

华东师范大学数据科学与工程学院

2023 秋-计算机系统-LAB1

实验报告



学生姓名：郭力维

学号：10225501439

目录

1. 解题思路
 - a. bitXor
 - b. tmin
 - c. isTmax
 - d. allOddBits
 - e. negate
 - f. isAsciiDigit
 - g. conditional
 - h. isLessOrEqual
 - i. logicalNeg
 - j. howManyBits
 - k. floatScale2
 - l. floatFloat2Int
2. 运行结果
3. 解题流程 (git log -all)

解题思路

bitXor:

使用 $a = x \& \sim y$ 以及 $b = \sim x \& y$ 分别确定“ $x = 1, y = 0$ ”的位和“ $x = 0, y = 1$ ”的位；再通过 $\sim a \& \sim b$ 确定了“ $x = 0, y = 0$ ”的位；最后使用一次 \sim 确定了“ $x \neq 0, y \neq 0$ ”的位，即实现了 \wedge 运算。因此使用 $\text{return } \sim(\sim a \& \sim b)$ 可以实现 \wedge 运算。

tmin:

直接使用 $1 \ll 31$ 找到补码所表示的最小整数。

isTmax:

构建 $a = \sim x$ 以及 $b = x + 1$ 两个变量并使用 $!(a \wedge b)$ 来判别输入的 x 是否符合最大二进制补码的性质。另外还使用了 $!!b$ 来判别 x 是否为 -1 。

allOddBits:

构建了掩码 $\text{oddBitMask} = 0xAAAAAAAA$ ，先使用 $x \& \text{oddBitMask}$ 运算获得 x 所有奇数位上的分量情况，再使用 $!((x \& \text{oddBitMask}) \wedge \text{oddBitMask})$ 对 x 上所有奇数位的分量情况进行一次性判别。

negate:

直接运用补码性质返回 $\sim x + 1$ 。

isAsciiDigit:

先构造上界和下界 $lowerBound = 0x30, upperBound = 0x39$ 。在对输入值 x 是否处于两个界之内分别进行一次判别 $!((x + \sim lowerBound + 1) \gg 31), !((upperBound + \sim x + 1) \gg 31)$ 。最后使用 $\&$ 连接两个判别值并返回。

conditional:

首先构建掩码 $mask = ((!!x) \ll 31) \gg 31$ ，之后使用一个返回值 $return (y \& mask) | (z \& \sim mask)$ 在判别的同时保留输出值。

isLessOrEqual:

首先构建变量 $sign = y + \sim x + 1$ 用于储存 $y - x$ 的值，再使用 $!(sign \gg 31)$ 根据该值正负直接返回大小比较的判别值。

更新：后续发现在处理较大正整数与较小负整数时会因溢出出现错误，于是增添了符号判别以根据符号直接进行比较。

logicalNeg:

构建掩码 $0x00000000$ ，之后直接使用 $!(x | a)$ 返回 x 相较于 0 值的判别值。

更正：使用 0 及其补码的性质，构造 $a = \sim((x \gg 31) \wedge x)$ ，再进行一次输入值是否为-1 的判别后，根据符号位返回判别值。

howManyBits:

先对输入的数进行符号判定，若为负数则取补码（除了-1）。之后再使用 $bit16, bit8, bit4, bit2, bit1$ 变量结合移位运算进行二分探查。最后加上符号位，返回正确值。

floatScale2:

分别获取其符号位，尾数，阶码之后，根据三者的值进行判别，对应不同情况进行相应的操作，得到正确函数。

floatFloat2Int:

先分别获取其符号位，位数，阶码，根据三者情况判断是否返回溢出值。对阶码对应范围进行简单的限定，并对应阶码做移位运算，以返回对应的整型值。

运行结果

```
jovyan@jupyter-10225501439:~/DaSE.CSAPP.Autumn/lab1/hwdata$ ./driver.pl
1. Running './dlc -z' to identify coding rules violations.

2. Compiling and running './btest -g' to determine correctness score.
gcc -O -Wall -m32 -lm -o btest bits.c btest.c decl.c tests.c
btest.c: In function 'test_function':
btest.c:332:23: warning: 'arg_test_range[1]' may be used uninitialized in this function [-Wmaybe-uninitialized]
    if (arg_test_range[1] < 1)
                        ^~~~~~

3. Running './dlc -Z' to identify operator count violations.

4. Compiling and running './btest -g -r 2' to determine performance score.
gcc -O -Wall -m32 -lm -o btest bits.c btest.c decl.c tests.c
btest.c: In function 'test_function':
btest.c:332:23: warning: 'arg_test_range[1]' may be used uninitialized in this function [-Wmaybe-uninitialized]
    if (arg_test_range[1] < 1)
                        ^~~~~~

5. Running './dlc -e' to get operator count of each function.
```

