THE AUSTRALIAN NATIONAL UNIVERSITY

Second Semester 2007

COMP3600/6466
(Algorithms)

Writing Period: 3 hours duration

Study Period: 15 minutes duration

Permitted Materials: None

Answer ALL Questions

All your answers must be written in the boxes provided in this booklet. You may be provided with scrap paper for working, but it must **not** *be used to write final answers.*

There is additional space at the end of the booklet in case the boxes provided are insufficient. Label such overflow boxes with the question number.

Do not remove this booklet from the examination room.

Name:

Student Number:

Official use only:

Q1 (20)	Q2 (15)	Q3 (10)	Q4 (15)	Q5 (15)	Q6 (15)	Q7 (10)

Q8 (10)	Total (100)

QUESTION 1 [20 marks]

(a) Order the following complexity classes from slowest-growing to fastest growing. If any of the classes are the same, indicate this fact and justify it.

$$\Theta(n^3),\ \Theta(\log n),\ \Theta(2^n),\ \Theta(n^2 \log^5 n),\ \Theta(\log_2(n^2)),\ \Theta(n^4/(2n+1))$$

QUESTION 1(a) (i)	**[3 marks]**

(b) Assume functions $f(n)$ and $g(n)$ are positive for large enough n. Either prove that the following statement is true, or provide a counterexample:

$$f(n) = O(g(n)) \implies e^{f(n)} = O(e^{g(n)})$$

QUESTION 1(b)	**[3 marks]**

(c) You are given three procedures that operate with singly-linked lists:

split(**In** : list; **Out1** : list; **Out2** : list)
– Divide list **In** into two lists **Out1** and **Out2** of equal size
(or as close to equal as possible if the number of nodes is odd)

cat(**In1** : list; **In2** : list; **Out** : list)
– Concatenate the lists **In1** and **In2**, in that order, into a single list **Out**.

isempty(**In** : list)
–Test if **In** is empty.

(i) Using the above procedures only, write a procedure that takes a list and reverses it.

QUESTION 1(c)(i) **[3 marks]**

(ii) Assume that **split** takes time proportional to the number of nodes of **In1**, while both **cat** and **isempty** take a constant amount of time. Write a recurrence for the running time of your procedure for input of n nodes. (You can assume that $n/2$ is always an integer.)

QUESTION 1(c)(ii) **[3 marks]**

(iii) Solve your recurrence using repeated substitution. Give the answer in terms of the $\Theta(\,)$ notation. (As before, assume dividing by 2 always gives an integer.)

QUESTION 1(c)(iii) (more space on next page) **[4 marks]**

QUESTION 1(c)(iii) (continued)

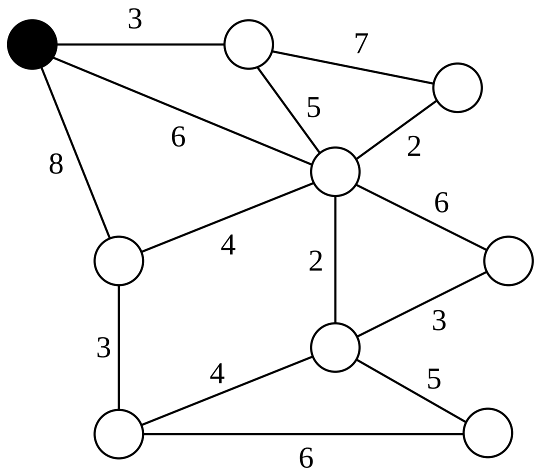
(d) Estimate the following summation. Express your answer in terms of the $\Theta(\,)$ function. Hint: the terms in the sum are increasing.

$$\sum_{k=2}^{n} \frac{k^3}{\lg k}$$

QUESTION 1(d)	**[4 marks]**

QUESTION 2 [15 marks]

(a) Use Prim's algorithm to find a minimum spanning tree on the following graph, starting with the black vertex. Indicate the order in which the edges are chosen.



