

CS 254: Introduction to Functional Programming • Sample Midterm

Fritz Ruehr • Willamette University

Name: _____

Instructions

This is a sample exam which shows what typical questions look like. A real exam would have about 100 points; for example, perhaps 15 T/F questions = 15 pts; 10 short answer = 50 points; 3 longer answer = 35 points. An hour and a half exam might run 6-7 pages total, but note that a lot of the space is for answers.

A. Basic knowledge—True or False (2 points each)

1. In Haskell, strings of text are presented as lists of individual characters. _____
2. In Haskell, numbers of type `Int` are presented as lists of digit characters. _____
- ...
15. Variables defined by *local declarations* (using **where**) are available throughout the whole module. _____

Typically 10-15 questions worth 1-2 points each in the “True or False” section.

B. Multiple choice and short answer questions (5 points each)

1. One danger of *recursion* (i.e., when a variable or function is used in its own definition) is that it might cause evaluation to go on forever. How does Haskell handle this possibility?
 - (a) Haskell uses its type system to reject such definitions as “nonsense”
 - (b) definitions that may evaluate forever are not allowed: they give an error when you load the file
 - (c) Haskell will keep trying to evaluate as long as the evaluation goes
 - (d) when WinHugs sees that evaluation is taking forever, it stops and asks if you want to continue_____

[choose the single best answer]

2. Say that `f` is a function which takes a list of characters and an (arbitrarily large) integer, as two separate arguments, and returns a list of Booleans; how would you write this fact (that `f` has this type) in Haskell?

Typically 8-12 questions worth 4-6 points each in the “Multiple choice/short answer” section.

