2 周, 3 天 194205 天

#### 2. 添加天数

- 初始时间范围：[1601 年 1 月 1 日, 2550 年 12 月 31 日]
- 添加时间范围：[0 年 0 月 0 天, 999 年 999 月 999 天]

前文使用了单故障假设做测试，这里选用最坏情况测试。（同样由于 GUI 限制了输入，这里没有采用健壮最坏情况测试）

编号	输入 1	输入 2	与边界值关系	期望输出	实际输出
1	1601 年 01 月 01 日	0 年 0 月 0 天	<1min, 2min>	1601 年 01 月 01 日	1601 年 1 月 1 日
2	1601 年 01 月 01 日	0 年 0 月 1 天	<1min, 2min+>	1601 年 01 月 02 日	1601 年 1 月 2 日
3	1601 年 01 月 01 日	1 年 1 月 1 天	<1min, 2nom>	1602 年 2 月 2 日	1602 年 2 月 2 日
4	1601 年 01 月 01 日	999 年 999 天 998 天	<1min, 2max->	2685 年 12 月 24 日	2685 年 12 月 24 日
5	1601 年 01 月 01 日	999 年 999 天 999 天	<1min, 2max>	2685 年 12 月 25 日	2685 年 12 月 25 日
6	1601 年 01 月 02 日	0 年 0 月 0 天	<1min+, 2min>	1601 年 01 月 02 日	1601 年 01 月 02 日
7	1601 年 01 月 02 日	0 年 0 月 1 天	<1min+, 2min+>	1601 年 01 月 03 日	1601 年 01 月 03 日
8	1601 年 01 月 02 日	1 年 1 月 1 天	<1min+, 2nom>	1602 年 02 月 03 日	1602 年 02 月 03 日
9	1601 年 01 月 02 日	999 年 999 天 998 天	<1min+, 2max->	2685 年 12 月 25 日	2685 年 12 月 25 日
10	1601 年 01 月 02 日	999 年 999 天 999 天	<1min+, 2max>	2685 年 12 月 26 日	2685 年 12 月 26 日
11	2019 年 04 月 14 日	0 年 0 月 0 天	<1nom, 2min>	2019 年 4 月 14 日	2019 年 4 月 14 日
12	2019 年 04 月 14 日	0 年 0 月 1 天	<1nom, 2min+>	2019 年 4 月 15 日	2019 年 4 月 15 日
13	2019 年 04 月 14 日	1 年 1 月 1 天	<1nom, 2nom>	2020 年 5 月 15 日	2020 年 5 月 15 日
14	2019 年 04 月 14 日	999 年 999 天 998 天	<1nom, 2max->	3104 年 4 月 7 日	3104 年 4 月 7 日
15	2019 年 04 月 14 日	999 年 999 天 999 天	<1nom, 2max>	3104 年 4 月 8 日	3104 年 4 月 8 日
16	2550 年 12 月 30 日	0 年 0 月 0 天	<1max, 2min>	2550 年 12 月 30 日	2550 年 12 月 30 日
17	2550 年 12 月 30 日	0 年 0 月 1 天	<1max, 2min+>	2550 年 12 月 31 日	2550 年 12 月 31 日
18	2550 年 12 月 30 日	1 年 1 月 1 天	<1max, 2nom>	2552 年 1 月 31 日	2552 年 1 月 31 日
19	2550 年 12 月 30 日	999 年 999 天 998 天	<1max, 2max->	3635 年 12 月 23 日	3635 年 12 月 23 日
20	2550 年 12 月 30 日	999 年 999 天 999 天	<1max, 2max>	3635 年 12 月 24 日	3635 年 12 月 24 日
21	2550 年 12 月 31 日	0 年 0 月 0 天	<1max, 2min>	2550 年 12 月 31 日	2550 年 12 月 31 日

22	2550 年 12 月 31 日	0 年 0 月 1 天	<1max, 2min+>	2551 年 1 月 1 日	2551 年 1 月 1 日
23	2550 年 12 月 31 日	1 年 1 月 1 天	<1max, 2nom>	2552 年 2 月 1 日	2552 年 2 月 1 日
24	2550 年 12 月 31 日	999 年 999 天 998 天	<1max, 2max->	3635 年 12 月 24 日	3635 年 12 月 24 日
25	2550 年 12 月 31 日	999 年 999 天 999 天	<1max, 2max>	3635 年 12 月 25 日	3635 年 12 月 25 日

### 3. 减去天数

初始时间范围：[1601 年 1 月 1 日, 2550 年 12 月 31 日]

减去时间范围：[0 年 0 月 0 天, 999 年 999 月 999 天]

