JOHN MONASH SCIENCE SCHOOL

VICTORIAN CERTIFICATE OF EDUCATION

2018

ALGORITHMICS (HESS)

Practice Exam 1

2018

SOLUTIONS

SECTION A – Multiple-choice Questions

Question	%A	%B	%C	%D	%No Answer	Comments
1	5	93	3	0	0	Correct Answer : B
2	53	35	5	5	0	Correct Answer : A
3	20	78	3	0	0	Correct Answer : B
4	25	10	3	60	3	Correct Answer : D
5	8	5	75	8	3	Correct Answer : C
6	15	25	8	50	0	Correct Answer : D
7	85	10	3	0	0	Correct Answer : A
8	25	5	3	63	0	Correct Answer : D
9	5	13	78	3	3	Correct Answer : C
10	85	5	3	8	0	Correct Answer : A
11	13	8	15	63	0	Correct Answer : D
12	3	10	13	70	0	Correct Answer : D
13	3	93	5	0	0	Correct Answer : B
14	75	8	13	3	3	Correct Answer : A
15	20	0	8	70	0	Correct Answer : D
16	48	23	23	5	0	Correct Answer : A
17	3	3	88	3	3	Correct Answer : C

18	15	55	18	13	0	**Correct Answer :** B
19	8	0	38	50	0	**Correct Answer :** C
20	30	35	3	28	3	**Correct Answer :** A

SECTION B

Question 1 (3 marks)

Kade has been given the role of Deputy Operational Performance Enforcer (DOPE) at the local Christmas Tree farm. In this role, he is required to place 'watchers' in specific paddocks around the farm to ensure that no illegal activities occur in each of these paddocks. When a watcher is in a given paddock, they can supervise that paddock as well as all immediately adjacent paddocks.

Outline an algorithm that Kade could use to determine the placement of watchers in paddocks around the farm in a reasonable amount of time. As part of your answer, identify an appropriate abstract data type (ADT) that could be used, referring to its specifications.

Marks	**0**	**1**	**2**	**3**	**Average**
%	8	28	30	25	**1.63**

Kade could treat this problem by modelling it as a graph.	A1
By representing each paddock as a node on a graph and corresponding edges representing paddocks which are sharing borders, he could then proceed to solve the problem.	A2
He could proceed to colour a given node one colour to indicate the location of a watcher and proceed to colour adjacent nodes another colour to indicate that they are supervised. The initial selection of which node to place a watcher at might be random or use a heurtistic such as the highest degree node remaining uncoloured in the graph. He could repeat this process until all nodes are coloured. Accept any other algorithmic approach that would result in the graph being coloured.	A3

Question 2 (6 marks)

Sai has recently given up on world domination due to an overwhelming feeling of 'meh'. Instead, he has focussed his attention on his colony of fire ants (they were an important part of his original plan to take over the world but that's not really relevant anymore).

The fire ants have created a colony of rooms that are connected by tunnels. Sai wishes to make his ants into a species of super ants through the application of gamma rays but has noticed that the fire ants have a tendency to burst into flames when irradiated. As a result, he has installed water sprinklers in each of the rooms in the colony as a way of trying to minimise the spread of any fires that might start.

When one of the fire ants bursts into flames, the following algorithm is used to determine which rooms in the colony the sprinklers should be turned on in so that the flames can's spread further.

```
Algorithm Hydrate(L)
input:   L a list of all of the rooms that are connected by a
         tunnel to the room in which the fire started in order of
         distance from the starting room.

While L is not empty
         Engage Sprinklers in the first element of L
         Remove the last element in L
```

a. The proposed algorithm has errors.

How many rooms in the fire ant colony will have the sprinklers turned on when a fire ant bursts into flames in a room which is connected to 6 adjacent rooms? 1 mark

Marks	**0**	**1**	**Average**
%	38	60	**0.60**

1 room (it will have the sprinklers turned on six times)	A1

b. Write a corrected version of the algorithm that will allow the sprinklers to be turned on in all of the adjacent rooms to the room where the fire ant burst into flames. 2 marks

Marks	**0**	**1**	**2**	**Average**
%	0	10	88	**1.85**

`While L is not empty`
`        Engage Sprinklers in the first element of L`
`        Remove the FIRST element in L`
 	
An attempt to correct this algorithm that isn't quite accurate but that is on the right track should be awarded one mark
	A1 + A2

c. Sai notices that when the sprinklers turn on, the water begins to spread through the ant colony, flooding all of the adjacent rooms and extinguishing the lives of his precious fire ants.

