

2020 시스템 프로그래밍

- Lab 01. DataLab -

제출일자	2020.9.27
분 반	01
이 름	박재우
학 번	201902694

실습 1

1	Int bitAnd(int x, int y)
---	--------------------------

드모르간의 법칙을 이용해 x and y 구현

2	Int bitNor(int x, int y)
---	--------------------------

드모르간의 법칙을 이용해

$$X \mid y = \sim x \& \sim y$$

3	Int getByte(int x, int n)
---	---------------------------

N 에 3 만큼 쉬프트 (n *8)

X 를 쉬프트한 n 만큼 왼쪽 쉬프트(n*8bit 만큼 쉬프트 된다)

원하는 비트만 출력을 위해 255(0xff)를 and 시켜준다.

4	Int byteSwap(int x, int n, int m)
---	-----------------------------------

N 과 m 을 3 씩 오른쪽 쉬프트(getByte)와 이유는 같다

255 를 각각 n,m 쉬프트 된 결과의 값만큼 왼쪽으로 쉬프트해준다

이때 이 값이 0 이면 byte 이동이 없으므로 x 를 리턴한다.

이후 n,m 쉬프트 값만큼 각각 x 를 오른쪽 쉬프트 해준다

각각 $x \gg (n \text{ 혹은 } m < 3) \& 255$ 를 해주면 n,m 위치의 byte 가 가장오른쪽에

있고 또한 나머지는 전부 0 이 되어있다.

이를 다시 오른쪽으로 n,m < 3 만큼 이동을 시켜주면 원래 바이트 위치에 있다.

이럴경우 각각 입력받은 n,m 위치의 바이트를 제외하고 다 0 이므로

Or 을 시켜주면 바이트위치가 바뀐채 출력된다.

5	Int addOK(int x, int y)
---	-------------------------

X 와 y 부호를 통해 overflow 확인
X+y 의 부호를 처리하기위해 >>31(최상위비트만 남긴다)
같은 이유로
 $x >> 31, y >> 31$
x,y,의 부호가 다르면 overflow
같다면 x+y 와 x,y,와의 부호가 다른지 검사
같으면 overflow

6	Int isGreater(int x, int y)
---	-----------------------------

x-y 계산후 부호 처리(최상위 비트만 남긴다)
그 후 x,y 도역시 최상위 비트만 남긴다.
X 의 부호가 0 yrk 1 이면 $x < y$
아닐경우
Cal 이 1(x-y 가 양수)이면 $x > y$ 이다

7	Int satMul3(int x)
---	--------------------

X 의 부호, x+x 의 부호, x+x+x 의 부호를 각각 처리
X 와 x+x 의 부호를 비교하여 overflow 가 났는지 확인
X 와 x+x+x 의 부호를 비교하여 overflow 가 났는지 확인
 $Of = (x \wedge (x+x)) \mid (x \wedge (x+x+x));$
Of == 1 이면 부호에 따라 최대 최소 리턴
Of == 0 이면 x+x+x 를 리턴

8	Int absVal(int x)
---	-------------------

X 의 부호처리 cal >> 31(1111...)

양수이면 $x + 0 \ll 0$ 이므로 그냥 x 가 리턴

음수이면

$\text{Cal} + x \rightarrow -x$ 의 1의 보수

위의 1의 보수를 구하면

X 가 나온다

大