

**THE AUSTRALIAN NATIONAL UNIVERSITY**

*Second Semester 2007*

**COMP1140  
(Data Structures and 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 (20)	Q3 (20)	Q4 (20)	Q5 (20)	Total (100)

**QUESTION 1 [20 marks]**

(a) Let  $f(n)$ ,  $g(n)$  and  $h(n)$  be functions that are positive for large enough  $n$ . For each of the following, either prove the claim or provide a counterexample.

(i) If  $f(n) = O(g(n))$  and  $f(n) = O(h(n))$ , then  $f(n) = O(g(n) + h(n))$ .

QUESTION 1(a)(i)

[5 marks]

(ii) If  $f(n) = O(g(n))$ , then  $e^{f(n)} = O(e^{g(n)})$ .

QUESTION 1(a)(ii)

[4 marks]

(b) 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(b)(i)

[4 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. What is the running time for your reversal procedure? Justify your answer.

QUESTION 1(b)(ii) (more space over page)

[7 marks]

QUESTION 1(b)(ii) (continued)

## QUESTION 2 [20 marks]

(a) In how many ways can a committee of 12 be formed from 10 men and 8 women candidates if

(i) there are no restrictions,

QUESTION 2(a)(i)	[2 marks]
------------------	-----------

(ii) there must be more women than men?

QUESTION 2(a)(ii)	[2 marks]
-------------------	-----------

(b) How many 8-bit bytes (strings of 0s and 1s) contain

(i) exactly two 1s,

QUESTION 2(b)(i)	[2 marks]
------------------	-----------

(ii) at least six 1s?

QUESTION 2(b)(ii)	[2 marks]
-------------------	-----------

(c) In how many ways can eight identical balls be distributed into three containers so that

(i) no container is empty,

QUESTION 2(c)(i)	[2 marks]
------------------	-----------

(ii) the third container contains an odd number of balls?

QUESTION 2(c)(ii)	[2 marks]
-------------------	-----------

(d) What's the probability that

(i) there's at least one 6 out of four throws of a fair 6-sided die,

QUESTION 2(d)(i)

[2 marks]

(ii) two cards dealt randomly from a standard 4-suit, 52-card deck are of the same suit?

QUESTION 2(d)(ii)

[2 marks]

(e) In a given population  $1/100$  of the people are X-positive. A test for X comes back positive with probability .9 when administered on a person who is X-positive, and with probability .2 otherwise. What's the probability that

(i) a test on a random person comes back positive,

QUESTION 2(e)(i)

[2 marks]

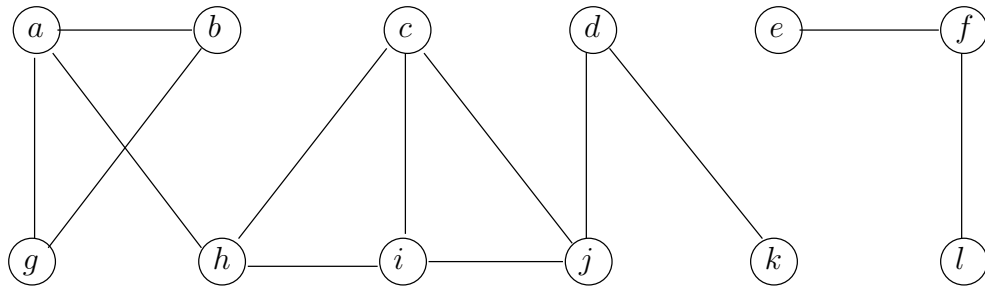
(ii) a random person is X-positive given that their test comes back positive?

QUESTION 2(e)(ii)

[2 marks]

### QUESTION 3 [20 marks]

Parts (a), (b) and (c) of this question refer to the following graph  $G$ :



- (a) Use the `find_tree` and `merge_tree` procedures to find the connected components of the graph  $G$ . Initially each vertex of  $G$  forms a tree. Add the edges in the following order:  $(a, b)$ ,  $(c, h)$ ,  $(d, k)$ ,  $(e, f)$ ,  $(f, l)$ ,  $(a, h)$ ,  $(i, j)$ ,  $(d, j)$ ,  $(h, i)$ ,  $(c, i)$ ,  $(c, j)$ ,  $(a, g)$ ,  $(g, b)$ .

Show the trees as they appear after each edge addition.

QUESTION 3(a)	[5 marks]

QUESTION 3(a) (continued)

(b) Apply DFS to  $G$ .

(i) Show the resulting trees.

(ii) For the largest component  $C$  of  $G$ :

1. Give the discovery time for each vertex;
2. Give the lowpoint for each vertex;
3. State the rule that identifies bridges according to the lowpoint. Apply it to find the bridges of  $C$ .

*Reminders:*

A *bridge* of a graph  $G$  is an edge whose removal disconnects  $G$ .

For a graph  $G$ , let  $d[u]$  be the DFS discovery time of  $u$  for each vertex  $u$  in  $G$ . For a given vertex  $v$ , the *lowpoint* of  $v$  is the least number  $d[u]$  of a vertex  $u$  which can be reached from  $v$  via a, possibly empty, path starting with tree edges in the direction away from the root, followed by at most one back edge.