What ADT would be most appropriate for modelling the rooms of the ant colony? Assuming that the water from the sprinklers spreads to all connected rooms to the one where the sprinkler was turned on, what type of algorithm could be used to with your selected ADT to model the flow of the water through the ant colony? 3 marks

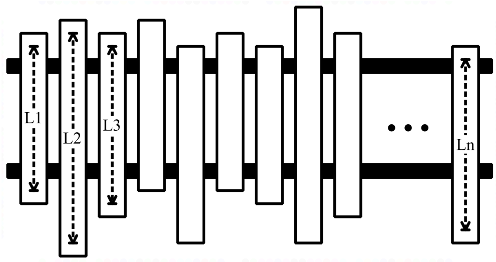
Marks	0	1	2	3	Average
%	0	18	15	63	**2.35**

The graph ADT would be the most appropriate for representing the rooms of the ant colony	A1
A breadth first search algorithm would be the most appropriate for modelling the spread of water through the rooms	A2
As the water must flow to adjacent rooms first before moving on to the next group of rooms.	A3

Question 3 (12 marks)

Sean and Parker have been given a work management by Mr Chisholm for claiming that the Melbourne Demons weren't the best team in the AFL (an obvious fallacy). As part of their punishment, they have been told to organise the painting of a new picket fence (in red and blue of course).

There are 5 painters available and each painter takes 1 unit of time to paint 1 unit length of fence. The fence is made up of vertical pickets of length {L1, L2, L3,… Ln}.



Sean and Parker wish to find the minimum time required to get the fence painted given that a picket can only have one painter and any given painter will only paint adjacent pickets (for example, an individual painter might paint pickets 2, 3 and 4 but they won't paint pickets 2, 4 and 6).

a. Write a brute-force algorithm that will return the minimum time required to paint the fence. 4 marks

Marks	**0**	**1**	**2**	**3**	**4**	**Average**
%	33	8	13	10	23	**1.53**

Below is an example of a high scoring response to this question:

```
n = number of pickets in the fence
Min = ∞
for i = 1 to i = n - 4
    for j = i + 1 to j = n – 3
        for k = j + 1 to k = n – 2
            for m = k + 1 to m = n – 1
                for p = m + 1 to p = n
                    POne = sum(L1, … ,Li)
                    PTwo = sum(Lj, … ,L(k-1))
                    PThree = sum(Lk, … ,L(m-1))
                    PFour = sum(Lm, … ,L(p-1))
                    PFive = sum(Lp, … ,Ln)
                    Time = max of (POne, PTwo, PThree, PFour, PFive)
                    If Time < Min
                        Min = Time
Return Min
```

Algorithm gives partial solution to the problem	A1
Algorithm uses a brute force approach	A2
Algorithm will correctly find the solution to the problem	A3
Algorithm returns minimum time required to paint the fence	A4

b. Riley, who was also involved in spreading blatant lies about the Melbourne Demons but who didn't get caught by Mr Chisholm, was feeling guilty and so wrote the following two algorithms to try and help Sean and Parker with their problem.

```
SumsAlgo(A[ ], start, end)
      Input A[ ]: An array of the lengths of each picket in
      order.
      Input start: an integer value indicating the index in A
      from which to begin.
      Input end: an integer value indicating the index value in A
      at which to end.

      Total = 0
      for i = start to i = end
            Total = Total + A[i]
      return Total

RileysAlgorithm(A[ ], n, k]
      Input A[ ]: An array of the lengths of each picket in
      order.
      Input n: The total number of pickets in the fence.
      Input k: The number of painters that are available to paint
      the picket fence.

      Let Split = ⌊ n / k ⌋
      Best = 0
      for i = 0 to i = k - 1
            SumsAlgo(A[ ], i*(Split + 1), (i+1)*(Split+i))
            if Total > Best
                  Best = Total
      return Best
```

i. Which algorithmic design patterns has Riley made use of in the design of his algorithms? 2 marks

Marks	0	1	2	Average
%	30	63	5	**0.73**

Riley's algorithm can be considered to utilise divide and conquer techniques as it divides the problem into smaller sub-problems that each need to be solved.	A1
Riley's algorithm can also be considered to be a greedy algorithm as it makes the assumption that the best solution will evenly split the pickets amongst the available painters.	A2

ii. Write a recurrence relation that reflects the running time of RileysAlgorithm as it is written. 2 marks

Marks	0	1	2	Average
%	70	0	0	**0.00**

This question was flawed as the algorithm is not recursive and hence a recurrence relation cannot be found. This question was not counted in the grading of the exam.
A better question would have been to determine the time complexity of Riley's algorithm. A suggested marking scheme for this question is shown below:

Identification of the SumsAlgo to have a time complexity of $O(n)$	A1
Overall complexity of RileysAlgorithm is $O(n^2)$	A2

c. Compare a brute force approach like the one used in **part a** with the approach used by Riley in **part b**.
Which approach should Sean and Patrick utilise in solving this problem. Justify your answer with appropriate reasoning. 4 marks

