School of Computer and Cyber Sciences / Department of Computer Science
Augusta University
Instructor: Dr. Vale, D.

Exercises

Exercise 1 (Propositional Logic). Prove these tautologies of propositional logic, using only the tactics `apply`, `assumption`, `constructor`, `destruct`, `intro`, `intros`, `left`, `right`, `split`, and `unfold`.

(a) $(True \lor False) \land (False \lor True)$

(b) $P \to \neg\neg P$

(c) $P \land (Q \lor R) \to (P \land Q) \lor (P \land R)$

Exercise 2 (First-Order Logic). Let some variable T of type Set be the set of individuals. Assume x is a constant, p is a unary predicate, q is a binary predicate, and f is a unary function. Prove the following tautology of first-order logic, using only the tactics `apply`, `assert`, `assumption`, `destruct`, `eapply`, `eassumption`, and `exists`.

$$p\,x \to (\forall x.\, p\,x \to \exists y.\, qxy) \to (\forall xy.\, q\,x\,y \to q\,y\,(f\,y)) \to \exists z.\, q\,z(f\,z)$$

Hint. You will probably need to use the `assert` tactic useful for stating and proving an intermediate lemma, enabling a kind of *forward reasoning*, in contrast to the backward reasoning that is the default for Rocq's tactics.

Exercise 3 (Let's prove Leftpad). The function `leftpad` takes a character, a length, and a string, and *pads* the string to that length. It pads the input string by adding the character to the left. Hence its name; padleft. For instance, `padleft("!", 5, "foo")` should return the string `"!!foo"`; and `leftpad("!", 0, "foo")` should return `"foo"`.

(a) Implement this function in Rocq.

(b) Design the specification for correctness of this function, that is, what are the precise statements that express that this function is correct?

(c) State and prove its correctness w.r.t your specification.

Hint.

- Rocq's standard library is missing some useful functions that you might need to implement.
- Notice that the length of the output string is $\max(n, \mathrm{len}(str))$.