UNIVERSITY OF SOUTHAMPTON MALAYSIA COMP1201

SEMESTER 2 EXAMINATION 2021/2022

ALGORITHMICS

DURATION 120 MINS (2 Hours)

This paper contains TWO sections.

Answer ALL questions from section A and only ONE of the THREE questions from section B.

Make sure your answers are clear and readable.

Write down thoughts and intermediate steps so you can get partial credit.

General Notes:

- The exam is CLOSED-BOOK. University approved calculators MAY be used.
- Cell phones and calculators are strictly NOT allowed.

7 pages examination paper

SECTION A
(75 marks)

INSTRUCTION
This part consists of **FIVE** (**5**) questions. Answer **ALL** the questions.

A1. Consider inserting data with integer keys 129, 315, 912, 439, 239, 328, 555, 101 in that order into a hash table of size 11, where the hash function is identity, $h(key) = key \,\%11$. Show the resulting hash table after doing the insertion. Resolve collisions using the following methods.

a) Separate Chaining

[4 marks]

b) Linear probing

[4 marks]

c) Quadratic probing

[4 marks]

A2. Consider inserting data with integer keys 45, 36, 54, 27, 63, 72, 61,18, 60 in that order into following empty data structures. Show all steps and clearly indicate the final tree.

a) Binary search tree

[4 marks]

b) Min-heap

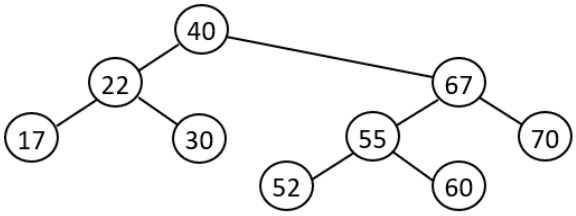
[6 marks]

c) AVL tree

[8 marks]

d) In the AVL tree shown below, show the AVL tree resulting from deleting 30, then 40, then 67.

[7 marks]



A3. **Max-Heap** uses a binary tree encoded into an array. Show the binary tree represented by the following array and then fix the max-heap property.

Array Index	1	2	3	4	5	6	7	8	9	10
Array element	70	10	60	50	20	30	40	90	80	100

[8 marks]

A4. The quicksort algorithm is as follows

```
QuickSort(a, start, end) {
   if (start < end)
        pivot ← ChoosePivot(a, start, end)
        part ← Partition(a, pivot, start, end)
        QuickSortT (a, start, part)
        QuickSortT(a, part + 1, end)
   else
        return
}
```

a) Assume that `Partition` algorithm takes n operations and that the pivot splits the array exactly in half at each step. Write a recursion relation for the number of partitioning operations, $T(n)$.

[4 marks]

b) Show that $T(n) = n\,log_2(n)$ satisfies the recursion relation in part (a).

[6 marks]

c) Modern sorting algorithms are often hybrids that combine practical features of different sorting algorithms. Explain why the quicksort algorithm can use insertion sort? Modify the `QuickSort` pseudocode to accommodate insertion sort.

[4 marks]

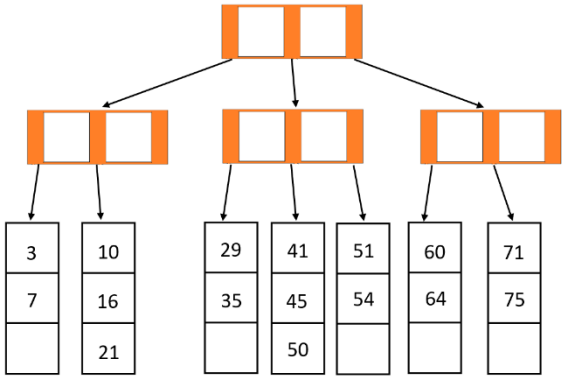
d) Provide the worst-case complexity for quicksort and explain why this does not discourage people from using it.

[4 marks]

A5. B-Trees use a multi-way tree to minimise the number of data accesses to secondary storage, such as disks.

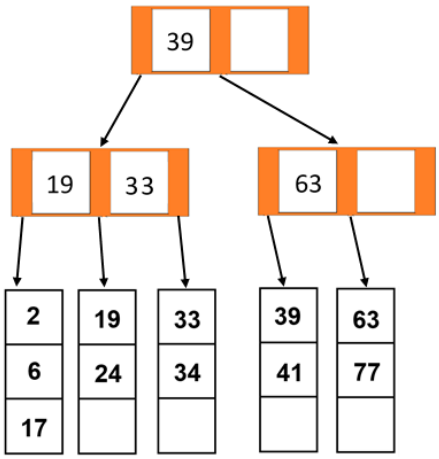
a) In the B-Tree shown below, write in the appropriate values for the internal nodes.

[4 marks]



b) Insert 8 into the B-Tree shown below. Draw the resulting tree.

[8 marks]



SECTION B
(25 marks)

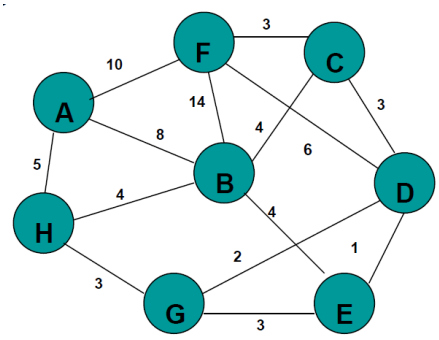
INSTRUCTION
This part consists of **THREE (3)** questions. Answer **only ONE** question.

B1. Graph theory is the study of graphs, which are mathematical structures used to model pairwise relations between objects

a) There are 25 telephones at UoSM. Is it possible to connect them with wires so that each telephone is connected with exactly 7 others? Justify your answer.

[5 marks]

b) Consider the below input graph



Find the minimum spanning tree of the given graph using:

i. Kruskal's algorithm

[10 marks]

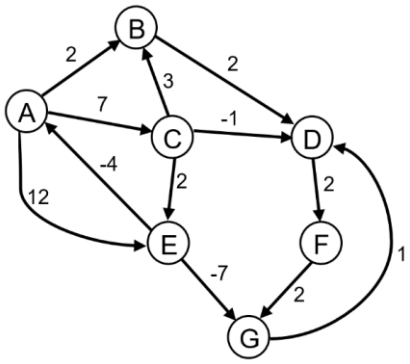
ii. Prim's algorithm

[10 marks]

Notes:

- The steps of each algorithm MUST be described.
- It is not required to draw all intermediate graphs. Explanation of each step and drawing the final graph will be acceptable

B2. Dijkstra's algorithm is an algorithm for finding the shortest paths between nodes in a graph, which may represent, for example, road networks. Consider the following directed, weighted graph:



a) Even though the graph has negative weight edges, step through Dijkstra's algorithm to calculate supposedly shortest paths from **A** to every other vertex. Cross out old values and write in new ones, from left to right within each cell, as the algorithm proceeds. Also, list the vertices in the order which you marked them known.

[10 marks]

b) Dijkstra's algorithm found the wrong path to some of the vertices. For just the vertices where the wrong path was computed, indicate both the path that was computed and the correct path

[6 marks]

c) What single edge could be removed from the graph such that Dijkstra's algorithm would happen to compute correct answers for all vertices in the remaining graph?.

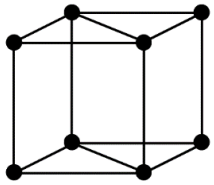
[4 marks]

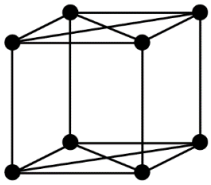
d) Suppose we define a different kind of graph where we have weights on the vertices and not the edges. Does the shortest-paths problem make sense for this kind of graph? If so, give a precise and formal description of the problem. If not, explain why not. Note we are not asking for an algorithm, just an explanation of the problem is or that it makes no sense.

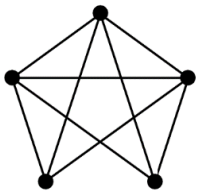
[5 marks]

B3. Eulerian graphs were first discussed by Leonhard Euler while solving the famous Seven Bridges of Königsberg problem in 1736.

a) For each of the following four graphs, decide if it has an Euler trail, and an Euler circuit, both, or neither. Give a brief justification in each case

i. 

ii. 

iii. 

iv. The complete graph K_n for $n \geq 4$ even.

[10 marks]

b) The distances between various cities are given in the table below. Draw the corresponding graph to the table.

	Urbana	Gotham	Metropolis	Bedrock	Quahog
Urbana		1	2	3	3
Gotham	1		4	2	5
Metropolis	2	4		7	4
Bedrock	3	2	7		13
Quahog	3	5	4	13	

[5 marks]

c) How many Hamiltonian cycles exist in the graph above?

[4 marks]

d) What is the traveling salesman problem? Briefly explain why we don't simply do a brute force search for solutions.

[6 marks]

END OF PAPER

UNIVERSITY OF SOUTHAMPTON COMP1201W1

SEMESTER 2 EXAMINATION 2018/2019

ALGORITHMICS

Duration: 120 mins

Student ID:

This paper is a WRITE-ON examination paper. You ***must*** write your Student ID on this Page and must not write your name anywhere on the paper.

All answers should be written within the designated boxes in this examination paper and sufficient space is provided for each question.

If, for some reason, space is required to complete or correct an answer to a question, use the "Additional Space" provided on the facing page to the question. Clearly indicate which question the answer corresponds to.

No credit will be given for answers presented elsewhere and without clear indication of to what question they correspond. Blue answer books may be used for scratch; they will be discarded without being looked at.

Answer **ALL** parts of the question in **Section A** and **TWO** questions from **Section B**.

Section A carries 40% of the total marks for the exam paper.

Section B carries 60% of the total marks for the exam paper.

Question	Mark	*Arithmetic checked*	*Double marked*
Total			

Only University approved calculators may be used.

A foreign language dictionary is permitted ONLY IF it is a paper version of a direct "Word to Word" translation dictionary AND it contains no notes, additions or annotations.

Section A

Question A 1

(a) Show how the numbers $37, 34, 51, 26, 33, 81, 65, 79, 17$ would be hashed using a hash function $d_2 + 11d_1$ where d_1 is the least significant digit and d_2 the second digit (the one on the left). Show how these would be stored in a hash table using separate chaining.

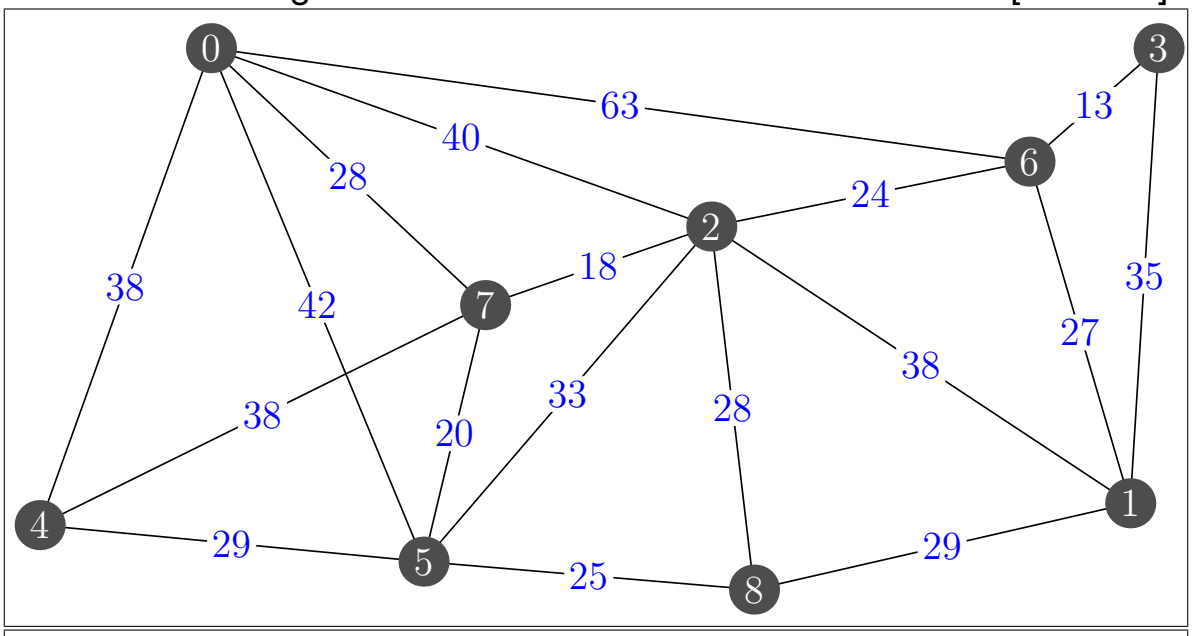
[10 marks]

d_2d_1	37	34	51	26	33	81	65	79	17
$d_2 + 11d_1$									
$(d_2 + 11d_1)\%10$									

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

Additional space. Do not use unless necessary. Clearly mark corresponding question.

