UNIVERSITY OF SOUTHAMPTON COMP1201W1

SEMESTER 2 EXAMINATION 2012/2013

ALGORITHMICS

Duration: 120 mins

You must enter your Student ID and your ISS login ID (as a cross-check) on this page. You must not write your name anywhere on the paper.

Student ID:

ISS ID:

Question	Marks
1	
2	
3	
4	
Total	

Answer the question in section A and TWO *questions out of* THREE *in section B.*

This examination is worth 75%. The tutorials were worth 25%.

University approved calculators MAY be used.

Each answer must be completely contained within the box under the corresponding question. No credit will be given for answers presented elsewhere.

You are advised to write using a soft pencil so that you may readily correct mistakes with an eraser.

You may use a blue book for scratch—it will be discarded without being looked at.

 Number of pages 14

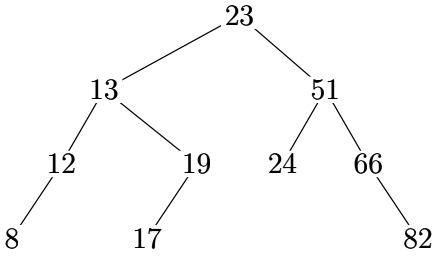
Section A

Question A 1

(a) Draw the binary search tree produced when inserting 66, 33, 70, 28, 46, 21, 73, 97, 52, 78. *(2 marks)*

2

(b) Draw the tree obtained by deleting 13 from the binary search tree shown. *(2 marks)*




```
delete(13)
```

2

(c) What is the expected time complexity of adding a new entry to a binary tree? *(1 marks)*

1

(d) Heaps use a binary tree encoded into an array. Show the binary tree represented by the following array.

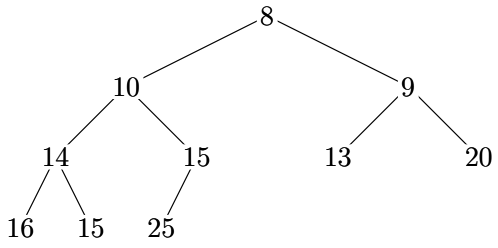
3	4	3	20	19	29	10	24	28	21

(1 marks)

1

(e) Show what happens to the heap shown on the left when you add 2. *(2 marks)*

Before

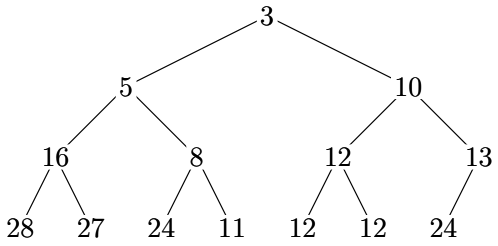


After `add(2)`

2

(f) Show what happens to the heap shown on the left when you remove the minimum entry. *(2 marks)*

