CS2020A Discrete Mathematics

Tutorial 04 | 25/Aug/2025

Prove the following or give a counterexample.

Theorem 1. For every natural number n,

$$\sum_{i=0}^{n} f_i = f_{n+2} - 1,$$

where $f_0, f_1, \ldots$ is the Fibonacci Sequence defined as $f_0 = 0, f_1 = 1$ and for every $k \geq 2$, $f_k = f_{k-1} + f_{k-2}$.

Theorem 2. Every natural number can be expressed as the sum of a *unique* set of powers of two.

Theorem 3. The number of subsets of $\{1, \ldots, n\}$ which do not contain any pair of consecutive numbers is f_{n+2}. (where f_n is a defined in the first task)

Theorem 4. For any two natural numbers a and b,

$$\gcd(a, b) = \gcd(b, a \;\%\; b),$$

where $a \;\%\; b$ is the remainder obtained when dividing a by b.

Prove the correctness of the following algorithms using the principle of induction. Also argue why the algorithms will terminate

Algorithm 1.

```python
def gcd(a, b):
    # Input:    Two natural numbers a and b
    # Output:   The greatest common divisor of a and b
    if b == 0:
        return a
    else:
        return gcd(b, a % b)
```

Algorithm 2.

```python
def gcd_ext(a, b):
    # Input:    Two natural numbers a and b
    # Output:   d, x, y, where d = gcd(a,b) and d = ax + by
    if b == 0:
        return a, 1, 0
    else:
        d, x, y = gcd_ext(b, a % b)
        return d, y, x - (a//b)*y
```