(b) Highlight the edges of the minimum spanning tree found by Prim's algorithm for the graph below and write down the edges (i.e., source node - target node, such as $1 - 2$) in the order they are found starting from node 0. [8 marks]



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

Additional space. Do not use unless necessary. Clearly mark corresponding question.

(c) Write pseudo code for the Merge sort algorithm. What is its average and worst case time complexity? Explain why the Java collections class uses Merge sort, while for arrays of primitive types Java uses Quick sort. [12 marks]

Additional space. Do not use unless necessary. Clearly mark corresponding question.

(d) Describe in brief outline the standard simplex algorithm. What is its running time complexity? [10 marks]

End of question 1

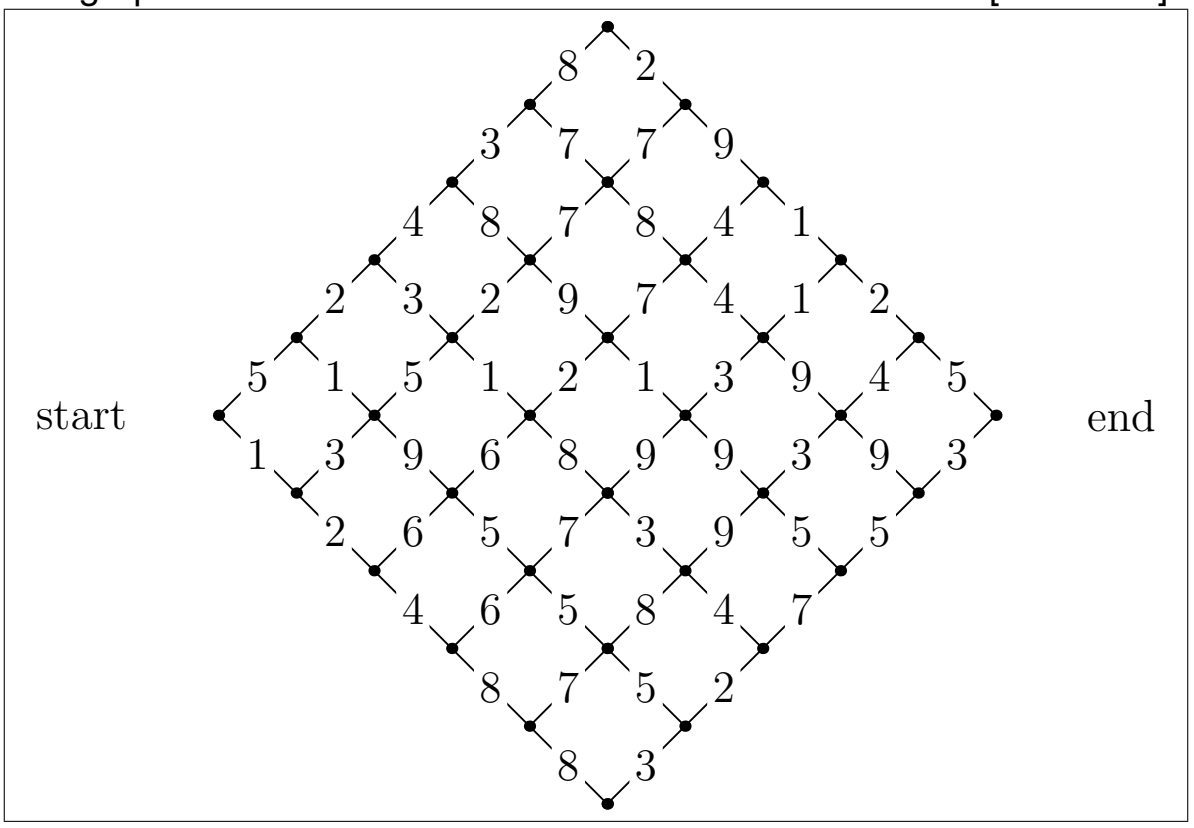
Q1: (a) $\frac{}{10}$	(b) $\frac{}{8}$	(c) $\frac{}{12}$	(d) $\frac{}{10}$	Total $\frac{}{40}$

Additional space. Do not use unless necessary. Clearly mark corresponding question.

Section B

Question B 2

(a) Use the dynamic programming forward algorithm to compute the minimum cost of each path from the left most node to each other node, where the cost of moving along an edge is equal to the number shown. An edge can only be traversed from left to right. Use the backwards algorithm to find the minimum cost path across the graph. [10 marks]



Additional space. Do not use unless necessary. Clearly mark corresponding question.

(b) Prove that in any Facebook community, there must exist two people who have the same number of friends. Mathematical equivalently, prove that any graph G contains two vertices of the same degree. [10 marks]

Additional space. Do not use unless necessary. Clearly mark corresponding question.

(c) When computing a maximum matching, what is the fundamental difference in searching for an augmenting path in bipartite graphs and general graphs? What changes does this difference bring to the algorithms searching for a maximum matching, in bipartite graphs and general graphs? [10 marks]

Additional space. Do not use unless necessary. Clearly mark corresponding question.

End of question 2

Q2: (a) $\frac{}{10}$ (b) $\frac{}{10}$ (c) $\frac{}{10}$ Total $\frac{}{30}$

Question B 3

(a) Which of the following statements are true and why? (Full marks will only be awarded if correct reasons are given). [10 marks]

(i) Insertion sort is a stable and in-place sorting algorithm.

(ii) Quicksort has the most efficient worst-case running time complexity among all the sorting algorithms.

Additional space. Do not use unless necessary. Clearly mark corresponding question.

(iii) For smaller arrays insertion sort is more preferred than other more sophisticated sorting algorithms.

(iv) The 1-pivot version of quicksort is always the fastest.

(v) The worst case running time complexity of any comparison based sorting algorithm is always $\Theta(n \log n)$.

Additional space. Do not use unless necessary. Clearly mark corresponding question.

(b) Given an array $A[1, \ldots, n]$ of n integers with $n > 5$. Let $A[i]$ denote the i^{th} element of the array (with $i = 0, 1, \ldots, n$). We also know that for any $0 \leq i \leq n/2 - 1$, we have $A[i] \leq A[2i]$ and $A[i] \leq A[2i + 1]$. Can we identify whether the smallest sum of 5 elements in A is smaller than 2019 with $O(\log n)$ running time complexity? Please explain your answer. [10 marks]

Additional space. Do not use unless necessary. Clearly mark corresponding question.

(c) Let $x_1, x_2, \ldots, x_n$ be integers (which can either be positive or negative, as well as 0). We would like to decide whether we can delete at most 10 numbers from this list such that the absolute value of all the remaining numbers is at most 2019. Propose a solution that only requires $O(n)$ comparisons. [10 marks]

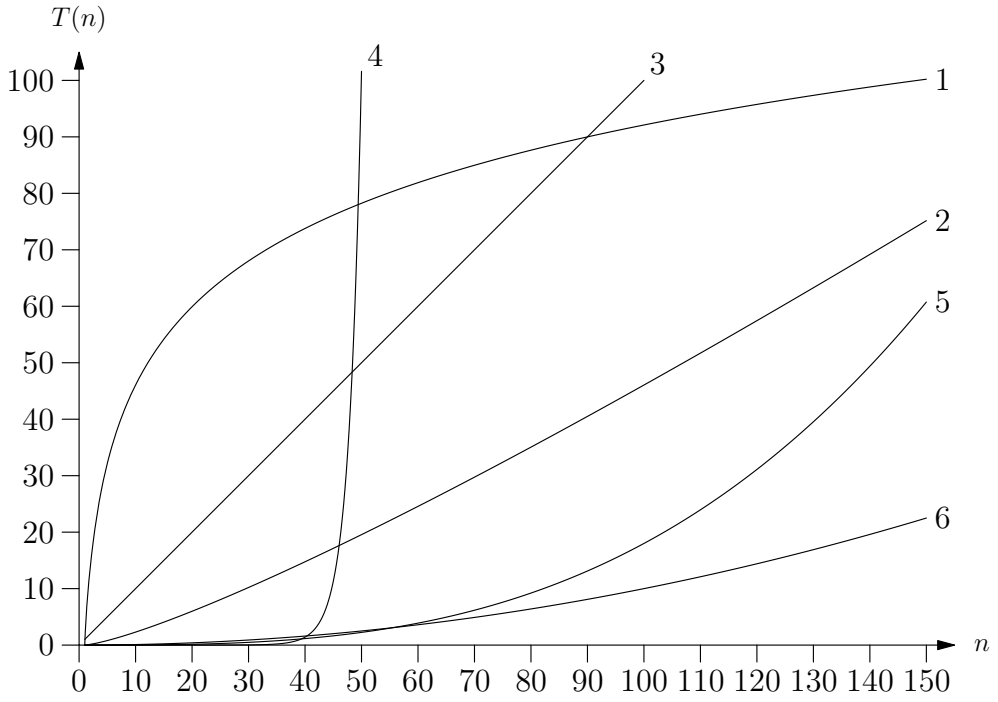
End of question 3

Q3: (a) $\frac{}{10}$ (b) $\frac{}{10}$ (c) $\frac{}{10}$ Total $\frac{}{30}$

Additional space. Do not use unless necessary. Clearly mark corresponding question.

Question B 4

(a) The graph below shows the time complexity for the following algorithms (a) $\Theta((n/a)!)$, (b) $\Theta(n^2)$, (c) $\Theta(n\ \log(n))$, (d) $\Theta(n)$, (e) $\Theta(n^3)$, and (f) $\Theta(\log(n))$. Match the time complexity classes with the curves on the graph. Please briefly explain your answers. [10 marks]



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

Additional space. Do not use unless necessary. Clearly mark corresponding question.

(b) Suppose that there is an algorithm A that takes two strings of binary (i.e., $0/1$) symbols as $s1$ and $s2$ and returns the total number of occurrences of $s2$ in $s1$. Let $T(|s1|, |s2|)$ denote the running time of A for inputs $s1$ and $s2$, where $|s|$ denotes the length of string s. Prove that there exists an algorithm that uses A as a subroutine to search for the total number of occurrences of the alphanumeric (i.e., English alphabet + 10 digits) substring $u2$ in another alphanumeric string $u1$ with running time complexity of $O(|u1| + |u2| + T(|u1|, |u2|))$. [10 marks]

Additional space. Do not use unless necessary. Clearly mark corresponding question.

(c) Given an array $A[1, \ldots, n]$ of n distinct positive integers ($n > 1$) in ascending order (i.e., $A[1] < A[2] < \ldots < A[n]$). Please provide a $O(\log n)$ algorithm to find whether there exists index i such that $A[i] = i$, and if yes, returns all the values of i that satisfies this property. [10 marks]

End of question 4

Q4: (a) $\frac{}{10}$	(b) $\frac{}{10}$	(c) $\frac{}{10}$	Total $\frac{}{30}$

Additional space. Do not use unless necessary. Clearly mark corresponding question.

END OF PAPER

UNIVERSITY OF SOUTHAMPTON COMP1201W2

SEMESTER 2 EXAMINATION 2017/2018

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 all parts of the question in section A (30 marks) and TWO *questions from section B (35 marks each).*

The examination is worth 50% of the course.

University approved calculators MAY be used.

A foreign language translation dictionary (paper version) is permitted provided it contains no notes, additions or annotations.

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.