这里同样选用最坏情况测试。

编号	输入 1	输入 2	与边界值关系	期望输出	实际输出
1	1601 年 01 月 01 日	0 年 0 月 0 天	<1min, 2min>	1601 年 01 月 01 日	1601 年 01 月 01 日
2	1601 年 01 月 01 日	0 年 0 月 1 天	<1min, 2min+>	提示用户超过下限	超出日期
3	1601 年 01 月 01 日	1 年 1 月 1 天	<1min, 2nom>	提示用户超过下限	超出日期
4	1601 年 01 月 01 日	999 年 999 天 998 天	<1min, 2max->	提示用户超过下限	超出日期
5	1601 年 01 月 01 日	999 年 999 天 999 天	<1min, 2max>	提示用户超过下限	超出日期
6	1601 年 01 月 02 日	0 年 0 月 0 天	<1min+, 2min>	1601 年 01 月 02 日	1601 年 01 月 02 日
7	1601 年 01 月 02 日	0 年 0 月 1 天	<1min+, 2min+>	1601 年 01 月 01 日	1601 年 01 月 01 日
8	1601 年 01 月 02 日	1 年 1 月 1 天	<1min+, 2nom>	提示用户超过下限	超出日期
9	1601 年 01 月 02 日	999 年 999 天 998 天	<1min+, 2max->	提示用户超过下限	超出日期
10	1601 年 01 月 02 日	999 年 999 天 999 天	<1min+, 2max>	提示用户超过下限	超出日期
11	2019 年 04 月 14 日	0 年 0 月 0 天	<1nom, 2min>	2019 年 4 月 14 日	2019 年 4 月 14 日
12	2019 年 04 月 14 日	0 年 0 月 1 天	<1nom, 2min+>	2019 年 4 月 13 日	2019 年 4 月 13 日



13	2019 年 04 月 14 日	1 年 1 月 1 天	<1nom, 2nom>	2018 年 3 月 13 日	2018 年 3 月 13 日
14	2019 年 04 月 14 日	999 年 999 天 998 天	<1nom, 2max->	提示用户超过下限	超出日期
15	2019 年 04 月 14 日	999 年 999 天 999 天	<1nom, 2max>	提示用户超过下限	超出日期
16	2550 年 12 月 30 日	0 年 0 月 0 天	<1max, 2min>	2550 年 12 月 30 日	2550 年 12 月 30 日
17	2550 年 12 月 30 日	0 年 0 月 1 天	<1max, 2min+>	2550 年 12 月 29 日	2550 年 12 月 29 日
18	2550 年 12 月 30 日	1 年 1 月 1 天	<1max, 2nom>	2549 年 11 月 29 日	2549 年 11 月 29 日
19	2550 年 12 月 30 日	999 年 999 天 998 天	<1max, 2max->	提示用户超过下限	超出日期
20	2550 年 12 月 30 日	999 年 999 天 999 天	<1max, 2max>	提示用户超过下限	超出日期
21	2550 年 12 月 31 日	0 年 0 月 0 天	<1max, 2min>	2550 年 12 月 31 日	2550 年 12 月 31 日
22	2550 年 12 月 31 日	0 年 0 月 1 天	<1max, 2min+>	2550 年 12 月 30 日	2550 年 12 月 30 日
23	2550 年 12 月 31 日	1 年 1 月 1 天	<1max, 2nom>	2549 年 11 月 30 日	2549 年 11 月 30 日
24	2550 年 12 月 31 日	999 年 999 天 998 天	<1max, 2max->	提示用户超过下限	超出日期
25	2550 年 12 月 31 日	999 年 999 天 999 天	<1max, 2max>	提示用户超过下限	超出日期

### 3.1.4 小结

测试结果显示出了一个比较怪异的现象：计算机支持出现大于输入上界的结果，但是不支持出现小于输入下界的结果。也即：可以出现在 2550 年 12 月 31 日后的结果，但是本应出现先与 1601 年 1 月 1 日的结果时会显示超出日期。

但是考虑到需要查找 1601 年以前的日期的使用情况较少，这种情况是可以接受的。

总体来说，日期计算器通过了黑盒测试，没有出现不可忍受的错误。

## 3.2 程序员计算器

### 3.2.1 功能测试

#### 1. GUI 测试

伸缩窗口大小，界面根据窗口形状有两种布局，显示皆正常，且内部组件能够保持较为美观的比例。

#### 2. 功能

在程序员计算器中，用户可以

1. 选择计算宽度（QWORD、DWORD、WORD、BYTE）
2. 选择/切换输入进制（HEX，DEC，OCT，BIN）
3. 进行四则运算、位运算等

### 3.2.2 等价类测试

编号	状态	输入	等价类描述	预期输出	
1	HEX, DWORD	1+2	无进位十六进制加法	HEX 3 DEC 3 OCT 3 BIN 0011	HEX 3 DEC 3 OCT 3 BIN 0011
2	HEX, DWORD	1+F	带进位的十六进制加法	HEX 10 DEC 16 OCT 20 BIN 0001 0000	HEX 10 DEC 16 OCT 20 BIN 0001 0000
3	HEX, DWORD	FFFFFF FFF+2	上溢的十六进制加法	HEX 1 DEC 1 OCT 1 BIN 1	HEX 1 DEC 1 OCT 1 BIN 1
4	HEX, DWORD	0-1	下溢的十六进制减法	HEX FFFF FFFF DEC -1 OCT 37 777 777 777 BIN 1111 1111 1111 1111 1111 1111 1111	HEX FFFF FFFF DEC -1 OCT 37 777 777 777 BIN 1111 1111 1111 1111 1111 1111 1111

