

Schedule

A week-by-week breakdown of the material.

Overview

- Introduction to Haskell and Fuctional Programming (1.1-1.14)
- Compiler and interpreter (2.1-2.7)
- Basic types (3.1-3.7)
- Designing and writing programs (4.1-4.8)
- Tuples and lists (5.1-5.7)
- More programming with lists (6.1-6.8)
- Defining functions over lists (7.1-7.6)
- Input and output in Haskell (8.1-8.6)
- Patterns of computation (10.1-10.5)
- Higher-order functions (11.1-11.5)
- Developing higher-order programs (12.1-12.7)
- Overloading and type-classes (13.1-13.8)
- Algebraic types (14.1-14.6)
- Case study: Huffman codes (15.1-15.7)
- Abstract data types (16.1-16.9)
- Lazy evaluation (17.1-17.8)
- I/O programming and Monads (18.1-18.6)

Week 1

- Mon**
- Reading: 1.1-1.11. Optional: 1.12-1.14
 - Setting up¹
 - Introduction to Haskell and Fuctional Programming²

- Wed**
- Reading: 2.1-2.7

¹[notes/setup.html](https://www.haskell.org/notes/setup.html)

²[notes/intro.html](https://www.haskell.org/notes/intro.html)

- Commands for GHCi interactive mode³
- Practice with GHCi⁴

- Fri**
- Reading: 3.1-3.7
 - Standard Haskell values and types.⁵
 - Conditionals. Guarded Expressions.⁶
 - Assignment 0. Due Wed 01/15⁷

Week 2

- Mon**
- Reading: 4.1-4.3, 4.8, 5.1-5.3
 - Compound Types⁸
 - Type Aliases and Custom Types.⁹

- Wed**
- Reading: 5.4-5.7
 - Working with the GHC compiler and interpreter. Lists.¹⁰
 - List Comprehensions.¹¹
 - Assignment 1. Due Mon 01/20¹²

- Fri**
- Catchup/Practice

Week 3

- Mon**
- List comprehension practice: Book Library¹³

- Wed**
- Reading: 6.1-6.3, 6.7
 - Parametric polymorphism¹⁴
 - The supermarket billing example¹⁵

Fri TBD

³[notes/ghci_commands.html](#)

⁴[notes/ghci_practice.html](#)

⁵[notes/standard.html](#)

⁶[notes/functions_conditionals.html](#)

⁷[assignments/assignment0.html](#)

⁸[notes/compoundTypes.html](#)

⁹[notes/types_custom.html](#)

¹⁰[notes/lists.html](#)

¹¹[notes/list_comprehensions.html](#)

¹²[assignments/assignment1.html](#)

¹³[notes/list_comp_practice.html](#)

¹⁴[notes/parametric_polymorphism.html](#)

¹⁵[notes/supermarket_billing.html](#)

Week 4

Mon TBD

Wed TBD

Fri TBD

Week 5

Mon TBD

Wed TBD

Fri TBD

Week 6

Mon TBD

Wed TBD

Fri TBD

Week 7

Mon TBD

Wed TBD

Fri TBD

Week 8

Mon TBD

Wed TBD

Fri TBD

Week 9

Mon TBD

Wed TBD

Fri TBD

Week 10

Mon TBD

Wed TBD

Fri TBD

Week 11

Mon TBD

Wed TBD

Fri TBD

Week 12

Mon TBD

Wed TBD

Fri TBD

Week 13

Mon TBD

Wed TBD

Fri TBD

Old links

- Currying¹⁶
- More advanced typing: Curried Functions. Polymorphism, Type classes.¹⁷ (3.6-3.9)
- More advanced typing: Curried Functions. Polymorphism, Type classes. (cont)¹⁸ (3.6-3.9)
- Pattern Matching.¹⁹ (4.4)

¹⁶[notes/currying.html](#)

¹⁷[notes/types_advanced.html](#)

¹⁸[notes/types_advanced.html](#)

¹⁹[notes/pattern_matching.html](#)

- More practice with Pattern Matching.²⁰
- Version Control²¹
- Assignment 2. Due 09/29²²
- Recursion²³ (6.1-6.6)
- Anonymous Functions. Sections.²⁴ (4.5-4.6)
- Assignment 3. Due 10/13²⁵
- The Maybe (Option) Type.²⁶
- Functions as Values: Difference Lists, Composition²⁷ (7.5)
- Functions as Values: Difference Lists, Composition (cont)²⁸ (7.5)
- MIDTERM (study guide²⁹)
- Interactive Programming³⁰ (10.1-10.5)
- Practice with Interactive Programming³¹ (10.6)
- BREAK
- Recursive Types³² (8.4)
- Assignment 4. Due 11/03³³
- Folding³⁴ (7.3-7.4)
- Overview of Software Development Practices³⁵
- Information hiding and abstraction with modules³⁶
- Testing³⁷

²⁰[notes/more_pattern_matching.html](#)

²¹[notes/version_control.html](#)

²²[assignments/assignment2.html](#)

²³[notes/recursion.html](#)

²⁴[notes/anonymous_functions.html](#)

²⁵[assignments/assignment3.html](#)

²⁶[notes/maybe.html](#)

²⁷[notes/difference_lists.html](#)

²⁸[notes/difference_lists.html](#)

²⁹[notes/midterm_study_guide.html](#)

³⁰[notes/interactive.html](#)

³¹[notes/interactive_hangman.html](#)

³²[notes/recursive_types.html](#)

³³[assignments/assignment4.html](#)

³⁴[notes/folding.html](#)

³⁵[notes/dev_overview.html](#)

³⁶[notes/modules.html](#)

³⁷[notes/testing.html](#)

- The State Monad³⁸
- Functors, Applicatives, Monads³⁹
- Specification Testing with HSpec⁴⁰
- Final Study Guide⁴¹

³⁸[notes/functors_monads.html](#)

³⁹[notes/functors_monads.html](#)

⁴⁰[notes/testing_hspec.html](#)

⁴¹[notes/final_study_guide.html](#)