Before

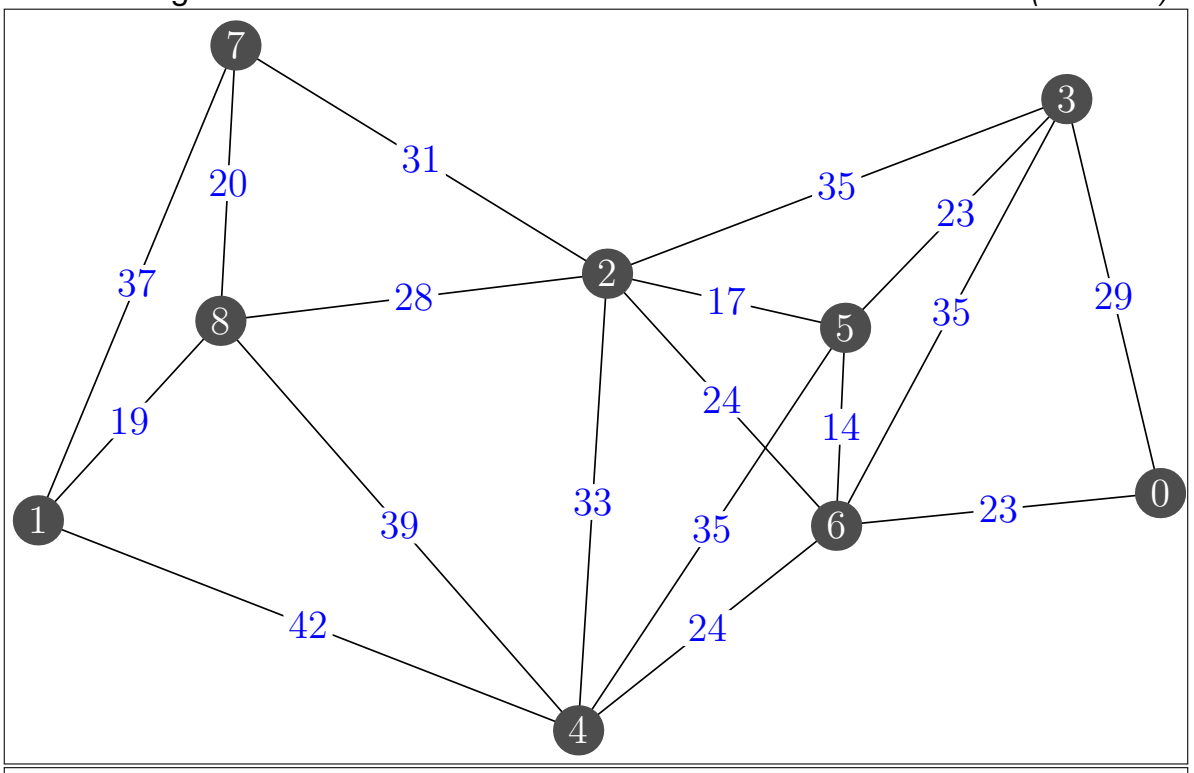


After `removeMin()`

2

TURN OVER

(g) Highlight the edges of the minimum spanning tree found by Prim's algorithm for the graph below and write down the edges in the order they are found starting from node 0. *(5 marks)*



1.	**2.**
3.	**4.**
5.	**6.**
7.	**8.**

15

(h) Compute a Huffman tree for the following alphabet.

Letter	a	b	c	d	e	f	g
Frequency	13	1	3	5	11	17	15

(4 marks)

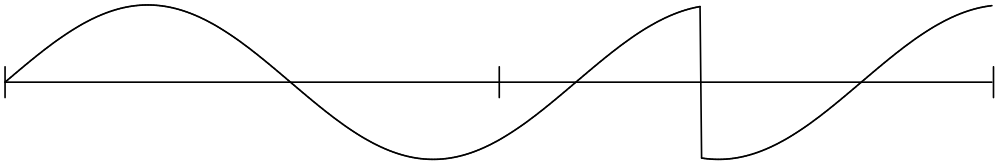
$\overline{4}$

(i) How would the word “deface” be coded in your Huffman tree. *(1 marks)*

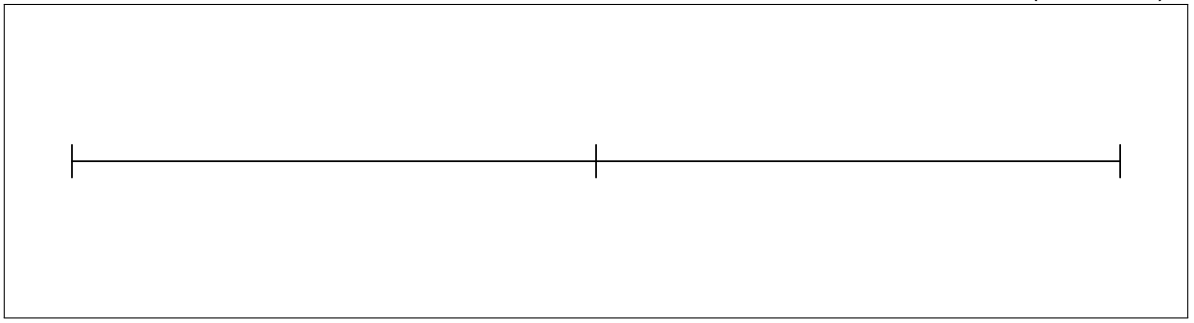
$\overline{1}$

TURN OVER

(j) Sketch the first Haar transform of the signal shown below.



(5 marks)



5

End of question 1

Q1: (a) $\frac{}{2}$ (b) $\frac{}{2}$ (c) $\frac{}{1}$ (d) $\frac{}{1}$ (e) $\frac{}{2}$ (f) $\frac{}{2}$ (g) $\frac{}{5}$ (h) $\frac{}{4}$ (i) $\frac{}{1}$ (j) $\frac{}{5}$ Total $\frac{}{25}$

Section B

Question B 2

(a) We can implement a fast set for a fixed number of integers using two arrays. Below we show the representation of the set $\{7, 3, 1, 4\}$.