				1111	1111
5	HEX, DWORD	3*3	无进位的十六进制乘法	HEX 9 DEC 9 OCT 11 BIN 1001	HEX 9 DEC 9 OCT 11 BIN 1001
6	HEX, DWORD	F*F	进位的十六进制乘法	HEX E1 DEC 225 OCT 341 BIN 1110 0001	HEX E1 DEC 225 OCT 341 BIN 1110 0001
7	HEX, DWORD	FFFF* FFFF	上溢的十六进制乘法	HEX FFEF 0001 DEC -1, 114, 111 OCT 37 773 600 001 BIN 1111 1111 1110 1111 0000 0000 0000 0001	HEX FFEF 0001 DEC -1, 114, 111 OCT 37 773 600 001 BIN 1111 1111 1110 1111 0000 0000 0000 0001
8	HEX, DWORD	A/5	整除十六进制除法	HEX 2 DEC 2 OCT 2 BIN 0010	HEX 2 DEC 2 OCT 2 BIN 0010
9	HEX, DWORD	9/5	非整除十六进制除法	HEX 1 DEC 1 OCT 1 BIN 0001	HEX 1 DEC 1 OCT 1 BIN 0001
10	HEX, DWORD	1/0	十六进制除零	“除数不能为零”	“除数不能为零”
11	HEX, BYTE	80, 切换为 QWORD	可转换的字节长度切换	HEX FFFF FFFF FFFF FF80 DEC -128 OCT 1 777 777 777 777 777 777 600 BIN (14 组 1111) 1000 0000	HEX FFFF FFFF FFFF FF80 DEC -128 OCT 1 777 777 777 777 777 777 600 BIN (14 组 1111) 1000 0000
12	HEX, DWORD	8000 0000, 切换为 WORD	越界的字节长度切换	HEX 0 DEC 0 OCT 0 BIN 0	HEX 0 DEC 0 OCT 0 BIN 0

### 3.2.3 边界值测试

项	边界值	用例设计思路
DWORD 整数范围	8000 0000 (INT_MIN), 7FFF FFFF (INT_MAX), 0 (ZERO)	将四则运算的两项用边界值附近的值运算

编号	状态	输入	描述	预期结果	实际结果
1	DEC, DWORD	-2147483648 + 5	INT_MIN + CONST	HEX 8000 0005 DEC - 2, 147, 483, 643 OCT 20 000 000 005 BIN 1000 (6 个 0000) 0101	HEX 8000 0005 DEC - 2, 147, 483, 643 OCT 20 000 000 005 BIN 1000 (6 个 0000) 0101
2	DEC, DWORD	-2147483647 + 5	(>INT_MIN) + CONST	HEX 8000 0006 DEC - 2, 147, 483, 642 OCT 20 000 000 006 BIN 1000 (6 个 0000) 0110	HEX 8000 0006 DEC - 2, 147, 483, 642 OCT 20 000 000 006 BIN 1000 (6 个 0000) 0110
3	DEC, DWORD	2147483646 + 5	(<INT_MAX) + CONST	HEX 8000 0003 DEC - 2, 147, 483, 645 OCT 20 000 000 003 BIN 1000 (6 个 0000) 0011	HEX 8000 0003 DEC - 2, 147, 483, 645 OCT 20 000 000 003 BIN 1000 (6 个 0000) 0011
4	DEC, DWORD	2147483647 + 5	INT_MAX + CONST	HEX 8000 0004 DEC - 2, 147, 483, 644 OCT 20 000 000 004 BIN 1000 (6 个 0000) 0100	HEX 8000 0004 DEC - 2, 147, 483, 644 OCT 20 000 000 004 BIN 1000 (6 个 0000) 0100
5	DEC, DWORD	5 + (- 2147483648)	CONST + INT_MIN	HEX 8000 0005 DEC - 2, 147, 483, 643 OCT 20 000 000 005 BIN 1000 (6 个 0000) 0101	HEX 8000 0005 DEC - 2, 147, 483, 643 OCT 20 000 000 005 BIN 1000 (6 个 0000) 0101
6	DEC, DWORD	5 + (- 2147483647)	CONST + (>INT_MIN)	HEX 8000 0006 DEC - 2, 147, 483, 642 OCT 20 000 000 006 BIN 1000	HEX 8000 0006 DEC - 2, 147, 483, 642 OCT 20 000 000 006 BIN 1000

				(6 个 0000) 0110	(6 个 0000) 0110
7	DEC, DWORD	5 + 2147483646	CONST + (<INT_MAX)	HEX 8000 0003 DEC - 2, 147, 483, 645 OCT 20 000 000 003 BIN 1000 (6 个 0000) 0011	HEX 8000 0003 DEC - 2, 147, 483, 645 OCT 20 000 000 003 BIN 1000 (6 个 0000) 0011
8	DEC, DWORD	5 + 2147483647	CONST + INT_MAX	HEX 8000 0004 DEC - 2, 147, 483, 644 OCT 20 000 000 004 BIN 1000 (6 个 0000) 0100	HEX 8000 0004 DEC - 2, 147, 483, 644 OCT 20 000 000 004 BIN 1000 (6 个 0000) 0100
9	DEC, DWORD	5 + 5	CONST + CONST	HEX A DEC 10 OCT 12 BIN 1010	HEX A DEC 10 OCT 12 BIN 1010
10	DEC, DWORD	-2147483648 - 5	INT_MIN - CONST	HEX 7FFF FFFB DEC 2, 147, 483, 643 OCT 17 777 777 773 BIN 0111 (6 个 1111) 1011	HEX 7FFF FFFB DEC 2, 147, 483, 643 OCT 17 777 777 773 BIN 0111 (6 个 1111) 1011
11	DEC, DWORD	-2147483647 - 5	(>INT_MIN) - CONST	HEX 7FFF FFFC DEC 2, 147, 483, 644 OCT 17 777 777 774 BIN 0111 (6 个 1111) 1100	HEX 7FFF FFFC DEC 2, 147, 483, 644 OCT 17 777 777 774 BIN 0111 (6 个 1111) 1100
12	DEC, DWORD	2147483646 - 5	(<INT_MAX) - CONST	HEX 7FFF FFF9 DEC 2, 147, 483, 641 OCT 17 777 777 771 BIN 0111 (6 个 1111) 1001	HEX 7FFF FFF9 DEC 2, 147, 483, 641 OCT 17 777 777 771 BIN 0111 (6 个 1111) 1001
13	DEC, DWORD	2147483647 - 5	INT_MAX - CONST	HEX 7FFF FFFA DEC 2, 147, 483, 642 OCT 17 777 777	HEX 7FFF FFFA DEC 2, 147, 483, 642 OCT 17 777 777