Section A

Question A 1

(a) What is the worst case time complexity of an unbalanced binary search tree and what is the worst case time complexity for a red-black tree? *(2 marks)*

2

(b) Draw the binary search tree produced when inserting 78, 43, 26, 26, 87, 55, 35, 54, 23, 55. *(4 marks)*

4

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

0	2	13	10	24	18	15	25	21	29

(4 marks)

4

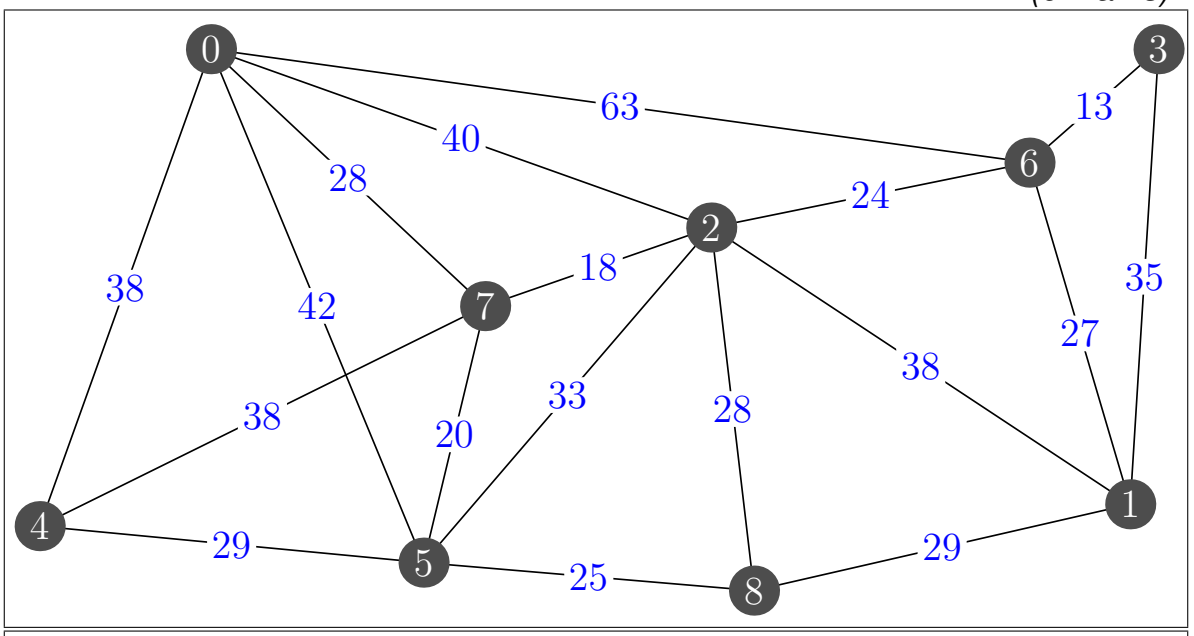
(d) Show how the numbers 23, 29, 84, 15, 58, 19, 81, 17, 48 would be hashed using a hash function $d_2 + 7d_1$ where d_1 is the first (least significant) digit and d_2 the second digit. Show how these would be stored in a hash table using separate chaining. *(8 marks)*

d_2d_1	23	29	84	15	58	19	81	17	48
$d_2 + 7d_1$									
$(d_2 + 7d_1)\%10$									

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

8

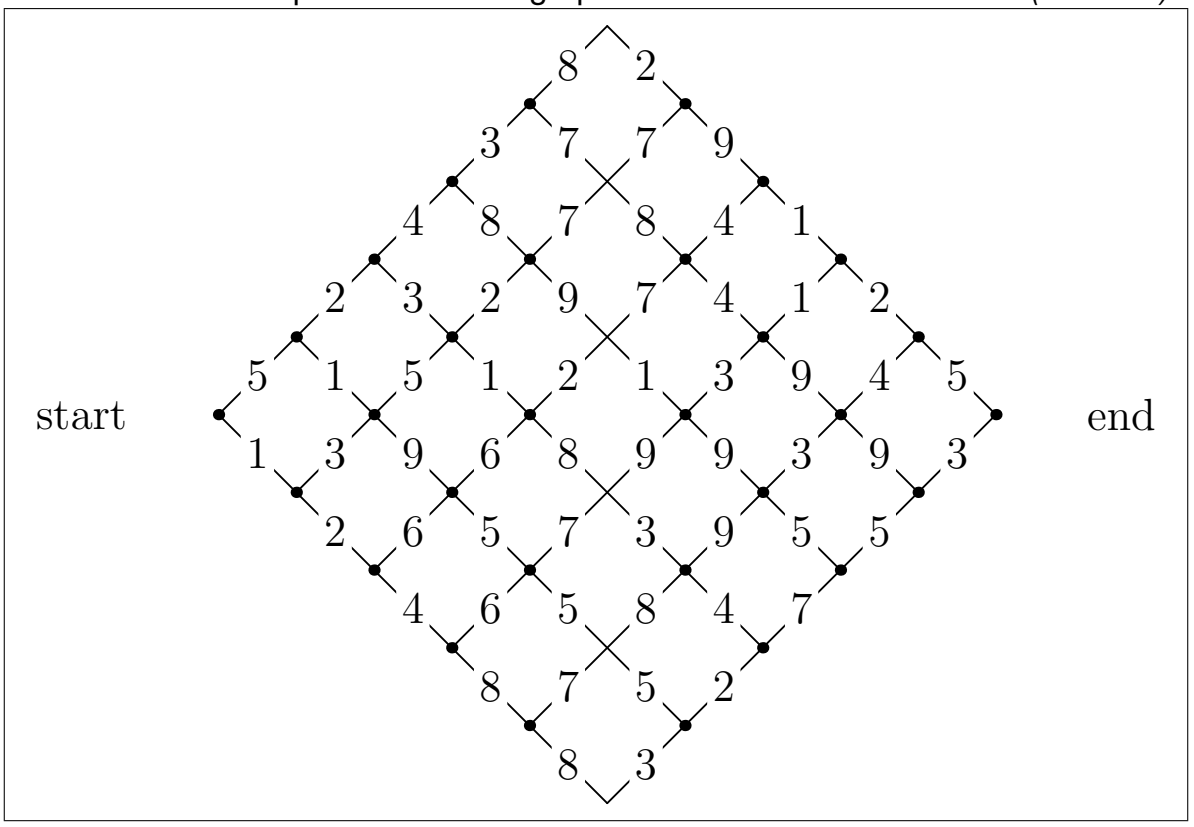
(e) Highlight the edges of the minimum spanning tree found by Prim's algorithm for the graph below and write down the edges (i.e., source node - target node, such as $1-2$) in the order they are found starting from node 0.
(6 marks)



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

6

(f) Use the dynamic programming forward algorithm to compute the minimum cost of each path from the left most node to each other node, where the cost of moving along an edge is equal to the number shown. An edge can only be traversed from left to right. Use the backwards algorithm to find the minimum cost path across the graph. *(6 marks)*



$\frac{}{6}$

End of question 1

Q1: (a) $\frac{}{2}$	(b) $\frac{}{4}$	(c) $\frac{}{4}$	(d) $\frac{}{8}$	(e) $\frac{}{6}$	(f) $\frac{}{6}$	Total $\frac{}{30}$

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 $\{2, 7, 3, 1\}$.

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

Show the state of the arrays when we add 5 to the set. *(5 marks)*

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

5

(b) Show the state of the arrays when you remove 2 from the original set shown in question 4a. *(5 marks)*

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

5

• Do not write in this space •

The Union-Find class is described by the following program

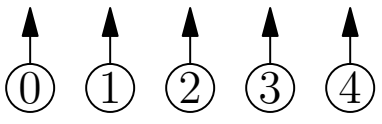
```
public class UnionFind
{
    private int[] s;

    public UnionFind(int numElements) {
        s = new int[numElements];
        for(int i=0; i<s.length; i++)
            s[i] = -1;
    }

    public void union(int root1, int root2) {
        if (s[root2]<s[root1]) {
            s[root1] = root2;
        } else {
            if (s[root1]==s[root2])
                s[root1]--;
            s[root2] = root1;
        }
    }

    public int find(int x) {
        if (s[x]<0)
            return x;
        else
            return s[x] = find(s[x]);
    }
}
```

We assume that we have created an instance of the Union-Find class

```
UnionFind unionFind = new UnionFind(5);
```

Below we show the initial settings of the array `s` and a graphical representation of the forest (set of trees) representing the array.

0	1	2	3	4
-1	-1	-1	-1	-1

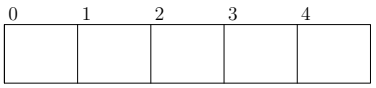


• Do not write in this space •

(c) Show the state of the array and the forest after performing the following operation

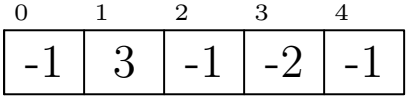
```
unionFind.union(unionFind.find(1), unionFind.find(3));
```

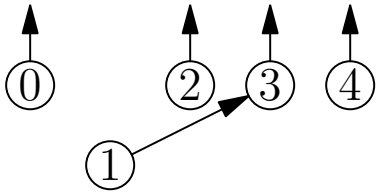
(5 marks)



5

(d) Given an array shown below

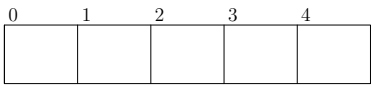




Show the state of the array and the forest after performing the following operation

```
unionFind.union(unionFind.find(1), unionFind.find(2));
```

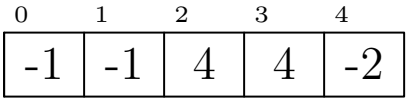
(5 marks)

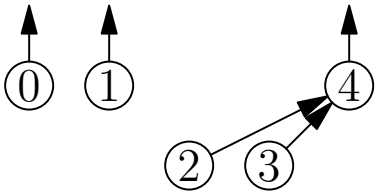


5

• Do not write in this space •

(e) Given an array shown below

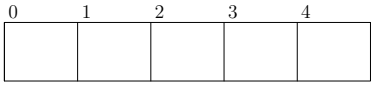




Show the state of the array and the forest after performing the following operation

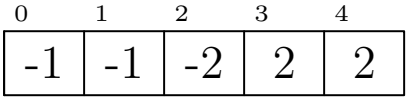
```
unionFind.union(unionFind.find(0), unionFind.find(1));
```

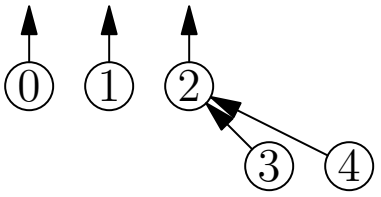
(5 marks)



5

(f) Given an array shown below

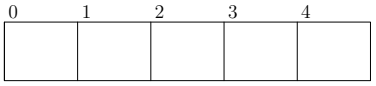




Show the state of the array and the forest after performing the following operation

```
unionFind.union(unionFind.find(1), unionFind.find(2));
```

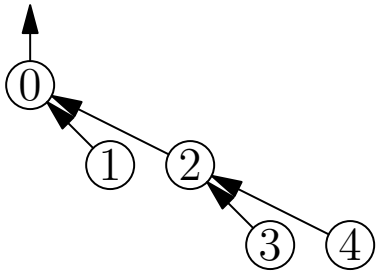
(5 marks)



5

(g) Given an array shown below

0	1	2	3	4
-3	0	0	2	2



Show the state of the array and the forest after performing the following operation

```
unionFind.find(2);
```

(5 marks)

0	1	2	3	4

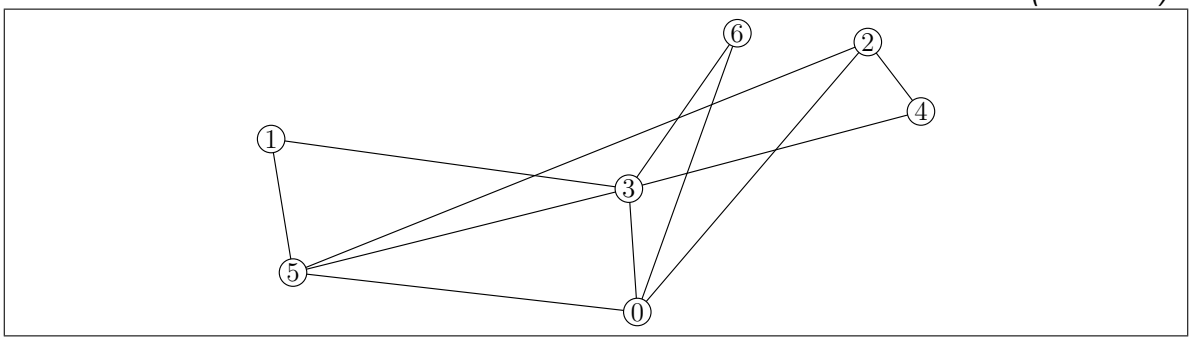
5

End of question 2

Q2: (a) $\frac{}{5}$ (b) $\frac{}{5}$ (c) $\frac{}{5}$ (d) $\frac{}{5}$ (e) $\frac{}{5}$ (f) $\frac{}{5}$ (g) $\frac{}{5}$ Total $\frac{}{35}$

