# COMP9024: Data Structures and Algorithms

Final Exam

Hui Wu

Term 1, 2019

http://www.cse.unsw.edu.au/~cs9024

1

1

## Final Exam (1/2)

- Start Time: 8:45 Tuesday 14/05
- Duration: 3 hours
- Venue: CSE labs
- Closed book
  - You cannot bring any material into the exam

2

2

## Final Exam (2/2)

- Two parts
  - Part 1: Basic data structures and algorithms
    - 8 questions (40 marks)
  - Part 2: Design and analysis of algorithms
    - 5 questions (60 marks)
- Descriptions of algorithms
  - Use pseudo code

3

3

## Topics Not Examinable

- C Programming Language
- Red-Black Trees
- Randomized algorithms

4

4

## Sample Questions in Part 1 (1/3)

What does a splay tree look like if its entries are accessed in increasing order of their keys?

5

5

## Sample Questions in Part 1 (2/3)

Draw the frequency array and Huffman tree for the following string:

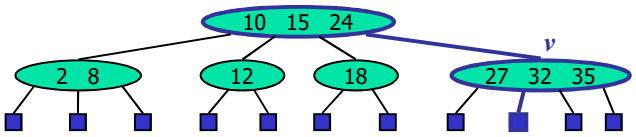
"dogs do not spot hot pots or cats"

6

6

## Sample Questions in Part 1 (3/3)

- Given a (2,4) tree shown below, draw the resultant (2,4) trees after deleting key 12 and inserting key 30, and the intermediate tree after each step (splitting, or fusion or transfer).




7

## Sample Questions in Part 2 (1/3)

Given a binary tree with n nodes where each node stores an entry (key, value), design an algorithm with the time complexity O(n) for testing if the tree is a search tree, and explain why your algorithm has O(n) time complexity.

8

8

## Sample Questions in Part 2 (2/3)

An vertex-weighted DAG (directed acyclic graph)is the DAG where each vertex has a weight. The length of a path from a vertex u to a vertex v is the sum of all the vertex weights of the path. A shortest path from a vertex u to a vertex v is the path with the minimum length. Describe an algorithm with O(m+n)time complexity for finding a shorted path from a vertex u to a vertex v in a vertex weighted DAG G, and show why your algorithm takes O(m+n) time.

9

9

## Sample Questions in Part 2 (3/3)

Show how to modify the KPM pattern matching algorithm so as to find every occurrence of a pattern string P that appears as a substring in T, while still running in O(n+m) time. (Be sure to catch even those matches that overlap.)

10

10