				772 BIN 0111 (6 个 1111) 1010	772 BIN 0111 (6 个 1111) 1010
14	DEC, DWORD	5 - (- 2147483648)	CONST - INT_MIN	HEX 8000 0005 DEC - 2, 147, 483, 643 OCT 20 000 000 005 BIN 1000 (6 个 0000) 0101	HEX 8000 0005 DEC - 2, 147, 483, 643 OCT 20 000 000 005 BIN 1000 (6 个 0000) 0101
15	DEC, DWORD	5 - (- 2147483647)	CONST - (>INT_MIN)	HEX 8000 0004 DEC - 2, 147, 483, 644 OCT 20 000 000 004 BIN 1000 (6 个 0000) 0100	HEX 8000 0004 DEC - 2, 147, 483, 644 OCT 20 000 000 004 BIN 1000 (6 个 0000) 0100
16	DEC, DWORD	5 - 2147483646	CONST - (<INT_MAX)	HEX 8000 0007 DEC - 2, 147, 483, 641 OCT 20 000 000 007 BIN 1000 (6 个 0000) 0111	HEX 8000 0007 DEC - 2, 147, 483, 641 OCT 20 000 000 007 BIN 1000 (6 个 0000) 0111
17	DEC, DWORD	5 - 2147483647	CONST - INT_MAX	HEX 8000 0006 DEC - 2, 147, 483, 642 OCT 20 000 000 006 BIN 1000 (6 个 0000) 0110	HEX 8000 0006 DEC - 2, 147, 483, 642 OCT 20 000 000 006 BIN 1000 (6 个 0000) 0110
18	DEC, DWORD	5 - 5	CONST - CONST	HEX 0 DEC 0 OCT 0 BIN 0	HEX 0 DEC 0 OCT 0 BIN 0
19	DEC, DWORD	-2147483648 * 5	INT_MIN + CONST	HEX 8000 000 DEC - 2, 147, 483, 648 OCT 20 000 000 000 BIN 1000 (7 个 0000)	HEX 8000 000 DEC - 2, 147, 483, 648 OCT 20 000 000 000 BIN 1000 (7 个 0000)
20	DEC, DWORD	-2147483647 * 5	(>INT_MIN) * CONST	HEX 8000 0005 DEC - 2, 147, 483, 643 OCT 20 000 000 005 BIN 1000	HEX 8000 0005 DEC - 2, 147, 483, 643 OCT 20 000 000 005 BIN 1000

				(6 个 0000) 0101	(6 个 0000) 0101
21	DEC, DWORD	2147483646 * 5	(<INT_MAX) * CONST	HEX 7FFF FFF6 DEC 2, 147, 483, 638 OCT 17 777 777 766 BIN 0111 (6 个 1111) 0110	HEX 7FFF FFF6 DEC 2, 147, 483, 638 OCT 17 777 777 766 BIN 0111 (6 个 1111) 0110
22	DEC, DWORD	2147483647 * 5	INT_MAX * CONST	HEX 7FFF FFFB DEC 2, 147, 483, 643 OCT 17 777 777 773 BIN 0111 (6 个 1111) 1100	HEX 7FFF FFFB DEC 2, 147, 483, 643 OCT 17 777 777 773 BIN 0111 (6 个 1111) 1100
23	DEC, DWORD	5 * (- 2147483648)	CONST * INT_MIN	HEX 8000 000 DEC - 2, 147, 483, 648 OCT 20 000 000 000 BIN 1000 (7 个 0000)	HEX 8000 000 DEC - 2, 147, 483, 648 OCT 20 000 000 000 BIN 1000 (7 个 0000)
24	DEC, DWORD	5 * (- 2147483647)	CONST * (>INT_MIN)	HEX 8000 0005 DEC - 2, 147, 483, 643 OCT 20 000 000 005 BIN 1000 (6 个 0000) 0101	HEX 8000 0005 DEC - 2, 147, 483, 643 OCT 20 000 000 005 BIN 1000 (6 个 0000) 0101
25	DEC, DWORD	5 * 2147483646	CONST * (<INT_MAX)	HEX 7FFF FFF6 DEC 2, 147, 483, 638 OCT 17 777 777 766 BIN 0111 (6 个 1111) 0110	HEX 7FFF FFF6 DEC 2, 147, 483, 638 OCT 17 777 777 766 BIN 0111 (6 个 1111) 0110
26	DEC, DWORD	5 * 2147483647	CONST * INT_MAX	HEX 7FFF FFFB DEC 2, 147, 483, 643 OCT 17 777 777 773 BIN 0111 (6 个 1111) 1100	HEX 7FFF FFFB DEC 2, 147, 483, 643 OCT 17 777 777 773 BIN 0111 (6 个 1111) 1100

