The University of Melbourne

Department of Computer Science and Software Engineering

433–521

Algorithms and Complexity

June 2006

Identical examination papers: None

Exam duration: Three hours

Reading time: Fifteen minutes

Length: This paper has 4 pages including this cover page.

Authorised materials: Foreign language dictionaries are permitted, in both book and electronic form.

Calculators: Permitted.

Instructions to invigilators: Students may *not* remove any part of the examination paper from the examination room. Students should be supplied with the exam paper and a script book, and with additional script books on request.

Instructions to students: This paper counts for 60% of your final grade. Please answer all questions in the script book provided, starting each question on a new page. Please write your student ID below and on your script book. When you are finished, place the exam paper inside the front cover of the script book.

Library: This paper is to be held in the Baillieu Library.

Student ID number:

Examiner's use only:

Total [60]:

Section A: Short Answer Questions (18 marks)

Answer each of the questions in this section *as briefly as possible*.

1. Finish the statement below, where $f(n)$ and $g(n)$ are functions of n.

   "A function $f(n)$ is $O(g(n))$ if and only if ..."

   [3 marks]

2. Which class $\Theta(g(n))$ does the following the function $f(n)$ belong to? Use the simplest $g(n)$ possible and explain why you chose this class:

   $f(n) = (n + 1)(3n - 2)$

   [3 marks]

3. The following summation describes the run-time of an algorithm. What is the complexity of the algorithm, in big-Oh terms?

   $$\sum_{i=0}^{\log n} \frac{1}{2^i}$$

   [3 marks]

4. Consider the abstract data type *dictionary*.

   (a) List 3 operations that must be defined for *every* dictionary.

   (b) List 1 additional operation, and state whether it needs to be defined for *every* dictionary, or whether it is optional.

   [4 marks]

5. You have developed a new algorithm that processes an array. The algorithm examines every item in the array as it rearranges the items, divides the file into three equal parts, and then makes recursive calls to the two subarrays at the left and right ends of the array.

   (a) To what family (or families) of algorithms does this algorithm belong?

   (b) Write a recurrence that describes the cost, or time, T_n for this algorithm in terms of the number of input items n.

   [5 marks]

Section B: Algorithm Families and Complexity (13 marks)

In this section you are asked to demonstrate that you have gained a high-level understanding of different algorithm families and complexity theory.

6. This question is about brute force algorithms.

   (a) Describe *briefly* what a brute force algorithm is.

   (b) In what situations would it be appropriate to use a brute force algorithm? Name two situations.

   [3 marks]

Continue ...

7. This question is about search trees. Answer each part of the question as *briefly* as possible.

   (a) What is the biggest drawback to using a binary search tree?

   (b) What strategy or strategies can be used to overcome this limitation?

   (c) Name one search data structure that employs the strategy you have named.

[5 marks]

8. You are managing a software project that involves building a computer-assisted instrument for medical surgery. The exact placement of the surgical knife is dependent on a number of different parameters, usually at least 20, sometimes more.

   Your programmer has developed two algorithms for positioning the cutting tool, and is seeking your advice about which algorithm to use:

   > **Algorithm A** has three nested loops. There is an `if` condition at the start of each loop, that is usually *not* met in the inner two loops, so the algorithm has an average case run time of n, and a worst case of n^3, where n is the number of input parameters.
   >
   > **Algorithm B** sorts the parameters first, then uses two nested loops. In the inner loop, there is an `while` condition that is rarely met, so the algorithm has an average case run time of $n \log n$, and a worst case of n^2.

   Which algorithm would you favor for inclusion in the software? Explain *briefly* how you arrived at this decision.

[5 marks]

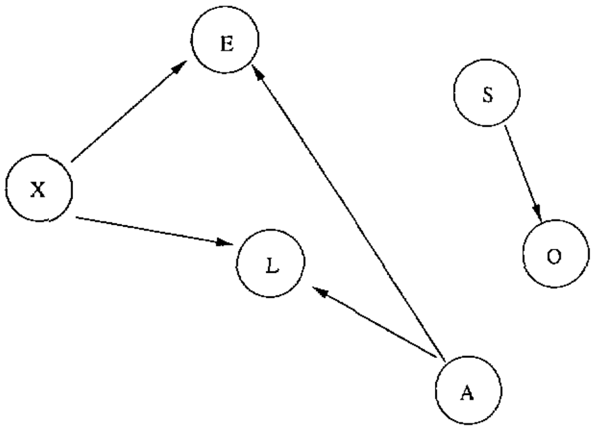
Section C: Method Questions (29 marks)

In this section you are asked to demonstrate that you understand how certain of the algorithms that we have studied work.

9. This question is about topological sorting of directed graphs.

   (a) Name, or describe *very briefly*, two algorithms that take as input a graph `G`, composed of vertices `V` and edges `E`, and output a topological sort of `G`.

   (b) Using either of these algorithms, show an output topological sort on the following directed graph:



[8 marks]

10. This question is about hashing.

    (a) Insert items with the following keys into a hash table of size 7:

    1 0 3 5

    Use the hash function $h(k) = k(k+3) \bmod 7$, and use linear probing for collision resolution. Show the hash value you have calculated for each input key, and show the hash table after all items have been inserted.

    (b) What happens to a hash table that uses linear probing for collision resolution when the number of inputs n is *equal* to the size of the hash table m? Answer *briefly*, considering both insertion and search.

    [7 marks]

11. In this question you are asked to use quicksort to *partially* sort the following letter keys. Using the rightmost key A (shown boxed) as pivot, partition the letters around this pivot. Subscripts show the order in which identical keys appear in the input data.

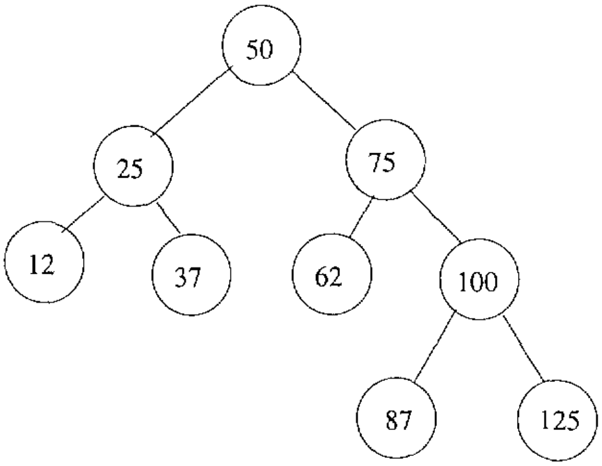
    $S_1\ A_1\ S_2\ P\ \ A_2\ R\ \ I\ \ L_1\ L_2\ \boxed{A_3}$

    (a) Show the contents of the array after this *one* partition, including the appropriate subscripts. Assume the conventional alphabetic ordering of letters, *i.e.* $A < I < L < P < R < S$

    (b) Would you have obtained a better partition using the median-of-three algorithm for choising the pivot element? The median-of-three method examines the first, last, and middle elements of the array, and uses the element with the middle *key value* as pivot.

    [7 marks]

12. The following AVL tree shows the numeric keys for items in the tree. Show the tree after insertion of an item with the key 150. Show or describe what step(s) you took to arrive at this new tree.



[7 marks]