AB IdeaLab, Competitive Programming Team, Fall 18–Spring 19 January 18, 2019

Lecture 2: Bit-String Flicking

American Computer Science League, January Contest

Lecturer: Alexander Sun *Editor: Sanjit Bhat*

1 Introduction/Lecture

Bit String Flicking is a general term to denote operations that can be done to bit strings/ binary strings. They can involve multiple strings or just a single string. There are 3 general types of operations, your basic truth operators, shifts, and circulates.

1.1 Truth Operations

Bit String flicking only involves 4 main operations(they should seem familiar), AND, OR, NOT, EXOR, and all the inverses of them. In contest AND $\rightarrow$ &, OR $\rightarrow$ |, NOT $\rightarrow \sim$, EXOR $\rightarrow \oplus$.

E.X. 101 | 010 = 111

1.2 Shift

Shifting is very straight forward involving 2 operations. You can shift right or left. A shift, literally shifts the digits the specified amount in the specified direction, while maintaining the same amount of digits, filling in 0's fordigits shifted out of the number.

E.X. LSHIFT-3 100010 = 010000, RSHIFT-3 100010 = 000100

The trick to solving these problems, is to cover up the amount of digits specified on the right or left side, then just fill in 0's for the rest of the digits.

1.3 Circulate

Circulating is extremely similar to shifting, except no digits will be deleted by getting shifted out of the number. Instead they are added back (or circulated) on to the other end of the number, either in the right or left direction, a specified amount.

E.X. RCIRC-3 10111 = 11110 LCIRC-3 10111 = 11101

The method to solving these problems, are to take the number of digits on the specified side and shifting them directly behind the remaining digits on the other side.

1.4 Miscellaneous/General Tips

Order of Operations: NOT, SHIFT / CIRC, AND, XOR, OR
Solving problems with variables often have more then one solution. Some questions will ask you to list out all values, and others will ask for the amount of possible values. Generally just work from the outside in, and if necessary assign values abced. for each digit the expression is equivalent to, and use that to help you.

2 Exercises

Some borrowed from here.

1. 10010 OR 11101
2. 10001 AND 11011 OR 11001
3. 11100 OR 10101 AND 10111
4. NOT 101 AND 100
5. 101011 XOR 110101
6. 101011 OR 100111 XOR 100010
7. NOT (11101 XNOR 10100)
8. 1011 OR 1101 XOR 1000 AND 1010 OR 1110 AND NOT 1000
9. Find all possible values of x: (RCIRC-2(LSHIFT-1 (NOT X)))=00101
10. List all possible unique values of x that solve the following equation:
    (LSHIFT-1 (10110 XOR (RCIRC-3 x) AND 11011)) = 01100
11. (RSHIFT-1 (LCIRC-4 (RCIRC-2 01101)))
12. ((RCIRC-14 (LCIRC-23 01101)) | (LSHIFT-1 10011) & (RSHIFT-2 10111))
    Hint: Order of operations, generally parenthesis dictate order