27	DEC, DWORD	5 * 5	CONST * CONST	HEX 19 DEC 25 OCT 31 BIN 0001 1001	HEX 19 DEC 25 OCT 31 BIN 0001 1001
28	DEC, DWORD	-2147483648 / 5	INT_MIN / CONST	HEX E666 6667 DEC - 429, 496, 729 OCT 34 631 463 147 BIN 1110 (6 个 0110) 0111	HEX E666 6667 DEC - 429, 496, 729 OCT 34 631 463 147 BIN 1110 (6 个 0110) 0111
29	DEC, DWORD	-2147483647 / 5	(>INT_MIN) / CONST	HEX E666 6667 DEC - 429, 496, 729 OCT 34 631 463 147 BIN 1110 (6 个 0110) 0111	HEX E666 6667 DEC - 429, 496, 729 OCT 34 631 463 147 BIN 1110 (6 个 0110) 0111
30	DEC, DWORD	2147483646 / 5	(<INT_MAX) / CONST	HEX 1999 999 DEC 429, 496, 729 OCT 3 146 314 631 0001 (6 个 1001) 1001	HEX 1999 999 DEC 429, 496, 729 OCT 3 146 314 631 0001 (6 个 1001) 1001
31	DEC, DWORD	2147483647 / 5	INT_MAX / CONST	HEX 1999 999 DEC 429, 496, 729 OCT 3 146 314 631 0001 (6 个 1001) 1001	HEX 1999 999 DEC 429, 496, 729 OCT 3 146 314 631 0001 (6 个 1001) 1001
32	DEC, DWORD	5 / (- 2147483648)	CONST / INT_MIN	HEX 0 DEC 0 OCT 0 BIN 0	HEX 0 DEC 0 OCT 0 BIN 0
33	DEC, DWORD	5 / (- 2147483647)	CONST / (>INT_MIN)	HEX 0 DEC 0 OCT 0 BIN 0	HEX 0 DEC 0 OCT 0 BIN 0
34	DEC, DWORD	5 / (-1)	CONST / (<ZERO)	HEX FFFF FFFB DEC -5 OCT 37 777 777 773 BIN (7 个 1111) 1011	HEX FFFF FFFB DEC -5 OCT 37 777 777 773 BIN (7 个 1111) 1011
35	DEC, DWORD	5 / 0	CONST / ZERO	“除数不能为 零”	“除数不能为 零”
36	DEC, DWORD	5 / 1	CONST / (>ZERO)	HEX 5 DEC 5 OCT 5 BIN 0101	HEX 5 DEC 5 OCT 5 BIN 0101
37	DEC, DWORD	5 / 2147483646	CONST / (<INT_MAX)	HEX 0 DEC 0 OCT 0 BIN 0	HEX 0 DEC 0 OCT 0 BIN 0



38	DEC, DWORD	5 / 2147483647	CONST / INT_MAX	HEX 0 DEC 0 OCT 0 BIN 0	HEX 0 DEC 0 OCT 0 BIN 0
39	DEC, DWORD	7 / 5	CONST / CONST	HEX 1 DEC 1 OCT 1 BIN 1	HEX 1 DEC 1 OCT 1 BIN 1

### 3.2.4 小结

测试中发现，对于 DEC 模式的输入，用户的输入范围为 INT\_MIN+1~INT\_MAX，因为用户不能直接输入负数，而必须通过(0-正数)的方式得到负数，因而不能直接在十进制下键入 INT\_MIN，设计不够人性化，有优化空间。

## 3.3 科学计算器

### 3.3.1 功能测试

#### 1. GUI 测试

伸缩窗口大小，界面根据窗口形状有两种布局，显示皆正常。

#### 2. 科学计算器功能

序号	功能
1	$\log(x)$
2	$\ln(x)$
3	$x \bmod y$
4	$\sqrt{x}$
5	$\sqrt[y]{x}$
6	$10^x$
7	$x^y$
8	$n!$
9	$\sin(x)$ [DEG, RAD]
10	$\sin^{-1}(x)$ [DEG, RAD]
11	$\cos(x)$ [DEG, RAD]
12	$\cos^{-1}(x)$ [DEG, RAD]
13	$\tan(x)$ [DEG, RAD]
14	$\tan^{-1}(x)$ [DEG, RAD]
15	复合运算符

### 3.3.2 等价类测试

$\log(x)$			
编号	输入	期望输出	等价类
1	$\log(2)$	0.69897000433601880478626110527551	正
2	$\log(0)$	Invalid input	负
3	$\log(-1)$	Invalid input	零

$\ln(x)$			
编号	输入	期望输出	等价类
1	$\ln(2)$	0.69314718055994530941723212145818	正
2	$\ln(-2)$	Invalid input	负
3	$\ln(0)$	Invalid input	零

$x \bmod y$			
编号	输入	期望输出	等价类
1	$7 \bmod 2$	0	x 正 y 正
2	$-3 \bmod 2$	-1	x 负 y 正
3	$5 \bmod -2$	1	x 正 y 负
4	$-5 \bmod -2$	-1	x 负 y 负
5	$0 \bmod 2$	0	x 零 y 正
6	$2 \bmod 0$	result is undefined	x 正 y 零
7	$0 \bmod 0$	result is undefined	x 零 y 零
8	$0 \bmod -2$	0	x 零 y 负
9	$-2 \bmod 0$	result is undefined	x 负 y 零

$\sqrt{x}$			
编号	输入	期望输出	等价类
1	$\sqrt{0}$	0	零
2	$\sqrt{2}$	0.69897000433601880478626110527551	正
3	$\sqrt{-2}$	Invalid input	负