Question B 3

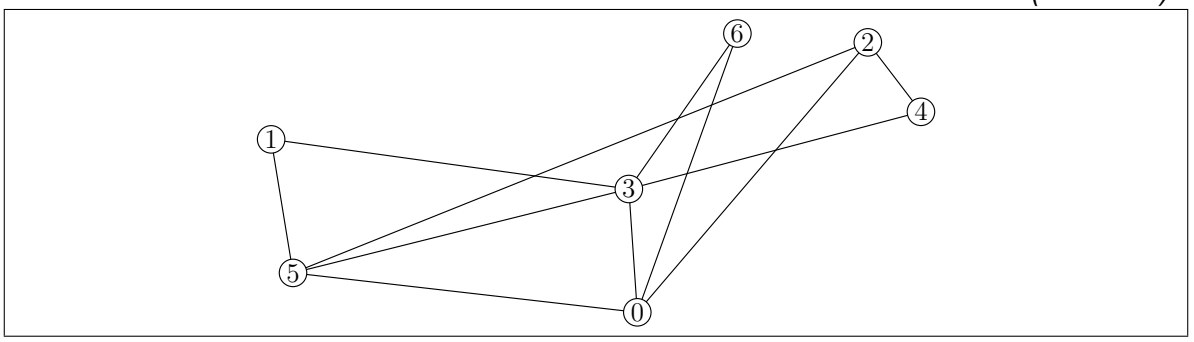
(a) Draw the edges on the graph used to find the vertices using **breadth first search** starting from vertex 0 where the lower numbered vertices are searched first. Write the order in which the vertices are discovered. *(8 marks)*



bfs order = _____

8

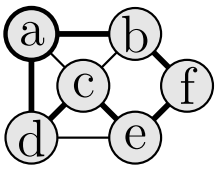
(b) Draw the edges on the graph used to find the vertices using **depth first search** starting from vertex 0 where the lower numbered vertices are searched first. Write the order in which the vertices are discovered. *(8 marks)*



dfs order = _____

8

(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

• Do not write in this space •

(d) Describe the branch and bound strategy. *(5 marks)*

5

(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. *(6 marks)*

1

2

3

4

6

End of question 3

Q3: (a) $\frac{}{8}$ (b) $\frac{}{8}$ (c) $\frac{}{8}$ (d) $\frac{}{5}$ (e) $\frac{}{6}$ Total $\frac{}{35}$

Question B 4

(a) Given an array $A[0, \ldots, n]$ of $n + 1$ integers with $n > 2$. Let $A[i]$ denote the i^{th} element of the array (with $i = 0, 1, \ldots, n$). We also know that for any $0 \leq i \leq n - 2$, we have $A[i] \leq A[i + 3]$. Propose a sorting algorithm that can sort array A in an ascending order with $O(n)$ comparisons (you do not need to provide the pseudo code, a high level description suffices). *(10 marks)*

10

• Do not write in this space •

(b) Let $x_1, x_2, \ldots, x_n$ be integers (which can either be positive or negative, as well as 0). We would like to decide whether we can delete at most 5 numbers from this list such that the sum of the remaining numbers is at most 2018. Propose a solution that only requires $O(n)$ comparisons. *(10 marks)*

$\overline{10}$

• Do not write in this space •

(c) Given n distinct positive integers $x_1, x_2, \ldots, x_n$. We would like to find the largest subset of these numbers such that the elements of this subset form an arithmetic sequence with difference of 2018 (i.e., the subset forms a sequence of $\{x, x+2018, x+4036, x+6054, \ldots\}$). Propose a solution that only requires $O(n \log n)$ operations. *(15 marks)*

$\overline{15}$

End of question 4

Q4: (a) $\overline{10}$ (b) $\overline{10}$ (c) $\overline{15}$ Total $\overline{35}$

END OF PAPER

UNIVERSITY OF SOUTHAMPTON COMP1201W1

SEMESTER 2 EXAMINATION 2016/2017

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 all parts of the question in section A (30 marks) and TWO *questions from section B (35 marks each).*

University approved calculators MAY be used.

A foreign language dictionary is permitted ONLY IF it is a paper version of a direct 'Word to Word' translation dictionary AND it contains no notes, additions or annotations.

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.

Section A

Question A 1

(a) Describe the two rules that define AVL trees *(4 marks)*

1

2

4

(b) Let $m(h)$ be the minimum number of nodes in an AVL tree of height h. Write down a recurrence relationship for $m(h)$ *(4 marks)*

4

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

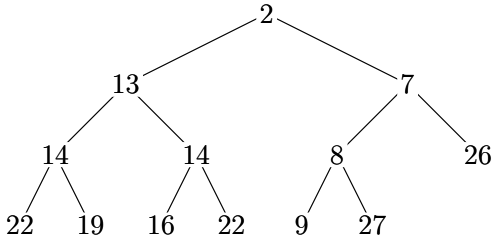
1	10	9	11	20	11	15	24	23	29

(2 marks)

2

(d) Show what happens to the heap shown on the left when you remove the minimum entry. *(3 marks)*

Before



After `removeMin()`

3

(e) Write pseudo code for the Merge sort algorithm. What is its average and worst case time complexity? *(4 marks)*

Average case complexity:

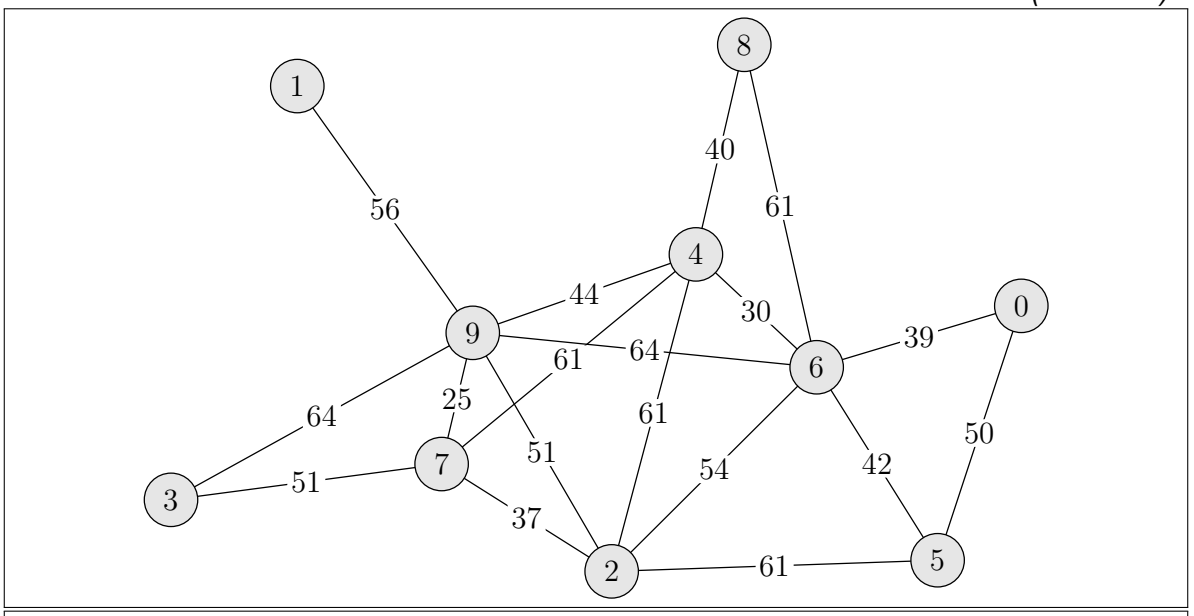
Worst case complexity:

4

(f) Explain why the Java collections class uses Merge sort, while for arrays of primitive types Java uses Quick sort. *(2 marks)*

2

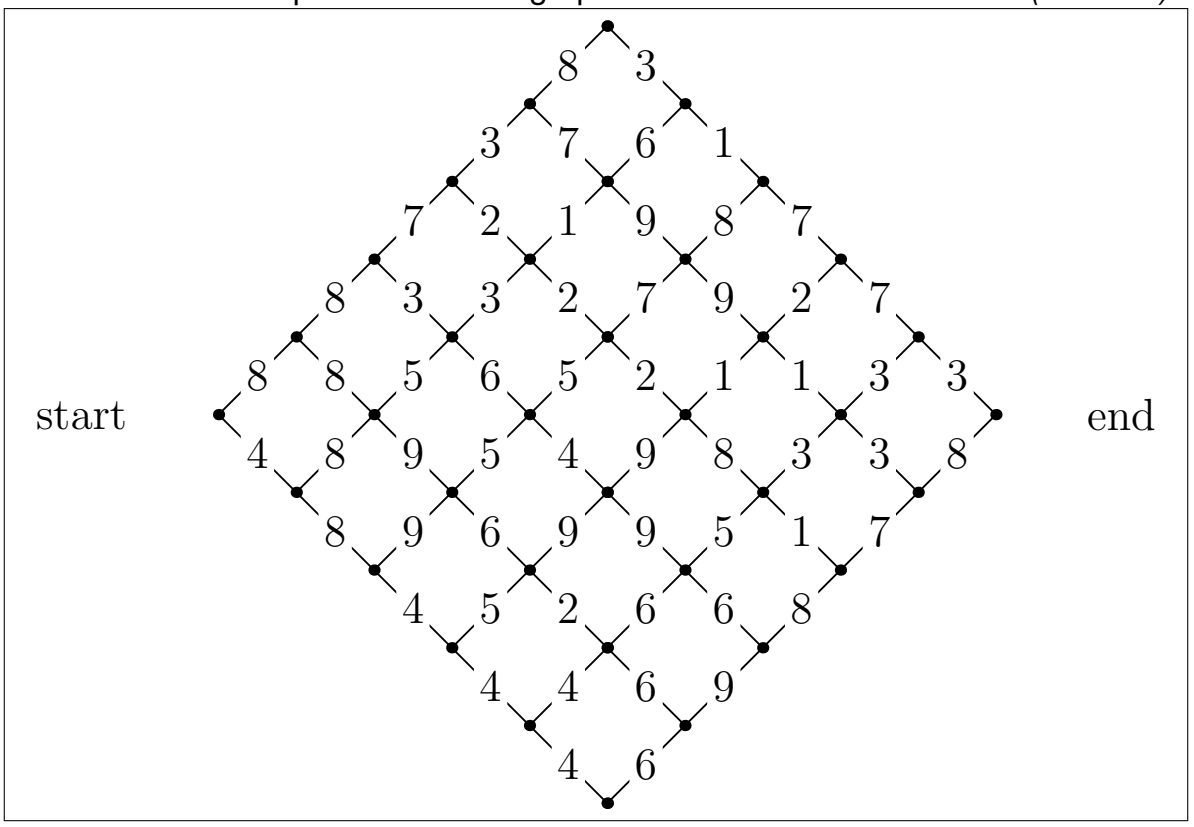
(g) Show the tree of edges found by Dijkstra's algorithm from node 0 and write down the order of the edges and the distance of the node to the source node. *(6 marks)*



1.	2.	3.
4.	5.	6.
7.	8.	9.

6

(h) Use the dynamic programming forward algorithm to compute the minimum cost of each path from the left most node to each other node, where the cost of moving along an edge is equal to the number shown. An edge can only be traversed from left to right. Use the backwards algorithm to find the minimum cost path across the graph. *(5 marks)*



5

End of question 1

Q1: (a) $\frac{}{4}$	(b) $\frac{}{4}$	(c) $\frac{}{2}$	(d) $\frac{}{3}$	(e) $\frac{}{4}$	(f) $\frac{}{2}$	(g) $\frac{}{6}$	(h) $\frac{}{5}$	Total $\frac{}{30}$

Section B

Question B 2

(a) What is the main application of B-Trees? *(2 marks)*

2

(b) Describe what problem B-Trees solve and how they solve it. *(5 marks)*

5

(c) Sketch a B-Tree of order 5 (i.e., $M = 5$). *(6 marks)*

6

(d) What is a trie or digital tree *(3 marks)*

3

(e) What is (1) the advantage and (2) disadvantage of using a trie compared with a binary search tree? *(4 marks)*

1

2

4

(f) Show how the following words would be inserted into a trie based on tables *(10 marks)*

Add words: {BE, WITH, YOU, MAY, THE, FOURTH, SAID, MASTER, YODA}

	0	1	2	3	4
$					
A					
B					
D					
E					
F					
H					
I					
M					
O					
R					
S					
T					
U					
W					
Y					

10

(g) Write down all the suffixes of the word "**keeper**" and draw the suffix tree for the word. *(5 marks)*

5

Note: This picture only depicts the first level of the solution, the other level can be depicted similarly.

End of question 2

Q2: (a) $\frac{}{2}$	(b) $\frac{}{5}$	(c) $\frac{}{6}$	(d) $\frac{}{3}$	(e) $\frac{}{4}$	(f) $\frac{}{10}$	(g) $\frac{}{5}$	Total $\frac{}{35}$

Question B 3

(a) Describe the difference between depth first search and breadth first search. Give an application of each. *(6 marks)*

DFS Application:

BFS Application:

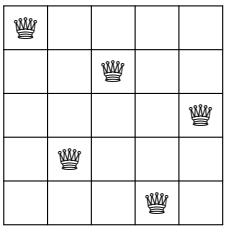
6

(b) What is topological sort? Give an application. *(6 marks)*

Application:

6

(c) The n-queens problem is to put n queens on a chess board such that no queen is in the same row, column or diagonal as any other queen. Either write pseudo code or describe in outline an algorithm to solve the n-queens problem.