	0	1	2	3	4	5	6	7	8	9
indexArray	-1	2	-1	1	3	-1	-1	0	-1	-1
memberArray	7	3	1	4						

Show the state of the arrays when we add 9 to the set. *(3 marks)*

	0	1	2	3	4	5	6	7	8	9
indexArray										
memberArray										

3

(b) Show the state of the arrays when you remove 3 from the set shown above. *(3 marks)*

	0	1	2	3	4	5	6	7	8	9
indexArray										
memberArray										

3

(c) Fill in the implementation of `add` and `remove`. *(8 marks)*

```
public class FastSet extends AbstractSet<Integer> {
    private int[] indexArray;
    private int[] memberArray;
    private int noMembers;

    public FastSet(int n) {
      indexArray = new int[n];
      memberArray = new int[n];
      for(int i=0; i<n; i++)
         indexArray [i] = -1;
      noMembers = 0;
    }

    public boolean add(int i) {









    }

    public boolean remove(int i) {

8

    }
```

(d) Implement an iterator class for the FastSet. *(11 marks)*

```
public Iterator<Integer> iterator() {
        return new FastSetIterator();
}

private class FastSetIterator implements Iterator<Integer> {

}
```

$\frac{}{11}$

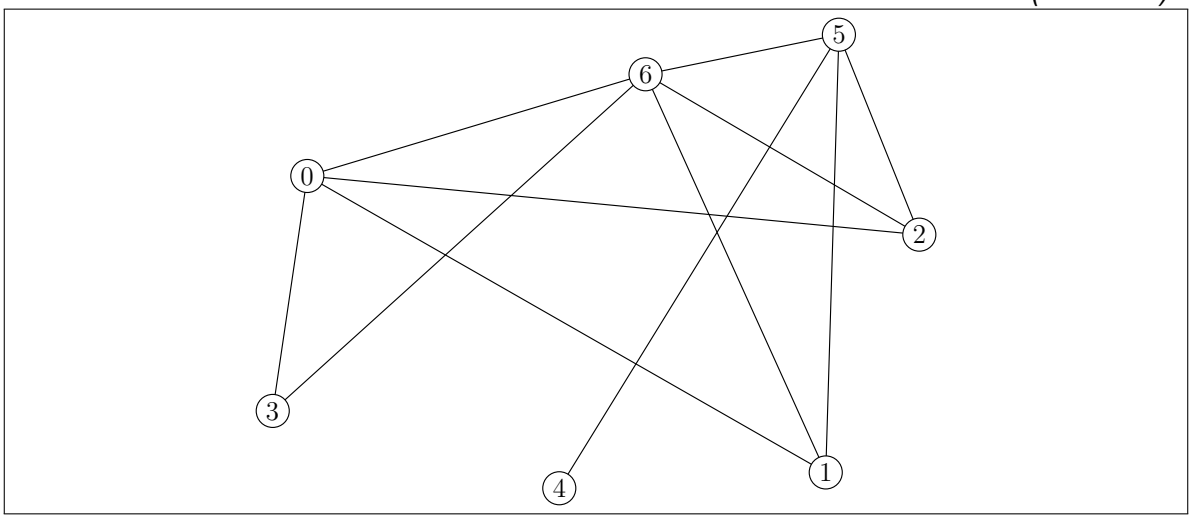
End of question 2

Q2: (a) $\frac{}{3}$ (b) $\frac{}{3}$ (c) $\frac{}{8}$ (d) $\frac{}{11}$ Total $\frac{}{25}$