$\sqrt[y]{x}$			
编号	输入	期望输出	等价类
1	3 yroot 1	3	x 正 y 正
2	1 yroot 0	Cannot divide by zero	x 正 y 零
3	3 yroot -2	0.57735026918962576450914878050196	x 正 y 负
4	0 yroot 3	0	x 零 y 正
5	0 yroot 0	Cannot divide by zero	x 零 y 零
6	0 yroot -2	Invalid input	x 零 y 负
7	-2 yroot 2	Invalid input	x 负 y 正
8	-2 yroot 0	Cannot divide by zero	x 负 y 零
9	-3 yroot -2	Invalid input	x 负 y 负

$10^x$			
编号	输入	期望输出	等价类
1	10 ^ 0	1	零
2	10 ^ 9999	1.e+9999	正
3	10 ^ -9999	1.e-9999	负

$x^y$			
编号	输入	期望输出	等价类
1	2^4	16	x 正 y 正
2	2^0	1	x 正 y 零
3	2^-4	0.0625	x 正 y 负
4	0^4	0	x 零 y 正
5	0^0	1	x 零 y 零
6	0^-4	Invalid input	x 零 y 负
7	-2^4	16	x 负 y 正
8	-2^0	1	x 负 y 零
9	-2^-4	0.0625	x 负 y 负

$n!$			
编号	输入	期望输出	等价类
1	10 ^ 0	1	零
2	10 ^ 9999	1.e+9999	正
3	10 ^ -9999	1.e-9999	负

$\sin(x)$ [DEG, RAD]			
编号	输入	期望输出	等价类
1	sin(0) DEG	0	零
2	sin(90) DEG	1	正
3	sin(-90) DEG	-1	负

$\sin^{-1}(x)$ [DEG, RAD]			
编号	输入	期望输出	等价类
1	sin-1(1)	90	正
2	sin-1(-1)	-90	负
3	sin-1(0)	0	零
4	sin-1(2)	Invalid input	正无效
5	sin-1(-2)	Invalid input	负无效

$\cos(x)$ [DEG, RAD]			
编号	输入	期望输出	等价类
1	cos(0) DEG	1	零
2	cos(90) DEG	0	正
3	cos(-90) DEG	0	负

$\cos^{-1}(x)$ [DEG, RAD]			
编号	输入	期望输出	等价类
1	cos-1(1) DEG	0	正
2	cos-1(-1) DEG	180	负
3	cos-1(0) DEG	90	零
4	cos-1(2) DEG	Invalid input	正无效
5	cos-1(-2) DEG	Invalid input	负无效

$\tan x$ [DEG, RAD]			
编号	输入	期望输出	等价类
1	tan(0) DEG	0	零
2	tan(90) DEG	invalid input	正无效
3	tan(-90) DEG	invalid input	负无效
4	tan(2147483647) DEG	-1.3270448216204100371594725740869	正

5	tan(- 2147483647) DEG	1.3270448216204100371594725740869	负
---	-----------------------------	-----------------------------------	---

$\tan^{-1}(x)$ [DEG, RAD]			
编号	输入	期望输出	等价类
1	tan-1(1)	45	正
2	tan-1(-1)	-45	负
3	tan-1(0)	0	零

### 3.3.3 边界值测试

$\log(x)$		
编号	输入	期望输出
1	log(2)	0.69897000433601880478626110527551
2	log(2147483647)	9.3319298653811829240651035172848
3	log(2147483648)	9.3319298655834170516259057364593
4	log(1)	0
5	log(0)	Invalid input
6	log(-1)	Invalid input

$\ln(x)$		
编号	输入	期望输出
1	ln(2)	0.69314718055994530941723212145818
2	ln(0)	Invalid input
3	ln(-2)	Invalid input

$x \bmod y$		
编号	输入	期望输出
1	7 mod 2	0
2	-3 mod 2	-1
3	5 mod -2	1
4	-5 mod -2	-1
5	0 mod 2	0
6	2 mod 0	result is undefined

7	0 mod 0	result is undefined
---	---------	---------------------

$\sqrt{x}$		
编号	输入	期望输出
1	sqrt(0)	0
2	sqrt(2)	0.69897000433601880478626110527551

$\sqrt[y]{x}$		
编号	输入	期望输出
1	$\sqrt[2]{3}$	1.7320508075688772935274463415059
2	$\sqrt[1]{3}$	3
3	$\sqrt[3]{0}$	0
4	$\sqrt[0]{1}$	Cannot divide by zero
5	$\sqrt[-2]{3}$	0.57735026918962576450914878050196
6	$\sqrt[-1]{3}$	0.33333333333333333333333333333333

$10^x$		
编号	输入	期望输出
1	$10^0$	1
2	$10^{9999}$	1.e+9999
3	$10^{-9999}$	1.e-9999
4	$10^{10000}$	Overflow
5	$10^{(-10000)}$	Overflow
6	$10^{2147483647}$	Invalid input

$x^y$		
编号	输入	期望输出
1	$3^7$	2187
2	$2147483647^{2187}$	Overflow
3	$2147483647^{2147483647}$	Invalid input
4	$1^{2147483647}$	1
5	$0^0$	1
6	$0^5$	0

