- 姓名：傅锦龙
- 班级：1621301
- 学号：162130117
- 报告阶段：lab1
- 完成日期：2023.4.17
- 本次实验，我完成了所有内容。

目录

1. Nuaa_question1

- 思路

~(x & y)实现的是只要不同时为1则为1，只要除掉同时为0取1的情况便实现了异或，而~x & ~y实现的是只有同时为0则为1，则~(~x & ~y)实现的是同时为0则为0，这样就排除掉了同时为0取1的情况。

- 代码

```
int bitXor(int x, int y)
{
    return ~(~x & ~y) & ~(x & y);
}
```

- 测试截图（dlc btest）

```
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc -e bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command
dlc:bits.c:170:bitXor: 7 operators
dlc:bits.c:181:negate: 2 operators
dlc:bits.c:193:divpwr2: 7 operators
dlc:bits.c:206:rotateRight: 9 operators
dlc:bits.c:218:addOK: 7 operators
dlc:bits.c:244:float_f2i: 10 operators

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./btest
Score   Rating  Errors  Function
 1      1       0       bitXor
 2      2       0       negate
 2      2       0       divpwr2
 3      3       0       rotateRight
 3      3       0       addOK
 4      4       0       float_f2i
Total points: 15/15
```

2. Nuaa_question2

- 思路

按位反再加一即可。

- 代码

```
int negate(int x)
{
    return ~x + 1;
}
```

- 测试截图（dlc btest）

```
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc -e bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command
dlc:bits.c:170:bitXor: 7 operators
dlc:bits.c:181:negate: 2 operators
dlc:bits.c:193:divpwr2: 7 operators
dlc:bits.c:206:rotateRight: 9 operators
dlc:bits.c:218:addOK: 7 operators
dlc:bits.c:244:float_f2i: 10 operators

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./btest
Score   Rating  Errors  Function
 1      1       0       bitXor
 2      2       0       negate
 2      2       0       divpwr2
 3      3       0       rotateRight
 3      3       0       addOK
 4      4       0       float_f2i
Total points: 15/15
```

3. Nuaa_question3

- 思路

先将x加上一个偏移量，然后再右移n位。偏移量的计算方式是，如果x是负数，则加上2^n-1，否则加上0，通过x符号位扩展到32位与上2^n-1的结果来实现。

- 代码

```
int divpwr2(int x, int n)
{
    return (x + ((x >> 31) & ((1 << n) + ~0))) >> n;
}
```

- 测试截图（dlc btest）

```
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc -e bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command
dlc:bits.c:170:bitXor: 7 operators
dlc:bits.c:181:negate: 2 operators
dlc:bits.c:193:divpwr2: 7 operators
dlc:bits.c:206:rotateRight: 9 operators
dlc:bits.c:218:addOK: 7 operators
dlc:bits.c:244:float_f2i: 10 operators

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./btest
Score   Rating  Errors  Function
 1      1       0       bitXor
 2      2       0       negate
 2      2       0       divpwr2
 3      3       0       rotateRight
 3      3       0       addOK
 4      4       0       float_f2i
Total points: 15/15
```

4. Nuaa_question4

- 思路

将x向右移位n位并将高n位清0，再将x向左移位32-n位，最后将这两个结果进行或运算得到结果。

- 代码

```
int rotateRight(int x, int n)
{
    int t = ~n + 33;
    return ((x >> n) & ((1 << t) + ~0)) | (x << t);
}
```

- 测试截图（dlc btest）

```
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc -e bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command
dlc:bits.c:170:bitXor: 7 operators
dlc:bits.c:181:negate: 2 operators
dlc:bits.c:193:divpwr2: 7 operators
dlc:bits.c:206:rotateRight: 9 operators
dlc:bits.c:218:addOK: 7 operators
dlc:bits.c:244:float_f2i: 10 operators

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./btest
Score   Rating  Errors  Function
 1      1       0       bitXor
 2      2       0       negate
 2      2       0       divpwr2
 3      3       0       rotateRight
 3      3       0       addOK
 4      4       0       float_f2i
Total points: 15/15
```