Correctness Results			Perf Results		
Points	Rating	Errors	Points	Ops	Puzzle
1	1	0	2	8	bitXor
1	1	0	2	1	tmin
1	1	0	2	7	isTmax
2	2	0	2	9	allOddBits
2	2	0	2	2	negate
3	3	0	2	11	isAsciiDigit
3	3	0	2	8	conditional
3	3	0	2	14	isLessOrEqual
4	4	0	2	12	logicalNeg
4	4	0	2	36	howManyBits
4	4	0	2	14	floatScale2
4	4	0	2	25	floatFloat2Int

Score = 56/56 [32/32 Corr + 24/24 Perf] (147 total operators)

以上是运行评分脚本的结果。

解题流程

```
commit f0a298d625f6028a7f4c7b537bb7fe2d6893ff00 (HEAD -> homework01)
Author: GAMA-Lab-10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Sun Oct 8 15:25:25 2023 +0800

    Run the driver.pl on shuishan.net and complete the report

commit 28b2867426c626f660c10079d69e44069c841c7f
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Sun Oct 8 17:18:12 2023 +0800

    Change the driver.pl to make sure it can run correctly

commit f875b7031b0e8abachebf8413f810effbbeb84bc
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Sun Oct 8 17:15:53 2023 +0800

    Complete all function, and update the function 'logicalNeg' accroding to the results from dlc

commit 5282f0b7fbcl9ad6eae8d5fc42df63010e4352a
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Fri Oct 6 20:17:38 2023 +0800

    Update the function 'isLessOrEqual'

commit 58852f64d998d50f22549db13f64a19d63272fb3
Author: GAMA-Lab-10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Wed Oct 4 19:30:42 2023 +0800

    Complete the code of function 'howManyBits' and commit it

commit e0d8aa598e1fd2830cc254aff3cd5c0b7770029
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Mon Oct 2 22:04:14 2023 +0800

    Update the code of function 'isLessOrEqual'

commit 5b51e8b844f9b3c27e0e5ff2840c1d6835d3439b
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Mon Oct 2 19:56:49 2023 +0800

    Complete the function 'logicalNeg' and commit its code

commit 701cd4969e63a7421dae4971bf2fe9f4d7309378
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Mon Oct 2 17:34:44 2023 +0800

    Complete the codes of 'conditional' and 'isLessOrEqual' and commit them. So I finished all codes without partition 4

commit 1d9976351738d87197217f0c28ec3199ea8356be
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Mon Oct 2 16:44:30 2023 +0800

    Complete function 'isAsciiDigit' and commit its code

commit 65fcd06403b5d957db8ff4d7d3a0bb0e2465780f6
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Mon Oct 2 16:18:13 2023 +0800

    Complete function 'allOddBits' and 'negate' and commit their codes

commit 41c5205f963a0345eb48b5400c2f27db4cecf70
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Mon Oct 2 15:03:24 2023 +0800

    Light the code of function 'isTmax' and commit it

commit 380d83a579b59fb3d0e4519f72eb14d97d5a426
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Mon Oct 2 14:35:35 2023 +0800

    Update function 'isTmax' and commit codes

commit 8ec3b8689fc307b3207cc3332941854bb98d56c6
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Sun Oct 1 18:27:59 2023 +0800

    Update function 'isTmax' and its code

commit cd50403bd6257c4d60cad9216d98636706a3c9b4
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Sun Oct 1 18:26:25 2023 +0800

    Complete function 'isTmax' and commit the code

commit ec65fbbdfc1cca711e31bbca3208e4cb7d870cb4
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Sun Oct 1 16:28:50 2023 +0800

    Complete function 'tmin' and commit the code

commit 693c958a4a5fd9e09f29a045a2e837ca33f33e6
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Sun Oct 1 15:44:09 2023 +0800

    Complete function 'bitXor' and commit it

commit cf5b53d2e01673a50cdd0f8e14831ef8d20fab
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Sun Oct 1 14:46:11 2023 +0800

    Read all instructions

commit 90c150231cbbb717c8e96d3bc3dfc2237bda08e (origin/master, origin/HEAD, master, dev)
Author: 10225501439 <10225501439@stu.ecnu.edu.cn>
Date: Wed Sep 27 11:39:59 2023 +0800

    clone the homework files and init the rep of lab1
:
```