$n!$		
编号	输入	期望输出
1	0!	1
2	1!	1
3	3!	6
4	(-2)!	Invalid input

$\sin(x)$ [DEG, RAD]		
编号	输入	期望输出
1	sin(0) DEG	0
2	sin(90) DEG	1
3	sin(-90) DEG	-1
4	sin(2147483647) DEG	0.79863551004729284628400080406894
5	sin(-2147483647) DEG	-0.79863551004729284628400080406894
6	sin(pai) RAD	0
7	sin(-pai) RAD	0
8	sin(2147483647) RAD	-0.7249165551445563905482932963398
9	sin(-2147483647) RAD	0.7249165551445563905482932963398

$\sin^{-1}(x)$ [DEG, RAD]		
编号	输入	期望输出
1	sin-1(1)	90
2	sin-1(-1)	-90
3	sin-1(0)	0
4	sin-1(2)	Invalid input
5	sin-1(-2)	Invalid input

$\cos(x)$ [DEG, RAD]		
编号	输入	期望输出
1	cos(0) DEG	1
2	cos(90) DEG	0
3	cos(-90) DEG	0
4	cos(2147483647) DEG	-0.60181502315204827991797700044149
5	cos(-2147483647) DEG	-0.60181502315204827991797700044149
6	cos(pai) RAD	-1

7	cos(-pai) RAD	-1
8	cos(2147483647) RAD	-0.68883669187794383467975822303793
9	cos(-2147483647) RAD	-0.68883669187794383467975822303793

$\cos^{-1}(x)$ [DEG, RAD]		
编号	输入	输出
1	cos-1(1) DEG	0
2	cos-1(-1) DEG	180
3	cos-1(0) DEG	90
4	cos-1(2) DEG	Invalid input
5	cos-1(-2) DEG	Invalid input
6	cos-1(1) RAD	0
7	cos-1(-1) RAD	3.1415926535897932384626433832795
8	cos-1(0) RAD	1.5707963267948966192313216916398

$\tan(x)$ [DEG, RAD]		
编号	输入	输出
1	tan(0) DEG	0
2	tan(90) DEG	invalid input
3	tan(-90) DEG	invalid input
4	tan(2147483647) DEG	-1.3270448216204100371594725740869
5	tan(-2147483647) DEG	1.3270448216204100371594725740869
6	tan(pai) RAD	0
7	tan(-pai) RAD	0
8	tan(2147483647) RAD	1.0523779637351339136698139284747
9	tan(-2147483647) RAD	-1.0523779637351339136698139284747

$\tan^{-1}(x)$ [DEG, RAD]		
编号	输入	输出
1	tan-1(1)	45
2	tan-1(-1)	-45
3	tan-1(0)	0



4	$\tan^{-1}(2)$	63.434948822922010648427806279547
5	$\tan^{-1}(-2)$	-63.434948822922010648427806279547
6	$\tan^{-1}(2147483647)$	89.999999973319573542213650732095
7	$\tan^{-1}(-2147483647)$	-89.999999973319573542213650732095
8	$\tan^{-1}(2147483647^8)$	90
9	$\tan^{-1}(-2147483647^8)$	-90

复合运算符		
编号	输入	输出
1	$\sin^{-1}(\sin(180))$	0
2	$\sin^{-1}(\sin(-90))$	-90
3	$\cos^{-1}(\cos(-180))$	180
4	$\cos^{-1}(\cos(-90))$	90
5	$\tan^{-1}(\tan(90))$	Invalid input
6	$\tan^{-1}(\tan(180))$	0
7	$\log(3^5)/\log(3)$	5
8	$\log(3^{-5})/\log(3)$	-5
9	$\log(10^{1000})$	1000
10	$\log(10^{10000})$	Overflow
11	$(10 \text{ yroot } 1000) ^ 1000$	10
12	$(10 \text{ yroot } 10000) ^ 10000$	10
13	$(10 \text{ yroot } 10000)^{(10000^2)}$	Overflow
14	$5 \bmod 0 \bmod 2$	Result is undefined
15	$5 \bmod 2 \bmod 0$	Result is undefined
16	$0 \bmod 5 \bmod 2$	0
17	$10!/(2!*8!)$	45
18	$10!/2!$	1814400

### 3.3.4 小结

所有测试用例皆符合期望输出。

测试时发现，退回键只能退回已输入的数字，不能退回运算符号，设计不够人性化。

该科学计算器缺少组合数、排列数的计算功能，只能通过复合  $n!$  计算，不够便捷；缺少计算  $\log_y x$  的功能，只能通过复合  $\log x$  计算，不够便捷。

## 3.4 标准计算器

### 3.4.1 功能测试

#### 1.GUI 测试

伸缩窗口大小，界面根据窗口形状有两种布局，显示皆正常。

#### 2. 标准计算器功能

序号	功能
1	+
2	-
3	*
4	/
5	%
6	$\sqrt{\quad}$
7	$x^2$
8	1/x
9	MS
11	$\pm$
12	M +
13	M -

### 3.4.2 等价类测试

+			
编号	输入	期望输出	等价类
1	1+1	2	正+正
2	1+-2	-1	正+负
3	-1+-1	-2	负+负
4	1+0	1	正+零
5	-1+0	-1	负+零
6	0+0	0	零+零

-			
编号	输入	期望输出	等价类
1	2-1	1	正-正
2	-1-2	-3	负-负
3	2-0	2	正-零
4	-1-0	-1	负-零
5	1-0	1	正-零
6	-1- -2	1	负-负
7	0-1	-1	零-正
8	0—1	1	零-负
9	0-0	0	零-零

*			
编号	输入	期望输出	等价类
1	7 * 2	14	正*正
2	7*-2	-14	正*负
3	-7*-2	14	负*负
4	0*0	0	零*零
5	-2*0	0	负*零
6	2*0	0	正*零