Marks	**0**	**1**	**2**	**3**	**4**	**Average**
%	18	15	48	3	10	**1.58**

A brute force approach is guaranteed to produce the correct answer but will require the generation of every possible way of solving the problem.	A1
An approach such as the one used by RileysAlgorithm can find a solution a lot more quickly but this solution may not be the optimal one.	A2
As time is an important consideration for Sean and Patrick in finding a solution to the problem (and they are not overly concerned with how long it will take for the actual fence to be painted), they should utilise something along the lines of RileysAlgorithm. ***Note: Justification of choice of approach is required to obtain both marks. If students made a valid but different argument to the one shown above, they can still be awarded full marks for this question but argument must be appropriately supported and reasonable.**	A3 + A4

Question 4 (9 marks)

Aaron is wanting to compute the shortest path for his video game character to journey from the town tavern to the Caves of Doom. He knows that locations in his game are represented as nodes in a graph and their distances are represented as weights on the connected edges. Aaron wishes to use Dijkstra's algorithm to find the shortest route for his game character to traverse.

a. By applying a Proof by Contradiction, prove that Dijkstra's algorithm will always correctly determine the shortest path between two nodes on a graph that does not contain any negative edge weights. 4 marks

Marks	0	1	2	3	4	Average
%	23	18	10	13	10	**1.15**

Below is an example of a high scoring response to this question:

- Assume that the minimum distance between nodes A and X is given by d(A,X).
- Assume that the minimum distance calculated by Dijkstra's Algorithm between nodes A and X is given by D(A,X) when node X is entered into the reached set.
- If Dikstra's algorithm is correct then D(A,X) = d(A,X) when node X is entered into the reached set.
- Let us assume that this **is incorrect** and that, in fact, D(A,X) > d(A,X) when X is entered into the reached set.
- Let us also assume that node X is the first node for which this inaccuracy occurs and for all prior nodes Dijkstra's algorithm did return the actual minimum distance between the nodes.
- Let Z be a node that is on the shortest path towards node X that has not yet been entered into the reached set.
- Let Y be the node immediately before node Z on the shortest path that has been entered into the reached set.
- As X was the first inaccurate addition, we know that D(A,Y) = d(A,Y).
- We also know that the shortest path from node A to Z is given by the distance from A to Y and the shortest connection between A and Y. That is D(A,Z) = d(A,Y) + d(Y,Z)
- Following on, as Y and Z are both on the shortest path to Z, the shortest path must therefore be d(A,X) = d(A,Y) + d(Y,Z) + d(Z,X)
- We must assume that D(A,X) < D(A,Z) as node X was chosen by Dijkstra's before node Z was.
- As D(A,X) < D(A,Z), we can logically follow that D(A,X) < d(A,Y)+d(Y,Z) which in turn must also mean that D(A,X) < d(A,Y) + d(Y,Z) + d(Z,X) as we are assuming no negative edge weights.
- This leads us to the statement that D(A,X) < d(A,X) which is a direct contradiction to our initial assumption.

Correct initial statement and set up of assumptions.	A1
Identification of a statement that will be assumed incorrect. (Equivalent statement of dot point 4)	A2
Partial attempt to logically follow through the assumptions .	A3
Valid argument proposed for the correctness of the algorithm.	A4

b. Unfortunately, the character never seems to budge from his initial position, and Aaron suspects that there is a negative cycle somewhere in his graph configuration.

He remembers how the Floyd-Warshall algorithm can be adapted to detect negative cycles in graphs and decides to give this a try.

In pseudocode, write a version of the Floyd-Warshall algorithm that Aaron could use for his problem. 4 marks

Marks	**0**	**1**	**2**	**3**	**4**	**Average**
%	15	5	18	10	33	**2.00**

Below is an example of a high scoring response to this question:

```
Create an array called distance of size n x n //where n is the number
of nodes in the graph
Initialise all values in the array to be ∞
for each node, u, in the graph
    for each node, v, in the graph
            distance[u,v] = the weight of the edge joining u and v
for each node, n, in the graph
    distance[n,n] = 0
for each node, k, in the graph
    for each node, i, in the graph
        for each node, j, in the graph
            if distance[i,j] > distance[i,k] + distance[k,j]
                Update distance[i,j] = distance[i,k] + distance[k,j
```

Creating of the array and initialisation of all values to ∞ and addition of edge weights for adjacent nodes	A1
Second for loop to make distance from a node to itself equal to 0	A2
Attempt at 3 nested for loops	A3
Accurate If statement to compare distances between nodes.	A4

c. Explain how Aaron can adapt the algorithm to detect whether or not there are negative cycles in a given graph. 1 mark

