FPP Term Project Report

Yale-NUS College AY 17/18

Jeremy Yew
A0156262H

6 December 2017

1.Table of Contents

2. 2 x 2 matrices

- **Property 10 (Associativity of multiplication):**
  - Expansion and then arithmetic manipulation. The lemma shuffle_helper stated precisely the shuffle pattern I required. Even though I still needed to give it arguments each time, it is still shorter than using the less precise shuffle lemmas provided by Coq.
- **Property 12 (I is neutral on the left and right of multiplication):**
  - The proof steps are almost identical in both except that we rewrite multiplications of 1 and 0 on the right instead of the left in **I_neutral_mul22_r**. This simply reflects the fact that the 0's in I are on the right in the multiplication.
- **Proposition 29 (M x M^n is commutative):**
  - This is why the proof about the exponentiation of M 1 1 0 1 works with both definitions of exponentiation.
  - **Exercise 31** also works because both definitions of exponentiation are equivalent.
- **Proposition 38 (transposition and exponentiation of M is commutative):**
  - This is true because transposition is distributive over multiplication; on the RHS, transposing an exponentiation involves transposing both M and M^n, which flips the order of the arguments to be multiplied. After which Proposition 29 helps us switch it back to match LHS.
- **Exercise 40 (Proof of Proposition 33 using Proposition 14):**
  - We may use Proposition 14 by doubly-transposing the matrices in LHS and RHS (the equation is the same since transposition is involutive). The singly-transposed arguments then reflect the statement of Proposition 14. After which Proposition 38 helps switch the order of operations for RHS to match LHS.
- **Exercise 25 (Powers of F):**
  - Since F^n effectively returns three consecutive Fibonnaci numbers, we might use F^n to implement another function fib_v4 which which calculates Fibonacci numbers iteratively. It is similar to fib_v3 (from week 4), in that F^n only makes n recursive calls (linear time), as opposed to much more with fib.
  - We may also implement fib_v5 which is also iterative, but makes 2 less calls than fib_v4, capitalizing on the fact that F^n stores both the n^{th} and $n+1^{th}$ Fibonacci numbers, which can be added to compute the $n+2^{th}$ Fibonacci number.
- **Subsidiary Question (fibonacci_addition_of_powers_of_F):**
  - The proposition stated is simply due to the fact that the elements of the matrix are successive fibonacci numbers – the same recurrence relation is thus reflected in the powers of F.
- **In general:**
  - Unfold lemmas are only for recursive functions.

- With arithmetic expansion in the context of this exercise, intuitively it seems to me the most efficient to rewrite in the following order:
  - distributivity and associativity of multiplication,
  - associativity of addition,
  - multiplications of 0 on the right or left,
  - multiplications of 1 on the right or left,
  - additions of 0 on the right or left and/or all additions of 1 on the right or left.

3. A commutative diagram

4. Reflections

5. Acknowledgements

Thank you Prof Danvy for your kind patience and generosity. It has been a fruitful experience.