5. Nuaa_question5

- 思路

如果x和y的符号位相同，但是x+y的符号位与x的符号位不同，那么就会溢出。先不取符号位，先运算，最后再移位得到符号位的情况。先通过异或运算判断x和y的符号位以及x+y与x的符号位是否相同，通过取反和与运算确定当溢出时运算结果符号位才为1，通过位运算得到符号位扩展，非一下得到最终结果。

- 代码

```c
int addOK(int x, int y)
{
    return !((~(x ^ y) & (x ^ (x + y))) >> 31);
}
```

- 测试截图（dlc btest）

```
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc -e bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command
dlc:bits.c:170:bitXor: 7 operators
dlc:bits.c:181:negate: 2 operators
dlc:bits.c:193:divpwr2: 7 operators
dlc:bits.c:206:rotateRight: 9 operators
dlc:bits.c:218:addOK: 7 operators
dlc:bits.c:244:float_f2i: 10 operators

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./btest
Score   Rating  Errors  Function
 1      1       0       bitXor
 2      2       0       negate
 2      2       0       divpwr2
 3      3       0       rotateRight
 3      3       0       addOK
 4      4       0       float_f2i
Total points: 15/15
```

6. Nuaa_question6

- 思路

先得到阶码exp和符号位sign，小数部分f通过uf向左移8位，并在最高位赋值为1，先作为正数看待，为了避免之后判断阶码来进行左移或右移，直接都进行右移，且为了避免算数右移，将结果赋值给无符号数。因为Nan和inf都返回0x80000000,因此0x80000000可包含在判断exp>157的情况内，而剩余有效的数的绝对值范围都在2^31内且对称，所以可行。为避免移位过多，须先判断exp是否小于127，最后f右移相应位数，根据sign判断是否需要取负数，结果为f。

- 代码

```c
int float_f2i(unsigned uf)
{
    int exp = (uf >> 23) & 0xff;
    int sign = uf & 0x80000000;
    unsigned f = (uf << 8) | 0x80000000;
    if (exp > 157)
        return 0x80000000;
    if (exp < 127)
        return 0;
    f >>= (158 - exp);
    if (sign)
        return -f;
    return f;
}
```

- 测试截图（dlc btest）

```
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./dlc -e bits.c
/usr/include/stdc-predef.h:1: Warning: Non-includable file <command
dlc:bits.c:170:bitXor: 7 operators
dlc:bits.c:181:negate: 2 operators
dlc:bits.c:193:divpwr2: 7 operators
dlc:bits.c:206:rotateRight: 9 operators
dlc:bits.c:218:addOK: 7 operators
dlc:bits.c:244:float_f2i: 10 operators

Compilation Successful (1 warning)
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./btest
Score   Rating  Errors  Function
 1      1       0       bitXor
 2      2       0       negate
 2      2       0       divpwr2
 3      3       0       rotateRight
 3      3       0       addOK
 4      4       0       float_f2i
Total points: 15/15
```

7. 最终结果

- ./driver.pl 截图

```
fujinlong@ubuntu:/mnt/hgfs/course/lab1-handout$ ./driver.pl
1. Running './dlc -z' to identify coding rules violations.
/usr/include/stdc-predef.h:1: Warning: Non-includable file <comm

Compilation Successful (1 warning)

2. Compiling and running './btest -g' to determine correctness s
gcc -O -Wall -m32 -lm -o btest bits.c btest.c decl.c tests.c

3. Running './dlc -Z' to identify operator count violations.
/usr/include/stdc-predef.h:1: Warning: Non-includable file <comm

Compilation Successful (1 warning)

4. Compiling and running './btest -g -r 2' to determine performa
gcc -O -Wall -m32 -lm -o btest bits.c btest.c decl.c tests.c


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       7       bitXor
2       2       0       2       2       negate
2       2       0       2       7       divpwr2
3       3       0       2       9       rotateRight
3       3       0       2       7       addOK
4       4       0       2       10      float_f2i

Score = 27/27 [15/15 Corr + 12/12 Perf] (42 total operators)
```

- 挑战教授截图（必须是你的学号）

1	2	3	4	5	6	Winner?	Score	Nickname
7	2	7	9	7	10	Winner!	12	162130117

8. 备注

助教真帅