

## HW #1 – Logical Foundations in Coq

Published: 30.3.23

Due: 22.4.23 by midnight.

The assignment consists of two parts. **Read the submission instructions carefully.**

1. Complete the following exercises in the files Basics.v, Induction.v, Lists.v, Poly.v **attached to this assignments.**

- from `Basics.v`:
  - andb\_true\_elim2
  - identity\_fn\_applied\_twice
  - andb\_eq\_orb
  - [binary](#)
- from `Induction.v`:
  - basic\_induction
  - evenb\_S
  - mult\_comm
  - eqb\_refl
  - plus\_swap'
- from `Lists.v`:
  - list\_funs
  - alternate
  - bag\_functions
  - bag\_more\_functions
  - list\_exercises
  - remove\_does\_not\_increase\_count
  - baz\_num\_elts
- from `Poly.v`:
  - mumble\_grumble
  - more\_poly\_exercises
  - split
  - partition
  - map\_rev
  - currying

2. Create a new file titled “drop” where you answer the following questions. Make sure to test your definitions.

- A. Write a function **drop n m lst** that meets the following (partial) specification: **drop n m lst** returns the **n** elements of **lst**, starting after the **m**th element. For example, `drop 3 2 [7,8,20,2,11,15,6]` returns `[20,2,11]`.
- B. Formulate two meaningful statements about this function and prove them correct. (Do not pick trivial statements)

### Submission instructions:

To solve the assignment, upload the files Basics.v, Induction.v ,Lists.v and Poly.v **attached to this assignments** to jsCoq using the scratchpad. (Instructions on how to use jsCoq can be found here:

<https://github.com/jscoq/jscoq/blob/v8.14/README.md>.)

Complete the exercises mentioned above in each of the files. Then, save each file into a .v file. The submission to Moodle should include the four completed .v files as well as a drop.v file for the second part of the assignment.

In order for the auto grader to work correctly on your Coq files (and give you full credit for your work!), please be careful to follow these rules:

- **Do not Zip the Coq files.**
- Use **only** the files attached to this assignment. Do not use any other version of these files. The other exercises in these files (not listed above) are marked optional in the exercise files
- Do not alter the "markup" that delimits exercises: the Exercise header, the name of the exercise, the "empty square bracket" marker at the end, etc. Please leave this markup exactly as you find it.
- Do not delete exercises. If you skip an exercise (e.g., because it is marked "optional," or because you can't solve it), it is OK to leave a partial proof in your [.v] file; in this case, please make sure it ends with [Admitted] (not, for example [Abort]).
- It is fine to use additional definitions (of helper functions, useful lemmas, etc.) in your solutions. You need to put these between the exercise header and the theorem you are asked to prove. The previous item also applies such lemmas that you end up being unable to prove, hence end it with [Admitted], then make sure to also end the main theorem in which you use it with [Admitted], not [Qed]. This will help you get partial credit, in case you use that main theorem to solve a later exercise.
- **A tip: make sure your file can run till the end. If you have any problematic lemmas - end them with [Admitted].**

The completed exercise (the Coq files and the written text) is to be submitted via Moodle.

**Good Luck!**