TURN OVER

Question B 3

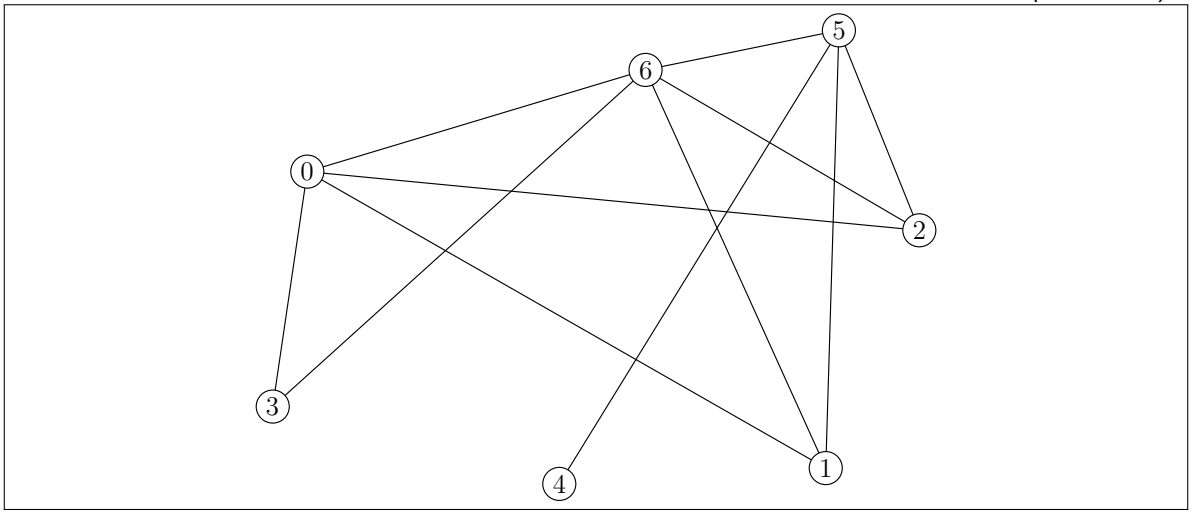
(a) Draw the edges on the graph used to find the vertices using **breadth first search** starting from vertex 0 where the lower numbered vertices are searched first. Write the order in which the vertices are discovered. *(5 marks)*



5

bfs order = _____

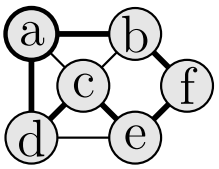
(b) Draw the edges on the graph used to find the vertices using **depth first search** starting from vertex 0 where the lower numbered vertices are searched first. Write the order in which the vertices are discovered. *(5 marks)*



5

dfs order = _____

(c) Write pseudo code for a recursive backtracking algorithm to solve a Hamiltonian circuit problem (see example below). Denote the graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the vertex set and $\mathcal{E}$ is the edge set. Let, $\mathcal{H}$ be the Hamiltonian cycle and `startVertex` be the starting vertex.



(8 marks)

8

(d) Describe the branch and bound strategy. *(3 marks)*

$\overline{3}$

(e) Describe fours ways in which branch and bound can be modified to reduce the proportion of search space needed to find an optimal solution of a Euclidean TSP. *(4 marks)*

1

2

3

4

$\overline{4}$

End of question 3

Q3: (a) $\frac{}{5}$ (b) $\frac{}{5}$ (c) $\frac{}{8}$ (d) $\frac{}{3}$ (e) $\frac{}{4}$ Total $\frac{}{25}$

Question B 4 Merge sort has the form

```
MERGESORT(a[1:n]) {
  if (n>1) {
    b ←a[1:n/2]
    c ←a[n/2+1:n]
    MERGESORT(b)
    MERGESORT(c)
    MERGE(b,c,a)
    }
  }
```

The number of comparison operations to merge two arrays of length $n/2$ is n.

(a) Let $T(n)$ be the number of comparison operations. Write down a recurrence relation for $T(n)$ valid if $n = 2^m$. *(4 marks)*

$T(n) =$

4

(b) Write down the boundary condition $T(1)$ and use the recurrence relation to compute $T(2)$, $T(4)$, and $T(8)$ *(4 marks)*

$T(1) =$

$T(2) =$

$T(4) =$

$T(8) =$

4

(c) Demonstrate, for $n = 2^m$, that $f(n) = n \log_2(n)$ satisfies the recurrence relation in part (a). *(6 marks)*

6

TURN OVER

(d) Explain why the java collections class uses Merge short, while for arrays of primitive types java uses Quick sort. *(4 marks)*

$\overline{4}$

(e) Write pseudo code for quick sort, explaining briefly what each method does. *(7 marks)*

$\overline{7}$

End of question 4

Q4: (a) $\overline{4}$ (b) $\overline{4}$ (c) $\overline{6}$ (d) $\overline{4}$ (e) $\overline{7}$ Total $\overline{25}$

END OF PAPER