QUESTION 2(a) (more room on next page) **[10 marks]**

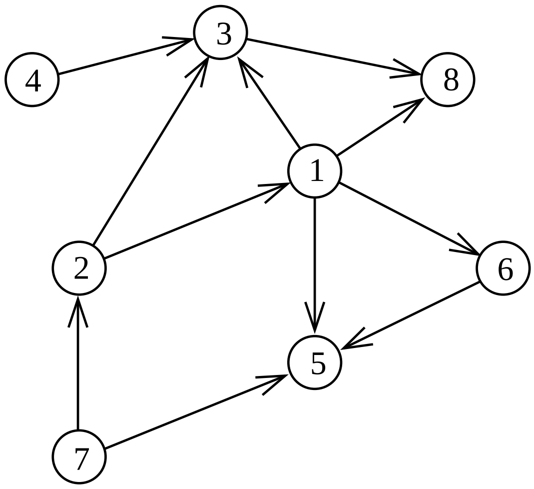
QUESTION 2(a) (continued)

(b) Suppose you have a connected graph G with edge weights, and let e be an edge with the maximum weight (there might be others also of maximum weight). Under what circumstances can you be sure that G has a minimum spanning tree that doesn't use e? Prove your answer.

QUESTION 2(b) **[5 marks]**

QUESTION 3 [10 marks]

(a) Perform Depth-first Search on the following digraph. Show the result as a redrawn graph with edges classified as tree edges, forward edges, back edges and cross edges. Also show the discovery and finishing times for each vertex.



QUESTION 3(a)	**[7 marks]**

(b) Using the results of the part (a), list the vertices in topological order.

QUESTION 3(b) **[3 marks]**

QUESTION 4 [15 marks]

(a) Consider a sequence of integers, for example:

$$6, 4, 7, 6, 5, 3, 8, 5, 7.$$

A *smooth* subsequence is one with the property that elements adjacent in the subsequence differ in value by at most 1. For example, the underlined elements form one example of a subsequence (not necessarily the longest):

$$\underline{6}, 4, \underline{7}, \underline{6}, \underline{5}, 3, 8, \underline{5}, 7.$$

(i) Give a dynamic programming solution to this problem. Assume the input is a sequence of integers $x_1, x_2, \ldots, x_n$. The output should be the length of the longest smooth subsequence.

QUESTION 4(a)(i) (more space on next page) **[10 marks]**

QUESTION 4(a)(i) (continued)

(ii) What is the running time of your solution as a function of n?

QUESTION 4(a)(ii) **[2 marks]**

(iii) Briefly indicate what extra needs to be done to find an example of a longest smooth subsequence rather than just its length.

QUESTION 4(a)(iii) **[3 marks]**

QUESTION 5 [15 marks]

(a) (i) Describe two common probing methods for hash tables with open addressing.

QUESTION 5(a)(i) **[5 marks]**

(ii) What are the average and worst case running times for insertion and lookup in hash tables? Explain why the average and worst case times are different.

QUESTION 5(a)(ii) **[5 marks]**

(b) One way to find the components of an undirected graph is to use the disjoint-union data structure that employs the weighted-union and path-compression heuristics.

Use this method to find the components of the graph with the following edges:

$$\{a, b\},\ \{c, d\},\ \{c, e\},\ \{f, g\},\ \{d, e\},\ \{b, g\},\ \{b, f\},$$

Show the state of the data structure after each edge is included.

QUESTION 5(b)	**[5 marks]**

QUESTION 6 [15 marks]

(a) (i) Create a binary search tree by inserting the following keys starting with an empty tree:

$$7,\ 4,\ 9,\ 2,\ 3,\ 10,\ 6.$$

You only need to show the tree at the end of all the insertions, but you should have made the insertions in the given order.

QUESTION 6(a)(i)	**[5 marks]**

(ii) Given the tree you constructed, delete the key 4 using the standard algorithm. Describe the intermediate steps.

