Peergrade assignment 4

Author: Kristoffer Højelse

May 28, 2020

Exercise 1

Base case n = 7

$3^7 < 7! \iff 2187 < 5040$

Base case holds.

Assume $P(k)$ to prove $P(k+1)$

I.H.: $P(k) := 3^k < k!$

$P(k+1) := 3^{k+1} < (k+1)!$

$3^{k+1} = 3^k \cdot 3$ by factorization

by I.H.

$3^{k+1} < k! \cdot 3$

$3^{k+1} < k! \cdot (k+1)$, since $k > 6$

$3^{k+1} < (k+1)!$

$\square$

Exercise 2

${f_0}^2 + {f_1}^2 + {f_2}^2 + \cdots + {f_n}^2 = f_n \cdot f_{n+1}$

$f(n) = \{0, 1, 1, 2, 3, 5, 8, ...\}$

${f_0}^2 = f_0 \cdot f_{n+1} \iff 0 = 0 \cdot 1$

Base case holds.

Definition of Fibonacci sequence $f_n = f_{n-1} + f_{n-2}$

Assume $P(k)$ to prove $P(k+1)$

I.H.: $P(k) := \sum_{i=0}^{k} {f_i}^2 = f_k \cdot f_{k+1}$

$P(k+1) := \sum_{i=0}^{k+1} {f_i}^2 = f_{k+1} \cdot f_{k+2}$

$\iff \sum_{i=0}^{k} {f_i}^2 + {f_{k+1}}^2 = f_{k+1} \cdot f_{k+2}$

by I.H.

$\iff f_k \cdot f_{k+1} + {f_{k+1}}^2 = f_{k+1} \cdot f_{k+2}$

$\iff f_k \cdot f_{k+1} + {f_{k+1}}^2 = f_{k+1} \cdot (f_k + f_{k+1})$ by definition of the Fibonacci sequence.
$\iff f_k \cdot f_{k+1} + {f_{k+1}}^2 = f_k \cdot f_{k+1} + {f_{k+1}}^2$

$\square$

Exercise 3

The recursive algorithm is probably more efficient for n = 3 (maybe also for 4 and 5), and the integrative algorithm is more efficient for larger numbers.

The execution time of the iterative algorithm will grow about linearly as n gets bigger.

The execution time of the recursive algorithm probably grows with some exponential function.

Recursive algorithm

$a : n \to a(n)$
$a(0) := 1$
$a(1) := 3$
$a(2) := 5$
$a(n) = a(n-1) \cdot (a(n-2))^2 \cdot (a(n-3))^3$

Iterative algorithm

```
int a(int n){
    int tmp0 = 1;
    int tmp1 = 3;
    int tmp2 = 5;

    if(i < 3){
        return 2*n+1;
    }

    if(i >= 3){
        int result;
        for(int i = 0; i < n+1; i++){
            result = tmp2 * tmp1*tmp1 * tmp0*tmp0*tmp0;
            tmp0 = tmp1;
            tmp1 = tmp2;
            tmp2 = result;
        }
        return result;
    }

}
```
