

# Functional and Logic Programming

**Web-page:** <https://web.unbc.ca/~casper/Semesters/2023-05F/370.php>

**Accommodations:** If there is any student in this course who, because of a disability, may have a need for special academic accommodations, please come and discuss this with me, or contact the Access Resource Centre located in Teaching & Learning 10-1048.

**Prerequisites:** A grade of C<sup>-</sup> or better in 141, and 281; or permission of instructor.

**Moodle-page:** <https://moodle.unbc.ca/>

**Instructor:** David Casperson

**Office:** T&L 10-2050

**Phone:** 960-6672

**e-mail:** [David.Casperson@unbc.ca](mailto:David.Casperson@unbc.ca)

## Grading Scheme and Dates:

Homework:	25%	
First class:		Wed, Sep 04
Midterm 1:	20%	2024-10-11
Midterm 2:	20%	2024-11-15
Last class:		Tue, Dec 03
Final Exam:	35%	

**Lecture times:** MWF 08:30–09:20. **Room:** 10-4072. There are *no* labs or tutorials.

**Office Hours:** Scheduled for:

M W 11:00-12:00

Please feel to interrupt me whenever you can find me, or schedule alternate times.

Times may be changed after consulting at the beginning of the semester.

**Use of Generative AI:** Use of Generative AI must be explicitly acknowledged where used, and can only be used with permission. Please avoid templates and/or Gen AI when sending me e-mail.

**Programming Assignments:** There will be many small programming assignments during the semester. Programming languages include HASKELL and PROLOG.

**Text Books and References:** There is no required textbook, but [2] is a great introduction to HASKELL freely available online. I assume that you have access to it.

Book [5] provides another good beginner reference, and also has a free web version.

[1, 4] are PROLOG books in the UNBC library.

## References

- [1] W. F. Clocksin and C. S. Mellish, *Programming in prolog*, Springer Verlag, 1981.
- [2] Miran Lipovaca, *Learn you a haskell for great good!: A beginner's guide*, 1st ed., No Starch Press, San Francisco, CA, USA, 2011.
- [3] Conor McBride and Ross Paterson, *Applicative programming with effects*, *Journal of Functional Programming* **18** (2008), no. 1, 1–13,, Available at <http://www.soi.city.ac.uk/~ross/papers/Applicative.html>.
- [4] Richard A. O'Keefe, *The craft of prolog*, Logic Programming, MIT Press, 1990.
- [5] Bryan O'Sullivan, John Goerzen, and Don Stewart, *Real world haskell*, 1st ed., O'Reilly Media, Inc., 2008.
- [6] P. L. Wadler, *Comprehending monads*, Proceedings of the 1990 ACM Conference on LISP and Functional Programming, Nice (New York, NY), ACM, 1990, pp. 61–78.

Students who successfully complete CPSC 370 will:

- appreciate the functional and logic programming paradigms;
- be cognizant of the associated theory and language implementation techniques (for instance, the  $\lambda$ -calculus, closures, unification, thunks, and so on),

and consequently will be stronger programmatic problem solvers in whatever programming environment they work.

### Approximate Course Content:

#### GENERAL

- Characteristics of functional programming languages,
- Strict and lazy evaluation
- Pure and impure functions
- static versus lexical scoping
- Basic syntax.
- Mathematical concepts: Relations, Functions, partial functions, Cartesian products, Disjoint unions, "Currying", categories, monads, combinators
- Programming: recursion, tail recursion, tail recursion strategies, pure data structures.

#### HASKELL:

- Types, polymorphic types, type classes
- data structures.
- immutability and its consequences
- Laziness: its implementation consequences
- Semigroups and Monoids; Foldables and Traversable.
- Functors, Applicatives, and Monoids.
- Monads, do-notation, monadic programming.

#### PROLOG

- What logic programming is.
- Facts. Rules. Goals. Variables. Conjunctions. Horn Clauses.
- The Unification algorithm.
- Programming strategies: Accumulator arguments, difference lists.
- Cuts. Negation.
- Arithmetic.