(10 marks)

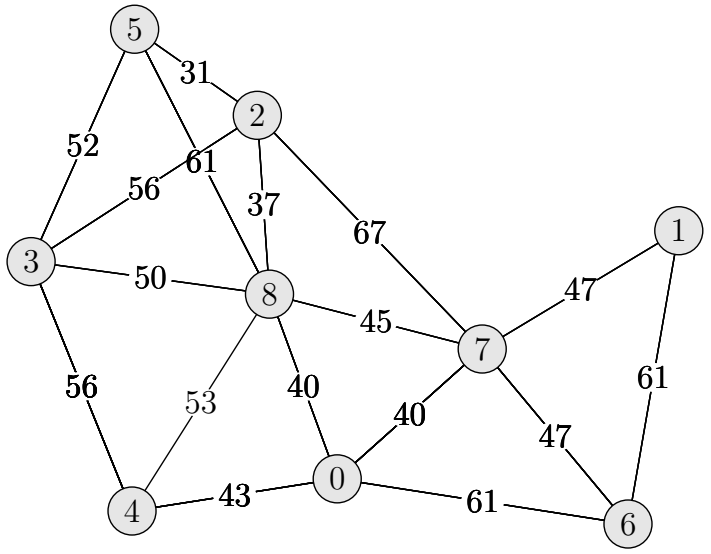
10

(d) Describe Kruskal's algorithm for constructing a minimum spanning tree. What data structures are needed to implement the algorithm? *(5 marks)*

Data Structures:

5

(e) In the graph below draw the minimum spanning tree (by highlighting the edges) and write down the order in which the edges would be found by Kruskal's algorithm. *(8 marks)*



8

1. **2.**

3. **4.**

5. **6.**

7. **8.**

End of question 3

Q3:	(a) $\frac{}{6}$	(b) $\frac{}{6}$	(c) $\frac{}{10}$	(d) $\frac{}{5}$	(e) $\frac{}{8}$	Total $\frac{}{35}$

Question B 4

(a) Describe the local (neighbourhood) search strategy. What is its main drawback? *(6 marks)*

Drawback:

6

(b) Describe a strategy to overcome this. *(3 marks)*

3

(c) How efficient would you expect heuristic search to be on a problem like TSP or graph colouring? *(4 marks)*

4

(d) Describe the three conditions required of a linear programing problem. *(6 marks)*

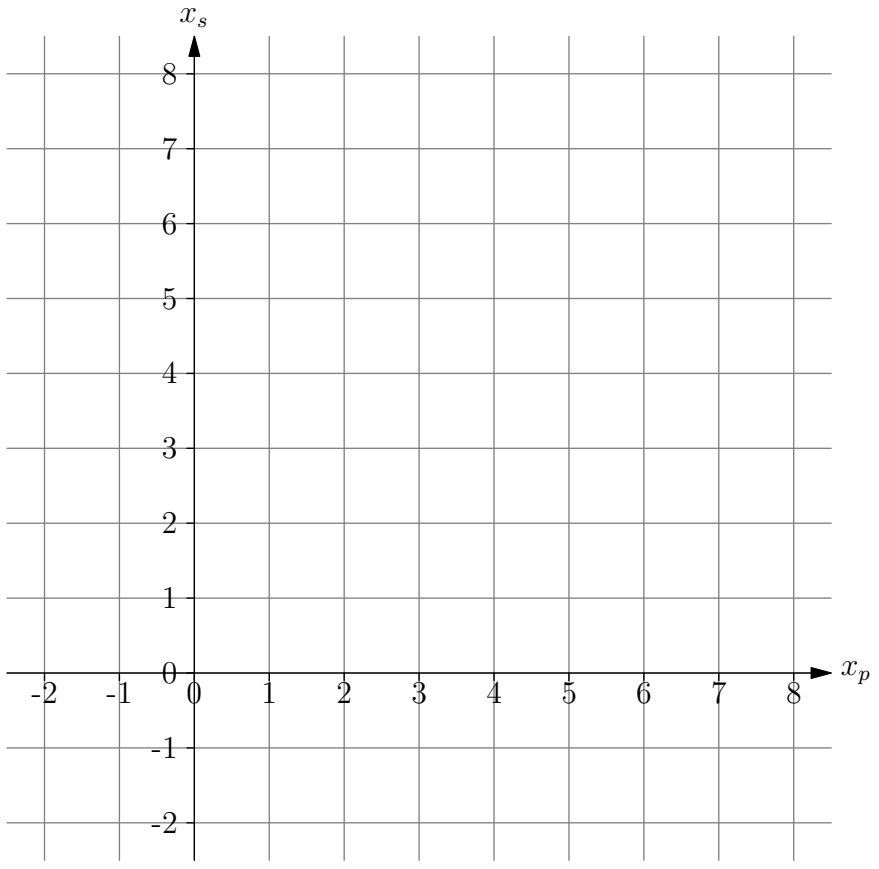
1

2

3

6

(e) Gulliver, in preparation for his next travel, would like to fill up his food supply. To do so, he can either buy from the Lilliputians or the Brobdingnags, but with twice as much price from the latter. Based on his calculations, Gulliver will need at least 12 kgs of potatoes. However, for each food package bought in Lilliput, there are 2 kgs of potatoes, and 3 kgs in each package from Brobdingnag. In addition, Gulliver also wants to have at least 8 kgs of beans, and there are 4 kgs in a Lilliputian food package but only 1 kg beans in a Brobdingnag package. Write down the linear programming problem that aims to minimise Gulliver's total cost spent on food and draw the direction that minimises the cost and the constraints (shade the infeasible region). Mark the optimal solution. *(10 marks)*



10

(f) Describe in brief outline the simplex algorithm. *(6 marks)*

$\frac{}{6}$

End of question 4

Q4: (a) $\frac{}{6}$	(b) $\frac{}{3}$	(c) $\frac{}{4}$	(d) $\frac{}{6}$	(e) $\frac{}{10}$	(f) $\frac{}{6}$	Total $\frac{}{35}$

END OF PAPER

UNIVERSITY OF SOUTHAMPTON COMP1201W1

SEMESTER 2 EXAMINATION 2015/2016

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 all parts of the question in section A (30 marks) and TWO *questions from section B (35 marks each).*

The examination is worth 85% of the course. The tutorials are worth 15%.

University approved calculators MAY be used.

A foreign language dictionary is permitted ONLY IF it is a paper version of a direct 'Word to Word' translation dictionary AND it contains no notes, additions or annotations.
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.

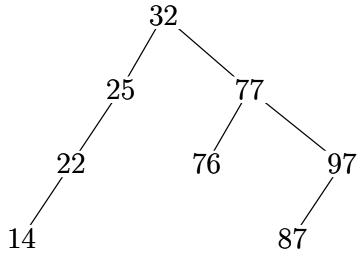
Section A

Question A 1

(a) Draw the binary search tree produced when inserting 26, 93, 81, 85, 58, 92, 9, 84, 95. *(2 marks)*

2

(b) Draw the tree obtained by deleting 77 from the binary search tree shown. *(2 marks)*



`delete(77)`

2

(c) What is the worst case time complexity of an unbalanced binary search tree and what is the worst case time complexity for a red-black tree? *(2 marks)*

2

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

0	1	6	13	13	22	18	20	29	25

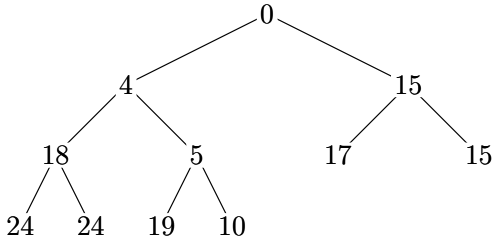
(2 marks)

2

(e) Show what happens to the heap shown on the left when you add 5.

(2 marks)

Before



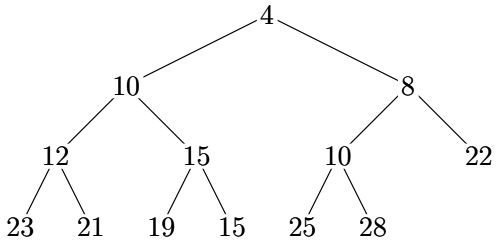
After `add(5)`

2

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

(3 marks)

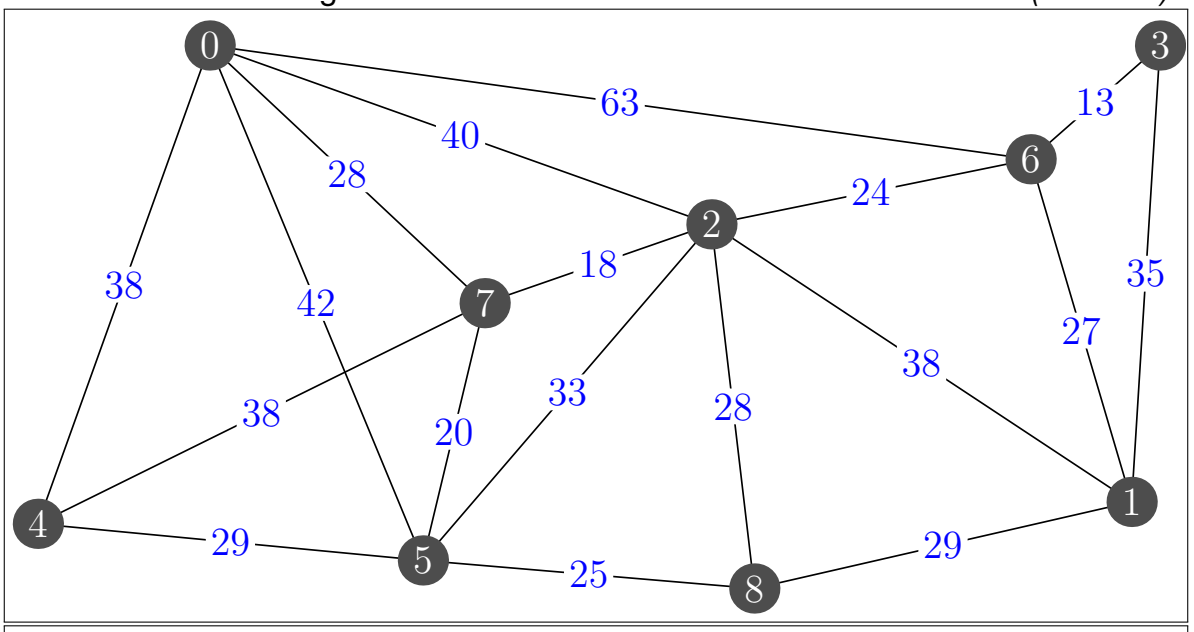
Before



After `removeMin()`

3

(g) Highlight the edges of the minimum spanning tree found by Prim's algorithm for the graph below and write down the edges (i.e. $v_1 - v_2$) in the order they are found starting from node 0. *(5 marks)*



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

5

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

Letter	a	b	c	d	e	f	g
Frequency	5	9	2	12	19	1	7

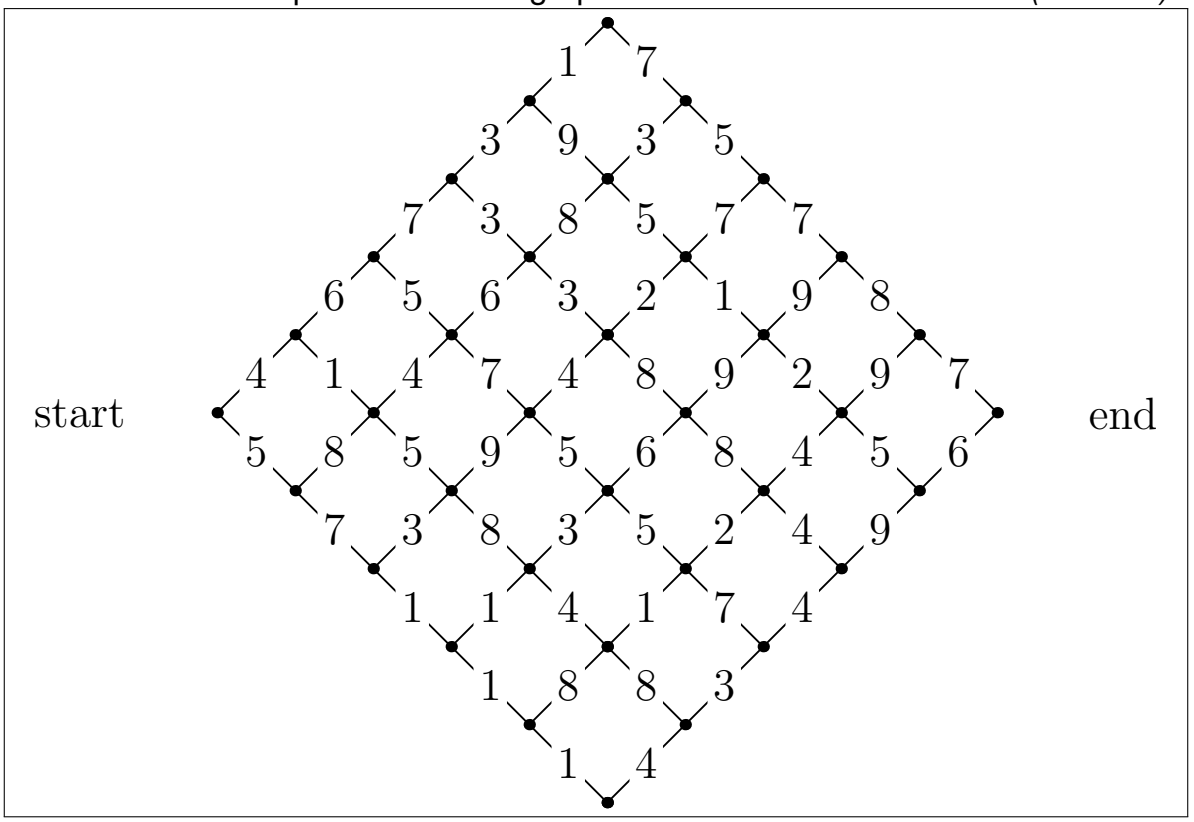
(5 marks)

5

(i) How would the word “gaffe” be coded in your Huffman tree (show the letter break with a hyphen). *(2 marks)*

2

(j) Use the dynamic programming forward algorithm to compute the minimum cost of each path from the left most node to each other node, where the cost of moving along an edge is equal to the number shown. An edge can only be traversed from left to right. Use the backwards algorithm to find the minimum cost path across the graph. *(5 marks)*



5

End of question 1

Q1: (a) $\frac{}{2}$ (b) $\frac{}{2}$ (c) $\frac{}{2}$ (d) $\frac{}{2}$ (e) $\frac{}{2}$ (f) $\frac{}{3}$ (g) $\frac{}{5}$ (h) $\frac{}{5}$ (i) $\frac{}{2}$ (j) $\frac{}{5}$ Total $\frac{}{30}$

Section B

Question B 2 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 in the worst case $n-1$.