QUESTION 3(b)(ii)

[5 marks]

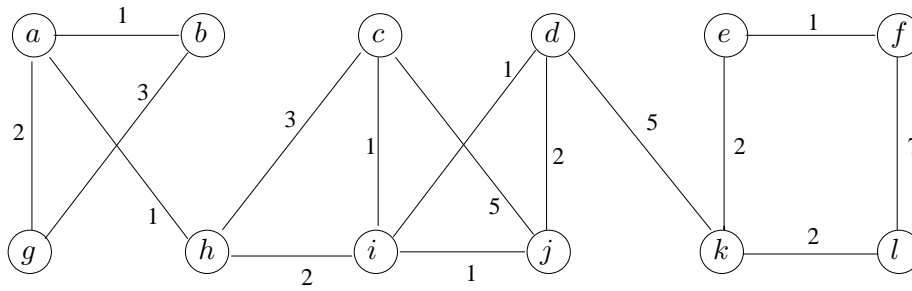
QUESTION 3(b)(ii) (continued)

- (c) Apply BFS to  $G$  starting at vertex  $a$ : Show the order in which the vertices are visited together with their current distance from  $a$ .

QUESTION 3(c)

[5 marks]

(d) Show the minimum spanning tree found by Prim's algorithm applied to the weighted graph below. Indicate the order in which the edges have been added.



QUESTION 3(d)

[5 marks]

#### QUESTION 4 [20 marks]

- (a) A set of logical operators is *complete* if every boolean function can be represented as a formula built using only the operators in the set. Let  $\oplus$  be a logical operator defined as follows:  $x \oplus y$  is true if both  $x$  and  $y$  are false, otherwise it is false. Prove that the set  $\{\oplus\}$  is complete.

QUESTION 4(a)

[5 marks]

(b) Let  $F$  be the formula

$$(a \wedge b) \vee (\neg b \wedge c) \vee (\neg a \wedge c) \vee \neg c.$$

Show that  $F$  is a tautology using the resolution proof method.

QUESTION 4(b)

[5 marks]

- (c) *Unit resolution* is a restricted form of resolution where one of the resolved clauses is a unit clause, i.e., a clause with a single element. Unit resolution is incomplete for refutation. That is, some unsatisfiable formulas cannot be refuted using unit resolution alone. Show a formula in clausal form such that it can be refuted using the (unrestricted) resolution method but not with unit resolution. Justify your answer (it is not enough to just give the formula)

QUESTION 4(c)

[5 marks]

(d) In the following questions, we assume that variables are ordered according to their indices, i.e.,  $x_i < x_j$  if  $i < j$ .

(i) Give the reduced BDD of the following formula:

$$(x_1 \wedge x_2) \vee (x_3 \wedge x_4) \vee (x_5 \wedge x_6).$$

QUESTION 4(d)(i)	[2 marks]
------------------	-----------

(ii) In general, given a formula

$$(x_1 \wedge x_2) \vee (x_3 \wedge x_4) \vee \cdots \vee (x_{2n-1} \wedge x_{2n})$$

for some  $n \geq 1$ , how many nodes does its reduced BDD have?

QUESTION 4(d)(ii)	[3 marks]
-------------------	-----------

## QUESTION 5 [20 marks]

- (a) What is an appropriate database scheme for a real estate agency, including all the information normally required about managed property, tenants and lessors, buyers and sellers, managing agents.

QUESTION 5(a)

[4 marks]

- (b) Suggest what attributes, if any, could serve as keys for your relations from question 1.

QUESTION 5(b)

[3 marks]

(c) A real estate database should be designed to handle the following sorts of queries:

- a. What is the address of a given tenant?
  - b. What is the rent of a given tenant?
  - c. Who is the lessor of a given property?
  - d. Which 2 bedroom apartments are available in Braddon?
  - e. Which tenants live in the Inner North?
- (i) Suggest appropriate primary index structures for the relations you designed in question 1 using a hash table in each case.

QUESTION 5(c)(i)	[2 marks]
------------------	-----------

- (ii) Tell how many buckets you suggest as well as what the domain attribute(s) are.

QUESTION 5(c)(ii)	[1 mark]
-------------------	----------

- (iii) How many of these type of queries can you answer efficiently, i.e., in average time  $O(1)$  independent of the size of the relations?

QUESTION 5(c)(iii)	[2 marks]
--------------------	-----------

(d) Suppose the real estate agency wants to be able to answer the following types of queries efficiently, i.e., much faster than by searching entire relations:

- a. Which agent is in charge of a given tenant?
- b. Which lessor earns more than \$1500pw in rent?
- c. Which properties might become available for rent in November?
- d. Which suburb has the highest sales value on average?

What would you pick as primary and secondary indexes for your relations. Explain how you would answer each type of query.

QUESTION 5(d)	[2 marks]
---------------	-----------

(e) Express two queries of question 3 and two queries of question 4 in relational algebra

QUESTION 5(e)	[4 marks]
---------------	-----------

(f) Prove or disprove the following:

$$\pi_L(R \bowtie S) \equiv (\pi_L(R) - \pi_L(S))$$

QUESTION 5(f)	<b>[3 marks]</b>

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.

Additional answers. Clearly indicate the corresponding question and part.

Additional answers. Clearly indicate the corresponding question and part.