QUESTION 6(a)(ii)	**[5 marks]**

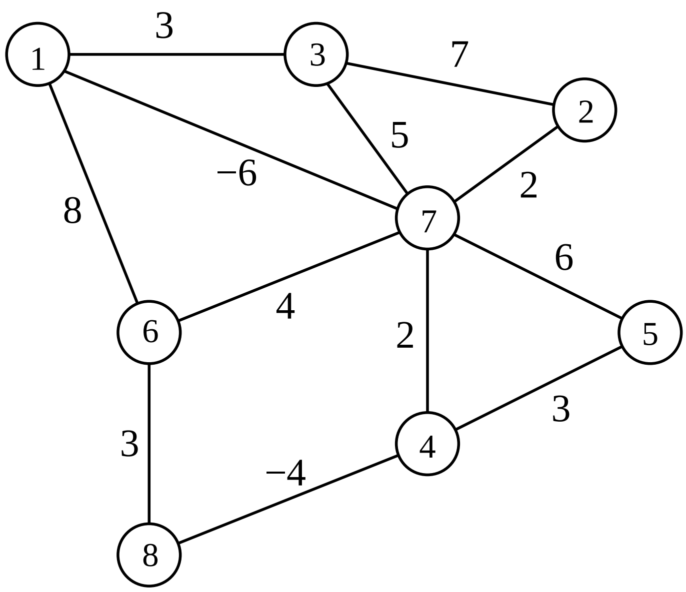
(b) What is the advantage of using a red-black tree rather than an ordinary binary search tree? Be explicit.

QUESTION 6(b)	**[5 marks]**

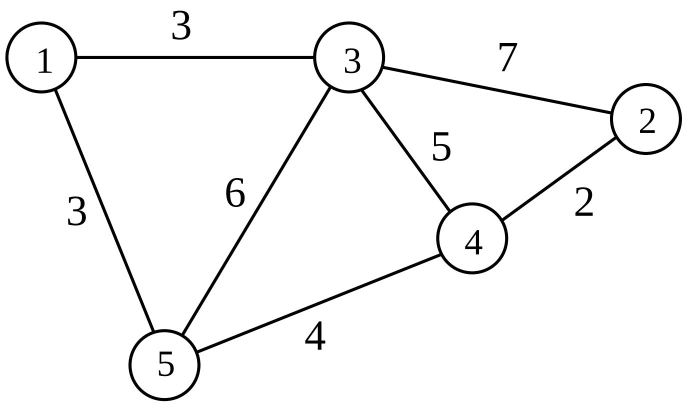
QUESTION 7 [10 marks]

Answer **either part (a) or part (b)**.

(a) Apply either the Dijkstra algorithm or the Bellman-Ford algorithm to find the distance from vertex 1 to each of the other vertices in the following graph. Explain your choice of which of the two algorithms you are using. Give enough intermediate information to demonstrate that you are applying the algorithm correctly (don't just give the final result).



(b) Apply the Floyd-Warshall algorithm to find the distance between each pair of vertices in the following graph. Give the state of the table after each iteration of the outer loop.



QUESTION 7 (a) or (b) (more room on next page) **[10 marks]**

QUESTION 7 (a) or (b) (continued)

QUESTION 8 [10 marks]

This question is only for COMP6466/PhD/Honours students.

Consider an undirected graph with n vertices and m edges.

Explain how the diameter of such a graph can be obtained using
(a) Breadth-first search,
(b) Fast matrix multiplication.

In each case, what is the running time?

QUESTION 8 (more room on next page) **[10 marks]**

QUESTION 8 (continued)

Additional answers. Clearly indicate the corresponding question and part.

Additional answers. Clearly indicate the corresponding question and part.

Additional answers. Clearly indicate the corresponding question and part.

Additional answers. Clearly indicate the corresponding question and part.