(a) Let $T(n)$ be the worst case number of comparison operations used by Merge-Sort to sort an array of size n. Write down a recurrence relation for $T(n)$ valid if $n = 2^m$ *(5 marks)*

$T(n) =$

5

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

$T(1) =$

$T(2) =$

$T(4) =$

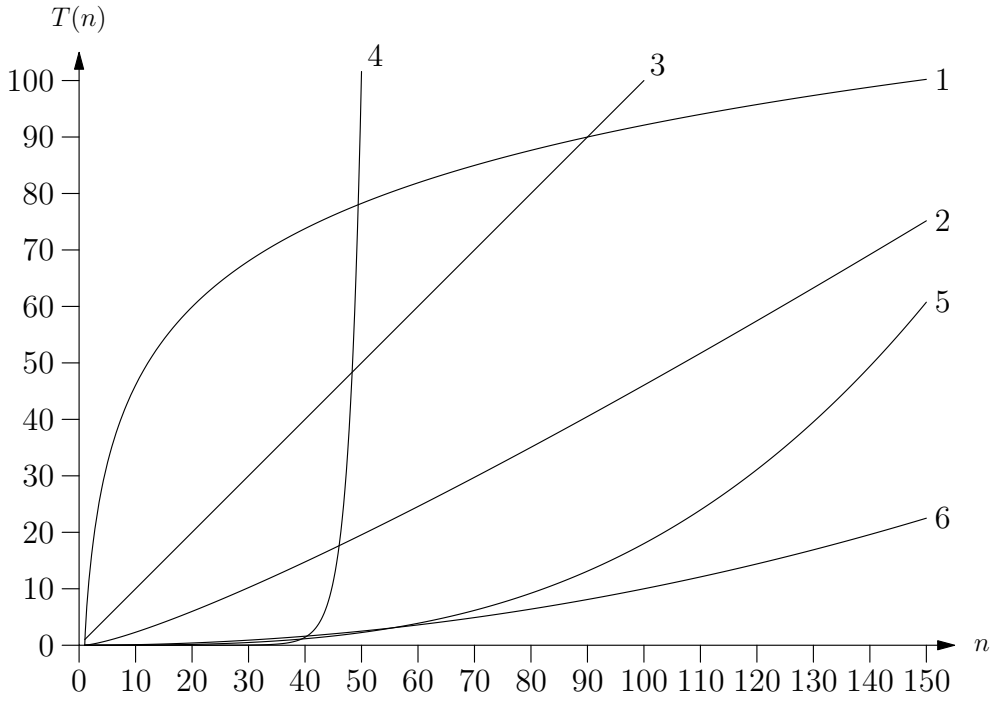
$T(8) =$

4

(c) Demonstrate, for $n = 2^m$, that $f(n) = n\log_2(n) - n + 1$ satisfies the recurrence relation in part (a). *(9 marks)*

$\overline{9}$

(d) The graph below shows the time complexity for the following algorithms (a) $\Theta((n/a)!)$, (b) $\Theta(n^2)$, (c) $\Theta(n\ \log(n))$, (d) $\Theta(n)$, (e) $\Theta(n^3)$, and (f) $\Theta(\log(n))$. Match the time complexity classes with the curves on the graph.



(6 marks)

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

6

• Do not write in this space •

(e) Which of the following statements are true? Give reasons why (marks will only be awarded if correct reasons are given). *(6 marks)*

(i) All $\Theta(n^2)$ algorithms are faster than all $\Theta(n^3)$ algorithms

(ii) An $O(n)$ algorithm will run faster than a $\Omega(n\ \log(n))$ algorithm for sufficiently large n

(iii) All $O(n^3)$ algorithms run slower than all $O(n^2)$ asymptotically

$\overline{6}$

(f) Why is it widely believed that $NP \neq P$? *(5 marks)*

5

End of question 2

Q2: (a) $\frac{}{5}$	(b) $\frac{}{4}$	(c) $\frac{}{9}$	(d) $\frac{}{6}$	(e) $\frac{}{6}$	(f) $\frac{}{5}$	Total $\frac{}{35}$

Question B 3

(a) What is a trie? *(3 marks)*

3

(b) What is its disadvantage? *(2 marks)*

2

(c) Show how the following words would be inserted into a trie based on tables *(10 marks)*

Add words: {ADD, THESE, WORDS, INTO, A, TRIE, IN, THEIR, RIGHT, PLACES}

	0	1	2	3	4
$					
A					
B					
C					
D					
E					
H					
I					
O					
P					
R					
S					
T					
W					

10

(d) Show how the numbers 23, 29, 84, 15, 58, 19, 81, 17, 48 would be hashed using a hash function $d_2 + 3d_1$ where d_1 is the first (least significant) digit and d_2 the second digit. Show how these would be stored in a hash table of size 10 using separate chaining. *(10 marks)*

d_2d_1	23	29	84	15	58	19	81	17	48
$d_2 + 3d_1$									
$(d_2 + 3d_1)\%10$									

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

10

(e) Show how the numbers 23, 29, 84, 15, 58, 19, 81, 17, 48 would be stored in a hash table using linear probing assuming the same hash codes. *(6 marks)*

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

6

(f) What is the disadvantage of linear probing and how can open addressing be modified to overcome this disadvantage? *(4 marks)*

1

2

$\frac{}{4}$

End of question 3

Q3: (a) $\frac{}{3}$ (b) $\frac{}{2}$ (c) $\frac{}{10}$ (d) $\frac{}{10}$ (e) $\frac{}{6}$ (f) $\frac{}{4}$ Total $\frac{}{35}$

Question B 4

(a) We can implement a fast set for a fixed number of integers using two arrays. Below we show the representation of the set $\{2, 7, 3, 1\}$.

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

Show the state of the arrays when we add 8 to the set. *(5 marks)*

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

5

(b) Show the state of the arrays when you remove 3 from the original set shown in question 4a. *(5 marks)*

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

5

• Do not write in this space •

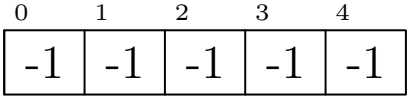
The disjoint set class is described by the following program

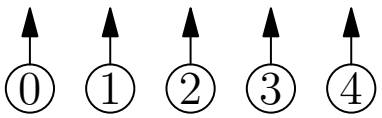
```
public class DisjSets
{
    private int[] s;

    public DisjSets(int numElements) {
        s = new int[numElements];
        for(int i=0; i<s.length; i++)
            s[i] = -1;
    }

    public void union(int root1, int root2) {
        if (s[root2]<s[root1]) {
            s[root1] = root2;
        } else {
            if (s[root1]==s[root2])
                s[root1]--;
            s[root2] = root1;
        }
    }

    public int find(int x) {
        if (s[x]<0)
            return x;
        else
            return s[x] = find(s[x]);
    }
}
```

We assume that we have created an instance of the disjoint sets class

```
DisjSets disjset = new DisjSets(5);
```

Below we show the initial settings of the array `s` and a graphical representation of the forest (set of trees) representing the array.



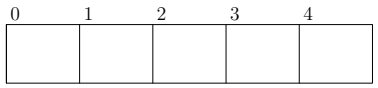


• Do not write in this space •

(c) Show the state of array and the forest after performing the following operation

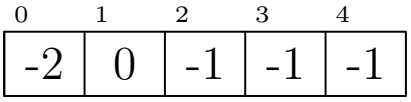
```
disjset.union(disjset.find(2), disjset.find(3));
```

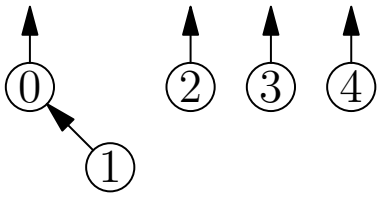
(5 marks)



5

(d) Given an array shown below

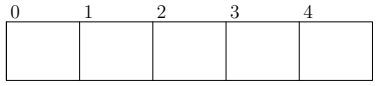




Show the state of array and the forest after performing the following operation

```
disjset.union(disjset.find(1), disjset.find(2));
```

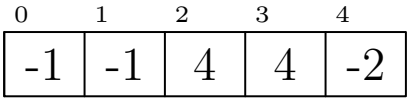
(5 marks)

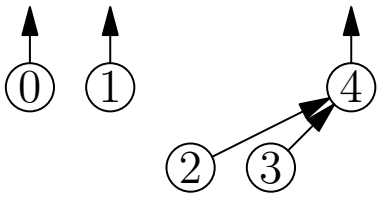


5

• Do not write in this space •

(e) Given an array shown below

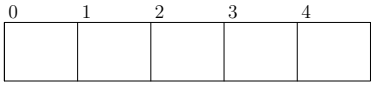




Show the state of array and the forest after performing the following operation

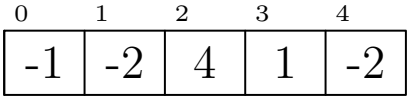
```
disjset.union(disjset.find(0), disjset.find(1));
```

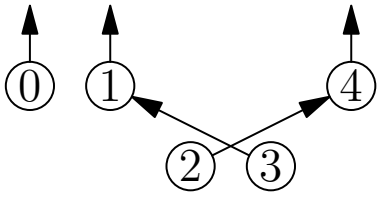
(5 marks)



5

(f) Given an array shown below

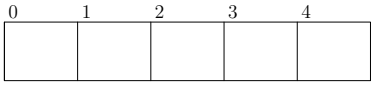




Show the state of array and the forest after performing the following operation

```
disjset.union(disjset.find(1), disjset.find(2));
```

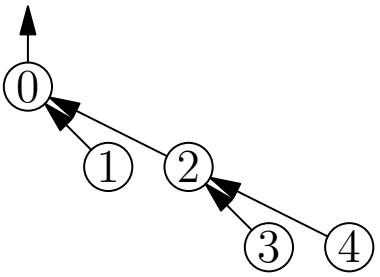
(5 marks)



5

(g) Given an array shown below

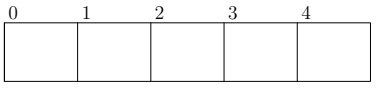
0	1	2	3	4
-3	0	0	2	2



Show the state of array and the forest after performing the following operation

```
disjset.find(3);
```

(5 marks)



5

End of question 4

Q4: (a) $\frac{}{5}$ (b) $\frac{}{5}$ (c) $\frac{}{5}$ (d) $\frac{}{5}$ (e) $\frac{}{5}$ (f) $\frac{}{5}$ (g) $\frac{}{5}$ Total $\frac{}{35}$

END OF PAPER

UNIVERSITY OF SOUTHAMPTON COMP1201W1

SEMESTER 2 EXAMINATION 2014/2015

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 all parts of the question in section A (25 marks) and TWO *questions from section B (30 marks each).*

This examination is worth 85%. The tutorials were worth 15%.

University approved calculators MAY be used.

A foreign language word to word® translation dictionary (paper version) is permitted provided it contains no notes, additions or annotations.