/			
编号	输入	期望输出	等价类
1	3/2	1.5	正/正
2	4/-2	-2	正/负
3	3/0	Invalid input	正/零
4	-4/2	-2	负/正
5	-4/-2	2	负/负
6	-2/0	Invalid input	负/零
7	0/2	0	零/正
8	0/-2	0	零/负
9	0/0	Invalid input	零/零

%			
编号	输入	期望输出	等价类
1	3	0.03	正
2	-3	-0.03	负
3	0	0	零

$\sqrt{\quad}$			
编号	输入	期望输出	等价类
1	0	0	零
2	4	2	正
3	-4	Invalid input	负

$x^2$			
编号	输入	期望输出	等价类
1	0	0	零
2	2	4	正
3	-2	4	负

$1/x$			
编号	输入	期望输出	等价类
1	0	Cannot divided by zeor	零
2	2	0.5	正
3	-2	-0.5	负

$\pm$			
编号	输入	期望输出	等价类
1	1	-1	正
2	-1	1	负

<i>MS</i>			
编号	输入	期望输出	等价类
1	0 MS	M=0	零
2	3 MS	M=3	正
3	-3 MS	M=-3	负

<i>M +</i>			
编号	输入	期望输出	等价类
1	M=0, 3M+	M=3	M=0,正 M+
2	M=0,-3M+	M=-3	M=0,负 M+
3	M=0,0M+	M=0	M=0,零 M+
4	M=1, 3M+	M=4	M 正,正 M+
5	M=1,-3M+	M=-2	M 正,负 M+
6	M=1,0M+	M=1	M 正,零 M+
7	M=-1, 3M+	M=2	M 负,正 M+
8	M=-1,-3M+	M=-4	M 负,负 M+
9	M=-1,0M+	M=-1	M 负,零 M+

<i>M -</i>			
编号	输入	期望输出	等价类
1	M=0, 3M-	M=-3	M=0,正 M+
2	M=0,-3M-	M=3	M=0,负 M+
3	M=0,0M-	M=0	M=0,零 M+
4	M=1, 3M-	M=-2	M 正,正 M+
5	M=1,-3M-	M=4	M 正,负 M+
6	M=1,0M-	M=1	M 正,零 M+
7	M=-1, 3M-	M=-4	M 负,正 M+
8	M=-1,-3M-	M=2	M 负,负 M+
9	M=-1,0M-	M=-1	M 负,零 M+

### 3.4.3 边界值测试

+		
编号	输入	期望输出
1	1+1	2
2	1+-1	0
3	1+0	1
4	-1+-1	-2
5	0+-1	-1
6	0.1+0.1	0.2

—		
编号	输入	期望输出
1	1-1	0
2	2-1	1
3	1-2	-1
4	1—1	2
5	1-0	1
6	-1-0	-1
7	-1—1	0
8	-1-1	-2
9	-1--2	1

*		
编号	输入	期望输出
1	7 * 2	14
2	7 * 0	0
3	-7*-2	14
4	7*-2	-14
5	-2*0	0
6	0*0	0

/		
编号	输入	期望输出
1	3/2	1.5
2	2/3	0.66666666666667
3	0/3	0
4	3/0	Can not divide by zer
5	-3/2	-1.5
6	-3/-2	1.5
7	4/2	2

%		
编号	输入	期望输出
1	0	0
2	8	0.08
3	-8	-0.08
4	100	1
5	-100	-1
6	101	1.01

$\sqrt{\quad}$		
编号	输入	期望输出
1	$8\sqrt{\quad}$	2.828427124746
2	$4\sqrt{\quad}$	2
3	$0\sqrt{\quad}$	0
4	$5.5\sqrt{\quad}$	2.345207879911
5	$-5\sqrt{\quad}$	Invalid input

$x^2$		
编号	输入	期望输出
1	$3 x^2$	9
2	$0 x^2$	0
3	$0.1 x^2$	0.01
4	$-5 x^2$	25

1/x		
编号	输入	期望输出
1	0 1/x	Cannot divided by zero
2	1 1/x	1
3	8 1/x	0.125
4	-1 1/x	-1
5	-8 1/x	-0.125

±		
编号	输入	期望输出
1	0	0
2	-1	1
3	1	-1

MS		
编号	输入	期望输出
1	1 MS	M = 1
2	0 MS	M = 0
3	-1 MS	M = -1

M +		
编号	输入	期望输出
1	M = 1, 5M+	M=6
2	M = 1, 0M+	M=1
3	M=0, -1M+	M= -1

M -		
编号	输入	期望输出
1	M = 1, 5M-	M=-4
2	M = 1, 0M-	M=1
3	M = -1, 5M-	M=-6
4	M = -1, -1M-	M=0



### 3.3.4 小结

所有测试用例皆符合期望输出。

测试时发现，退回键只能退回已输入的数字，不能退回运算符号，设计不够人性化。标准计算器没有括号，不能连续运算，只能先把上一步算好才能做下一步。

## 4. 总结

通过对 Windows10 内嵌计算器的黑盒测试，我们认为程序能够基本满足用户的需求，基本功能覆盖全面且没有错误，尽管有一些部分交互设计不够合理，但是该计算器仍然是一个出色的软件。

## 5. 参考

1. Microsoft Calculator : <https://github.com/Microsoft/calculator>
2. 软件测试黑盒报告模板：  
<https://wenku.baidu.com/view/5887260ce45c3b3566ec8b2a.html>