Midterm review

CS 314

Spring 2019

1 List of topics

- Parsing
- Dynamic programming
- Python
- Lambda calculus
  - beta reduction
  - alpha conversion
  - eta conversion
  - variables, abstractions, applications
  - normal forms
  - evaluation order
  - Y-combinator
  - Church booleans
  - Church numerals
- Racket
- Functional programming
  - higher-order functions
  - map and reduce
  - filter
  - anonymous functions
- Haskell
  - type constructors
  - map, foldr, filter

2 Review questions

2.1 General

1. What does it mean to say a language is statically typed? Dynamically typed?
2. What is one advantage of using a dynamically typed language? One disadvantage?
3. What is one advantage of using a statically typed language? One disadvantage?

2.2 Parsing

4. What is a DFA?
5. How could you use a DFA to parse an input program?
6. What is operator precedence?
7. What is operator associativity?
8. What is a parse tree?

2.3 Python

9. What is a REPL?
10. What is a list comprehension?
11. Write a list comprehension that applies a function f to every element in a list l.

2.4 Lambda calculus

12. What are the three kinds of lambda expressions?
13. What is beta reduction?
14. What is alpha conversion and why do we need it?
15. Give an example of variable capture.
16. What are free and bound variables?
17. What is a normal form?
18. Do all lambda expressions have a normal form? If so, why? If not, give an example of one that doesn't.

19. What are applicative order and normal order?
20. Does it matter which order an expression is evaluated in?
21. What is the Y-combinator and what is it used for?
22. How can boolean values be defined using lambda calculus?
23. How can boolean functions be defined using lambda calculus?
24. How can integers be defined using lambda calculus?

2.5 Functional programming

25. What is one difference between imperative and functional programming?
26. What is one advantage of using a functional programming language? One disadvantage?
27. What is a higher-order function?
28. What is an anonymous function?

2.6 Racket

29. What is an S-expression?
30. What is a cons cell?
31. What is an atom? A pair? A list?
32. What do car and cdr do?
33. What do the map and reduce functions do?
34. Write a recursive function to compute the length of a list.
35. Write a function to compute the length of a list using reduce.

2.7 Haskell

36. What is lazy evaluation?
37. What is one advantage of using a lazily evaluated language? One disadvantage?
38. What is the filter function?
39. What is currying?