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

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

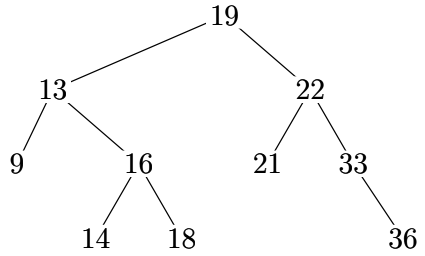
Section A

Question A 1

(a) Draw the binary search tree produced when inserting 57, 90, 13, 9, 50, 43, 55, 82, 56, 51. *(2 marks)*

2

(b) Draw the tree obtained by deleting 22 from the binary search tree shown. *(2 marks)*



`delete(22)`

2

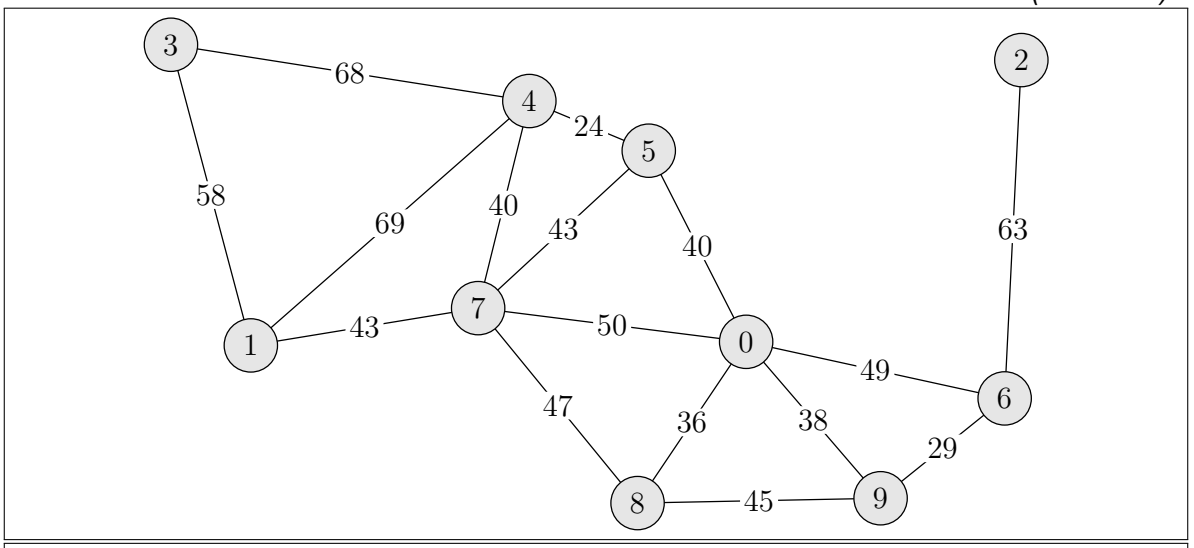
(c) What type of binary search tree is used in the java TreeSet? *(1 marks)*

1

(d) When would you prefer to use a set rather than a list and the vice versa? *(3 marks)*

3

(e) Show the tree of edges found by Dijkstra's algorithm from node 0 and write down the order of the edges and the distance of the node to the source node. *(7 marks)*



7

1.	2.	3.
4.	5.	6.
7.	8.	9.

TURNOVER

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

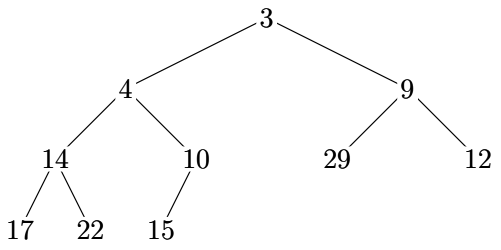
4	19	9	19	20	22	15	24	27	22

(1 marks)

1

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

Before

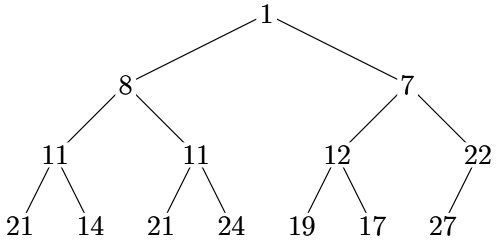


After `add(2)`

2

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

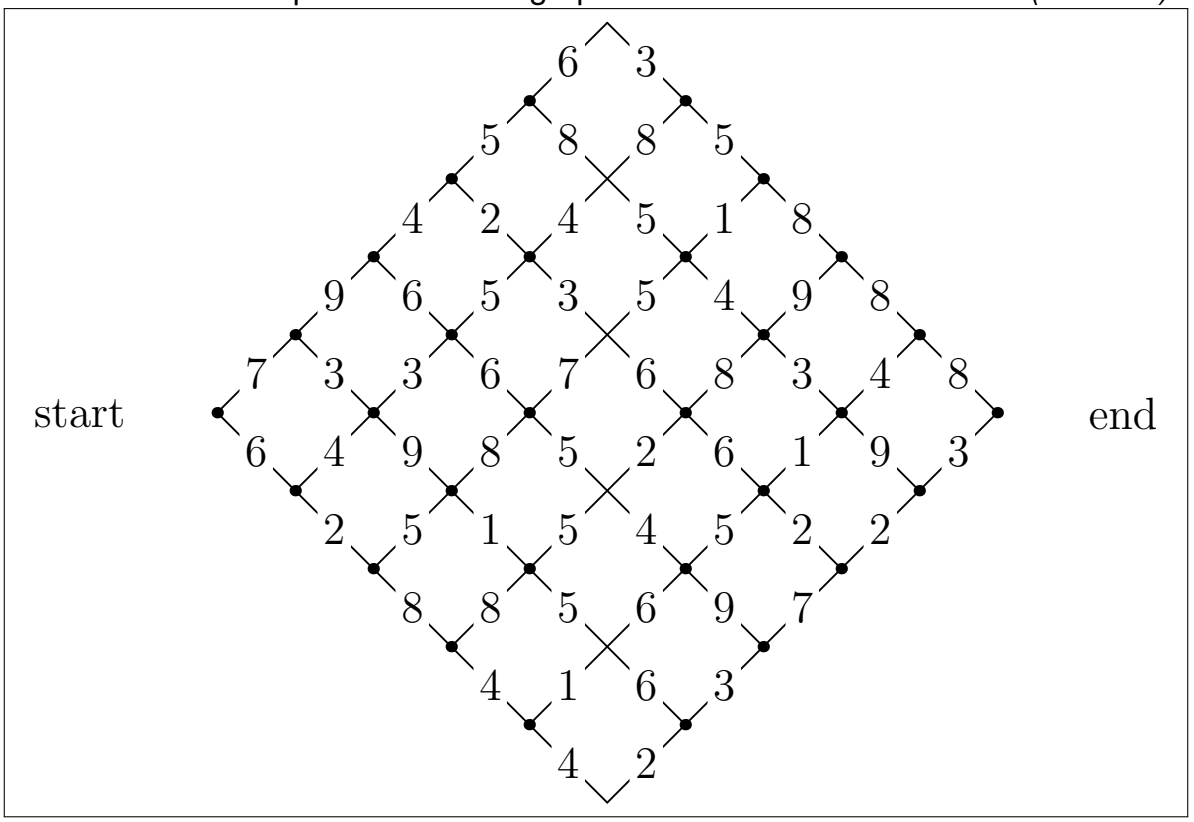
Before



After `removeMin()`

2

(i) Use the dynamic programming forward algorithm to compute the minimum cost of each path from the left most node to each other node where the cost of moving along an edge is equal to the number shown. An edge can only be traversed from left to right. Use the backwards algorithm to find the minimum cost path across the graph. *(5 marks)*



5

End of question 1

Q1: (a) $\frac{}{2}$ (b) $\frac{}{2}$ (c) $\frac{}{1}$ (d) $\frac{}{3}$ (e) $\frac{}{7}$ (f) $\frac{}{1}$ (g) $\frac{}{2}$ (h) $\frac{}{2}$ (i) $\frac{}{5}$ Total $\frac{}{25}$

TURNOVER

Section B

Question B 2

(a) What is the main application of B-Trees? *(1 marks)*

1

(b) Describe what problem B-Trees solve and how they solve it. *(3 marks)*

3

(c) Sketch a B-Tree. *(6 marks)*

6

(d) What is a trie or digital tree *(3 marks)*

3

(e) What is (1) the advantage and (2) disadvantage of using a trie compared with a binary search tree? *(2 marks)*

1

2

2

(f) Show how the following words would be inserted into a trie based on tables *(10 marks)*

Add words: {BIN, BINARY, MAP, MAPPER, MAPPERS, MAPS, OTHER, SET, SETTER, THAT, THE, THIS, TREE}

	0	1	2	3	4	5	6
$							
A							
B							
C							
D							
E							
I							
L							
M							
N							
O							
P							
R							
S							
T							
U							

10

(g) Write down all the suffixes of the word "**queues**" and draw the suffix tree for the word. *(5 marks)*

5

End of question 2

Q2: (a) $\frac{}{1}$	(b) $\frac{}{3}$	(c) $\frac{}{6}$	(d) $\frac{}{3}$	(e) $\frac{}{2}$	(f) $\frac{}{10}$	(g) $\frac{}{5}$	Total $\frac{}{30}$

Question B 3

The quicksort algorithm is as follows

```
QuickSort(a, left, right) {
  if (right-left < threshold)
    InsertionSort(a, left, right)
  else
    pivot = ChoosePivot(a, left, right)
    part = Partition(a, pivot, left, right)
    QuickSort(a, left, part)
    QuickSort(a, part+1, right)
  endif
}
```

(a) Describe the ChoosePivot algorithm for finding a pivot. *(3 marks)*

3

(b) How does the Partition algorithm work? *(5 marks)*

5

(c) Explain why quicksort uses INSERTIONSORT? *(3 marks)*

3

(d) Assume that PARTITION takes n operations and that the pivot splits the array exactly in half at each step (assume also that the `threshold` equals 1). Write a recursion relations for the number of partitioning operations, $T(n)$. *(4 marks)*

T(n) =

4

(e) Show that $T(n) = n \log_2(n)$ satisfies the recursion relation in part (d). *(4 marks)*

4

(f) Assume that PARTITION takes n operations and that the pivot splits the array into an array of size $n - 1$ and another array of size 1. Write a recursion relations for the number of partitioning operations, $T(n)$. *(4 marks)*

T(n) =

4

(g) Show that if $T(1) = 0$ then the time complexity, $T(n)$, of quicksort (given the unlucky partitioning described in part (f)) is equal to the function $f(n) = n(n+1)/2 - 1$. *(4 marks)*

$\overline{4}$

(h) Give the worst case complexity for quicksort and explain why this does not put off people using it. *(3 marks)*

$\overline{3}$

End of question 3

Q3: (a) $\overline{3}$ (b) $\overline{5}$ (c) $\overline{3}$ (d) $\overline{4}$ (e) $\overline{4}$ (f) $\overline{4}$ (g) $\overline{4}$ (h) $\overline{3}$ Total $\overline{30}$

• Do not write in this space •

TURNOVER

Question B 4

(a) Which of the following algorithms are in class P and which are in class NP-complete *(5 marks)*

(i) TSP:

(ii) Minimum Spanning Tree:

(iii) Maximum Flow:

(iv) Graph Colouring:

(v) Linear Assignment:

5

(b) Describe what it means to be class NP-complete. *(5 marks)*

5

(c) Describe neighbourhood search and explain how it could be used to find a good solution to an optimisation problem. *(5 marks)*

5

(d) Briefly describe simulated annealing and why it is used. *(5 marks)*

5

(e) Briefly describe branch and bound and its expected performance. *(5 marks)*

15

(f) Briefly describe how dynamic programming can be used to solve TSP and describe its time complexity *(5 marks)*

15

End of question 4

Q4: (a) $\frac{}{5}$	(b) $\frac{}{5}$	(c) $\frac{}{5}$	(d) $\frac{}{5}$	(e) $\frac{}{5}$	(f) $\frac{}{5}$	Total $\frac{}{30}$

END OF PAPER

UNIVERSITY OF SOUTHAMPTON COMP1201W1

SEMESTER 2 EXAMINATION 2013/2014

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 all parts of the question in section A (25 marks) and TWO *questions from section B (30 marks each).*

This examination is worth 85%. The tutorials were worth 15%.

University approved calculators MAY be used.

A foreign language translation dictionary (paper version) is permitted provided it contains no notes, additions or annotations.

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.

