Author: Keith Stellyes

A few proofs regarding bitwise operations.

Proof `k << 1 = k * 2` where `<<` is the bitshift left operation and k is an integer represented as a series of bits.

The bit string ordered set B is equal to the integer $I(B)$ where:

$$I(B) = \sum_{k=0}^{|B|-1} 2^k \times B_k$$

For convenience, we define a function, $bi(b, k)$ where $b = 0 \oplus b = 1$ where:

$$bi(b, k) = 2^k \times b$$

Which lets us re-express $I(B)$ as:

$$I(B) = \sum_{k=0}^{|B|-1} bi(B_k, k)$$

The bit string representing the left bitshift is defined by the function $lshift(B)$ where:

$$lshift(B)_k = \begin{cases} 0 & \text{if } k \leq 0 \\ B_{k-1} & \text{if } k > 0 \end{cases}$$

We want to prove the following:

$$I(lshift(B)) = 2 \times I(B) =$$

$$\sum_{k=0}^{|B|-1} bi(B_k, k) = 2 \times \sum_{k=0}^{|B|-1} bi(lshift(B)_k, k)$$

$$bi(m, n+1) = 2 \times bi(m, n)$$

$$2 \times 2^n \times m = 2^{n+1} \times m$$

For $k > 0$

$k + 1$ to maintain the magnitude...:

$$bi(lshift(B)_k, k) = bi(B_{k-1}, k+1) = 2 \times bi(B_{k-1}, k)$$

$$\sum_{k=0}^{|B|-1} bi(B_k, k) = 2 \times \sum_{k=0}^{|B|-1} bi(B_{k-1}, k) =$$

$$2 \times \sum_{k=0}^{|B|-1} bi(leftshift(B_k), k)$$

Proof `1 << k =` 2^k

Case I. $k = 0$

$$n << 0 = n \text{ where } n \in \mathbb{Z}$$

$$1\texttt{<<}0 = 1$$

Case II. $k > 0$

$$\text{Where } n > 0 \text{ and } n \in \mathbb{Z}$$

$$m\texttt{<<}n = m \times 2_1 \times 2_2 \ldots 2_{n-1}$$

$$2\texttt{<<}n = 2 \times 2_1 \times 2_2 \times 2_2 \ldots 2_{n-1}$$

$$2\texttt{<<}k = 2^k$$

Proof `k & 1 = isOdd(k)`

A `&` B is the bitwise AND which returns a set defined as:

$$(A \texttt{ \& } B)_k = A_k \times B_k$$

$$(A \texttt{ \& } B)_k = \begin{cases} 1 & \text{if } A_k = B_k = 1 \\ 0 & \text{else} \end{cases}$$

Prove:
$I(A \texttt{ \& } M) = 1 \rightarrow I(A \texttt{ \& } B) = 2k + 1$
$I(A \texttt{ \& } M) = 0 \rightarrow I(A \texttt{ \& } B) = 2k$

Where M is a set where $k = 0 \rightarrow M_k = 1$ and $k > 0 \rightarrow M_k = 0$
1. The sum of two even numbers is also even.

$$\text{Where } i, j, k \in \mathbb{Z} : 2i + 2j = 2k$$

2. The sum of an even number and an odd number is odd.

$$\text{Where } i, j, k \in \mathbb{Z} : 2i + 1 + 2j = (2i + 2j) + 1 = 2k + 1$$

3. A power of 2 is even if that power is an integer greater than 0.

$$2^n \rightarrow 2^n = 2k \text{ where } n > 0$$

Case I. I(A) = 2k + 1

$$2^0 = 2k + 1$$
$$A_0 = 1 \rightarrow I(B) = 2k + 1$$
$$(A \texttt{ \& } M)_0 = 1 \rightarrow A_0 = 1 \rightarrow I(A) = 2k + 1$$

Case II. I(A) = 2k

$$A_0 = 0 \rightarrow I(A) \rightarrow 2k$$
$$(A \texttt{ \& } M)_0 = 0 \rightarrow A_0 = 0$$
$$I(A) = 2k$$

Proof `n >> 1 = n div 2`

$$rshift(B)_k = \begin{cases} 0 & \text{if } k = |B| - 1 \\ B_{k+1} & \text{if } k < |B| - 1 \end{cases}$$

Where $k \in \mathbb{Z}$:
Even number $= 2k$
Odd number $= 2k + 1$
We define the a `div` 2 operator as follows:

$$a \ \texttt{div} \ 2 = \begin{cases} a \div 2 & \text{if } a = 2k \\ (a-1) \div 2 & \text{if } a = 2k+1 \end{cases}$$

If $B_0 = 0$:

$$B = rshift(lshift(B)) = lshift(rshift(B))$$

We must prove the following:

$$I(rshift(B)) = I(B) \ \texttt{div} \ 2$$

To do so, we will prove by cases:

Case I. $I(B) = 2k$

$$n{>>}1 = n \text{ div } 2$$
$$n{>>}1 = \frac{n}{2}$$
$$({<<})n{>>}1 = 2 \times \frac{n}{2}$$
$$n = n$$

Case II. $I(B) = 2k + 1$

$$n{>>}1 = n \text{ div } 2$$
$$n{>>}1 = \frac{n-1}{2}$$
$$n{>>}1 = (n-1){>>}1$$
$$(n-1){>>}1 = \frac{n-1}{2}$$
$$n - 1 = n - 1$$
$$n = n$$