Section A

Question A 1

(a) Write pseudo code for the quick sort algorithm. What is its average and worst case time complexity? *(5 marks)*

Average case complexity:

Worst case complexity:

5

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

3	5	5	13	20	12	29	25	29	29

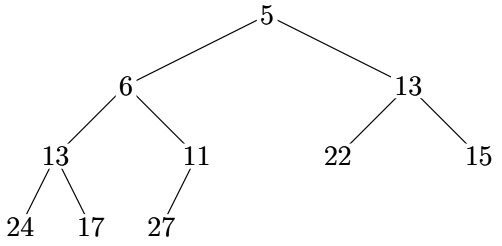
(1 marks)

1

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

(2 marks)

Before

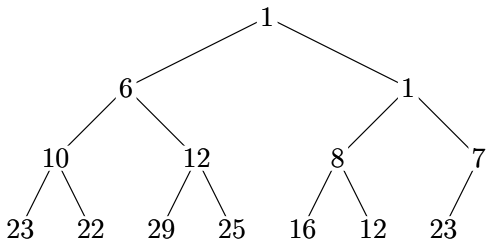


After `add(2)`

2

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

Before



After `removeMin()`

2

TURN OVER

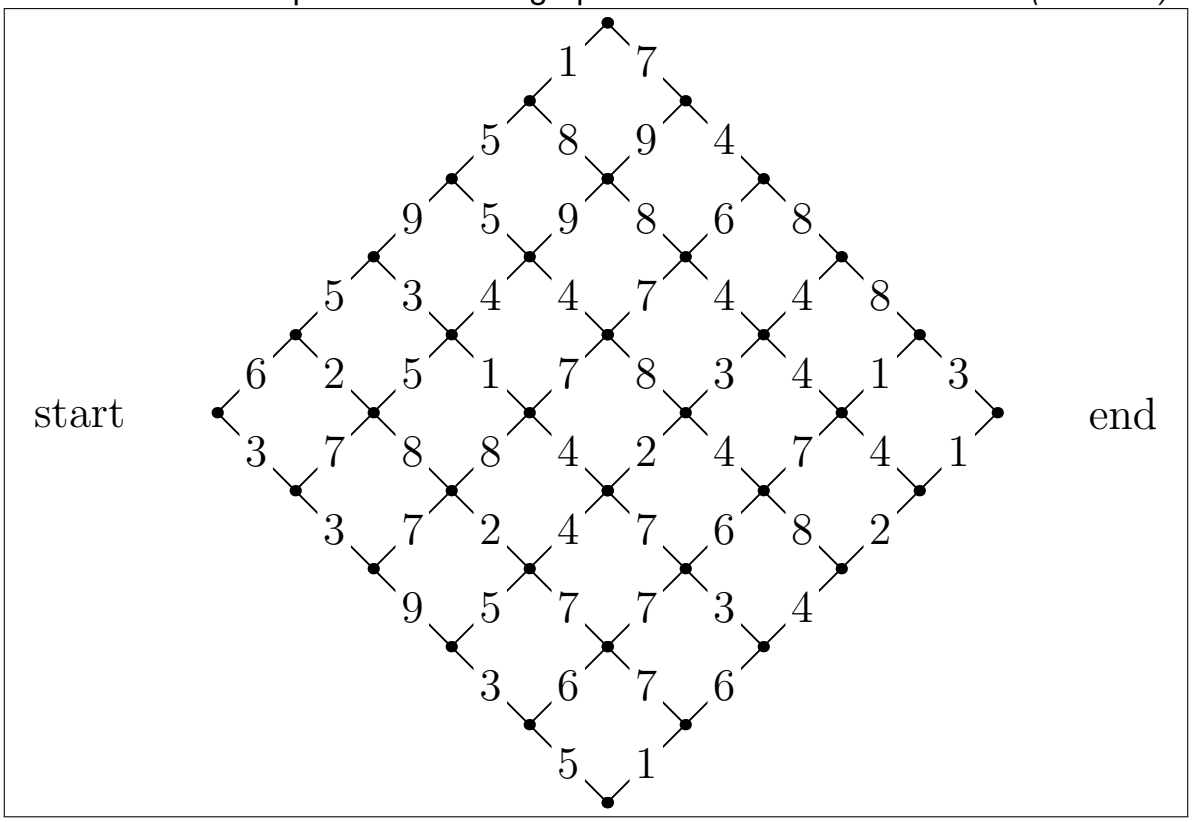
(e) Show how the numbers 12, 3, 7, 19, 15, 52, 46, 23, 22 would be hashed using a hash function $d_2 + 3d_1$ where d_1 is the first (least significant) digit and d_2 the second digit. Show how these would be stored in a hash table using separate chaining. *(10 marks)*

d_2d_1	12	3	7	19	15	52	46	23	22
$d_2 + 3d_1$									
$(d_2 + 3d_1)\%10$									

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

10

(f) Use the dynamic programming forward algorithm to compute the minimum cost of each path from the left most node to each other node where the cost of moving along an edge is equal to the number shown. An edge can only be traversed from left to right. Use the backwards algorithm to find the minimum cost path across the graph. *(5 marks)*



5

End of question 1

Q1: (a) $\frac{}{5}$	(b) $\frac{}{1}$	(c) $\frac{}{2}$	(d) $\frac{}{2}$	(e) $\frac{}{10}$	(f) $\frac{}{5}$	Total $\frac{}{25}$

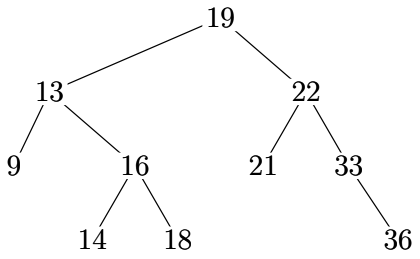
Section B

Question B 2

(a) Draw the binary search tree produced when inserting 64, 48, 35, 50, 26, 77, 43, 94, 0, 61. *(2 marks)*

2

(b) Draw the tree obtained by deleting 22 from the binary search tree shown. *(2 marks)*




```
delete(22)
```

2

(c) Describe the two rules that define an AVL tree *(4 marks)*

1

2

4

(d) Let $m(h)$ be the minimum number of elements in an AVL tree of height h. Write down a recurrence relationship for $m(h)$ *(4 marks)*

4

(e) Write down the boundary condition for $m(1)$ and $m(2)$ *(2 marks)*

2

(f) Using proof by induction show that the minimum number of elements in an AVL tree of height h is greater than or equal to $b(h) = \left(\frac{3}{2}\right)^{h-1}$ *(6 marks)*

6

(g) Use this bound to show that the complexity of insertion and search for an AVL tree is $O(\log(n))$. *(3 marks)*

3

(h) What is the average and worst case time complexity of search in an unbalanced binary search tree and why is balancing a binary search tree regarded as so important? *(3 marks)*

Average case complexity:

Worst case complexity:

$\frac{}{3}$

(i) Describe how a tree map is implemented in the Java collections. *(4 marks)*

$\frac{}{4}$

End of question 2

Q2: (a) $\frac{}{2}$ (b) $\frac{}{2}$ (c) $\frac{}{4}$ (d) $\frac{}{4}$ (e) $\frac{}{2}$ (f) $\frac{}{6}$ (g) $\frac{}{3}$ (h) $\frac{}{3}$ (i) $\frac{}{4}$ Total $\frac{}{30}$

TURN OVER

Question B 3

(a) Describe the difference between depth first search and breadth first search. Give an application of each. *(4 marks)*

DFS Application:

BFS Application:

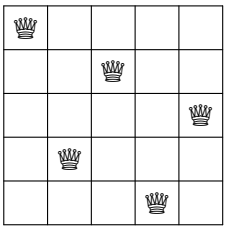
4

(b) What is topological sort? Give an application. *(4 marks)*

Application:

4

(c) The n-queens problem is to put n queens on a chess board such that no queen is in the same row, column or diagonal as any other queen. Either write pseudo code or describe in outline an algorithm to solve the n-queens problem.



(10 marks)

10

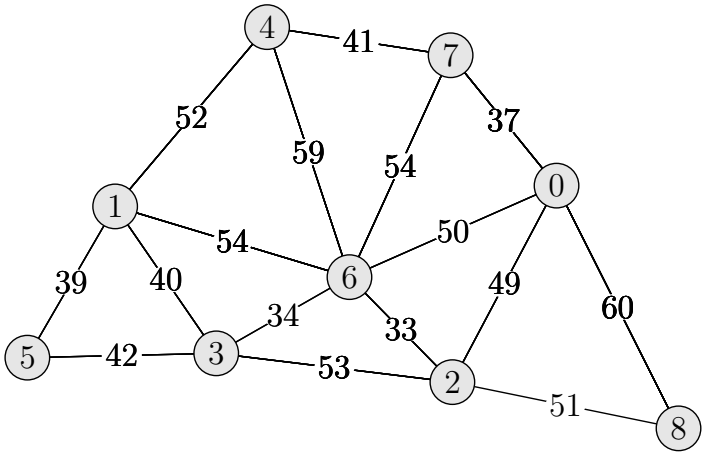
TURN OVER

(d) Describe Kruskal's algorithm for constructing a minimum spanning tree. What data structures are need to implement the algorithm? *(6 marks)*

Data Structures:

6

(e) In the graph below draw the minimum spanning tree (by highlighting the edges) and write down the order in which the edges would be found by Kruskal's algorithm. *(6 marks)*



6

1. **2.**

3. **4.**

5. **6.**

7. **8.**

End of question 3

Q3: (a) $\frac{}{4}$ (b) $\frac{}{4}$ (c) $\frac{}{10}$ (d) $\frac{}{6}$ (e) $\frac{}{6}$ Total $\frac{}{30}$

Question B 4

(a) Describe the local (neighbourhood) search strategy. What is its main drawback? *(4 marks)*

Drawback:

4

(b) Describe a strategy to overcome this. *(4 marks)*

4

(c) How efficient would you expect heuristic search to be on a problem like TSP or graph colouring? *(2 marks)*

2

(d) Describe the three conditions required of a linear programing problem. *(6 marks)*

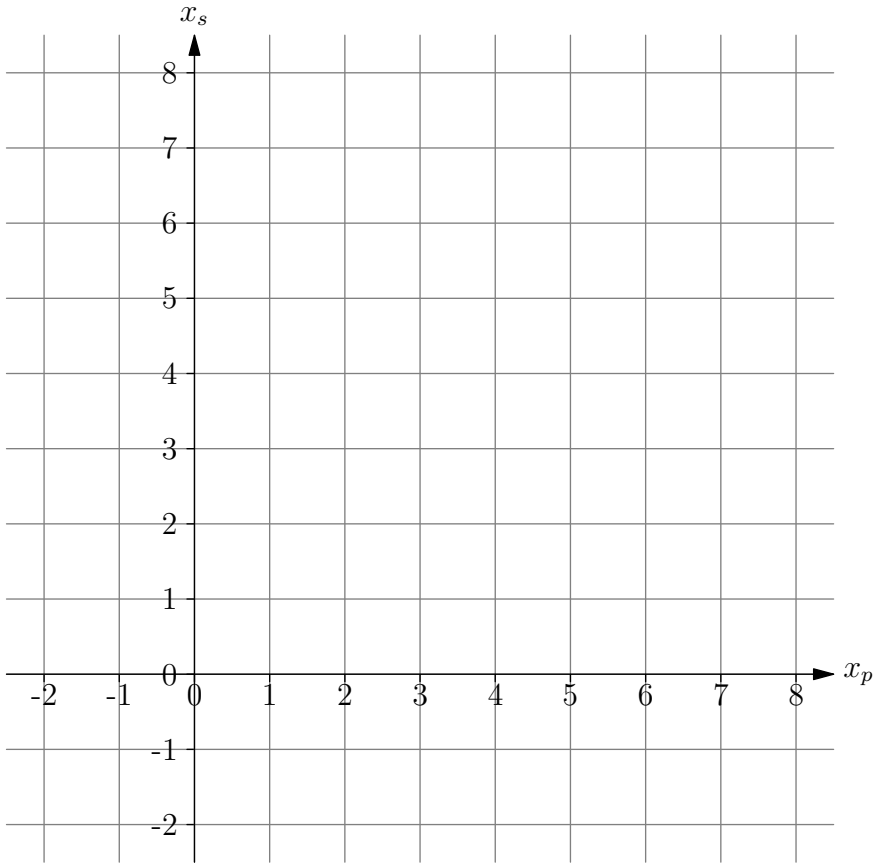
1

2

3

6

(e) Consider the problem of deciding the cheapest balance between potatoes and soybean. Let x_p be the amount of potatoes and x_s the amount of soybeans. The cost of soybean is taken to be twice the cost of potatoes. We require the vitamin C content from the two ingredients should be at least 12 units where the vitamin content of 1 unit of potatoes is 2 and for soybean 3. We also require the vitamin B6 content of the two ingredients should be at least 8 units where one unit of potatoes supplies 4 while soybean provides 1 unit. Write down the linear programming problem and draw the direction that minimises the cost and the constraints (shade the infeasible region). Mark the optimal solution. *(10 marks)*



10

(f) Describe in brief outline the simplex algorithm. *(4 marks)*

$\frac{}{4}$

End of question 4

Q4: (a) $\frac{}{4}$ (b) $\frac{}{4}$ (c) $\frac{}{2}$ (d) $\frac{}{6}$ (e) $\frac{}{10}$ (f) $\frac{}{4}$ Total $\frac{}{30}$

END OF PAPER