Marks	**0**	**1**	**Average**
%	53	40	**0.40**

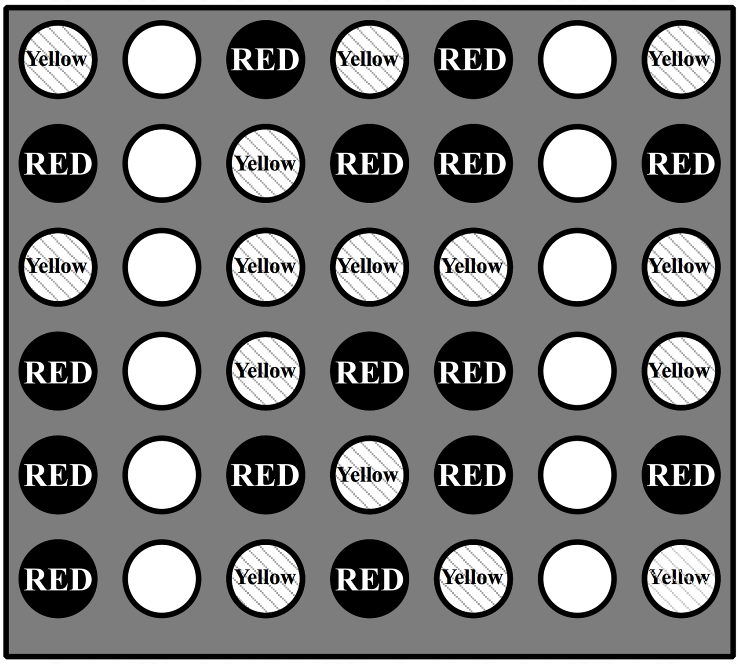
As each node initially has a distance of 0 from itself to itself, the only way this can become negative if is there is a negative cycle present. Thus, by looking for a negative number that appears along the diagonal of the array Aaron can identify if a negative cycle appears.	A1

Question 5 (8 marks)

The game of Connect-4 is a ***solved*** game, which means that an outcome can be predicted from any state of the game with the assumption that both players play perfectly.

Below is the current state of the game of Connect-4 between Vlad and David. It is currently Vlad's turn and he is playing as RED*.

*In case you are unsure, as Vlad is RED, it means that David is playing as Yellow.



a. Explain how the Minimax algorithm would go about determining what move Vlad should make as his next move. As part of your answer make sure that you reference any assumptions that the Minimax algorithm would make. 3 marks

Marks	**0**	**1**	**2**	**3**	**Average**
%	3	30	53	15	**1.80**

Minimax will generate a tree of all possible moves from the current position	A1
Minimax will assume that David will always make the optimum move available to him on his turn and will then assign a value to each end state based on who wins (1 for Vlad winning, -1 for David winning and 0 for drawing)	A2
It will then work backwards to assign either a maximal value possible (if Vlad's turn) or a minimal value possible (if David's turn) until it returns to the current state where it can then tell Vlad which move has the highest value that he should then make.	A3

b. Draw the Minimax state-space graph the illustrates all of the possible moves from this current state until an outcome is obtained.

Indicate on your state-space graph what move Vlad should make in order to maximise his chance of beating David. 3 marks

Marks	0	1	2	3	Average
%	15	45	5	28	**1.38**

This question, in hindsight, was far too involved to warrant the drawing of the entire tree. Students who succeeded beyond 3 levels of the tree were considered to have answered the question sufficiently.

An attempt to draw the state tree has been made (with minimal errors if any)	A1
An attempt to label each state with a value based on each turn has been made (with minimal errors if any).	A2
Identification of the correct move that Vlad should make based off the labels.	A3

c. How would the feasibility of using Minimax to determine the next move change if Vlad were to try and use it from the very beginning of the game? 2 marks

Marks	0	1	2	Average
%	20	20	60	**1.40**

Running minimax from the beginning of the game is not feasible as the number of the states that the algorithm would be required to generate would become exceptionally large.	A1 + A2

Question 6 (3 marks)

Describe the Chinese Room Argument as defined by John Searle. In your response refer to the ideas of strong AI and weak AI and how the Chinese Room Argument supports or does not support these ideas.

Marks	0	1	2	3	Average
%	10	13	40	35	**1.98**

John Searle proposed that a computer could be thought of as a non-chinese speaking man who was inside a room with a book of instructions. Someone would pass messages, in Chinese, under the door of the room and the man could follow the instructions to create a response that would convince the person outside of the room that whoever was inside understood Chinese perfectly whereas, in reality, the man has no idea what was said.	A1
Searle uses this argument to propose the idea that computers are only capable of simulated intelligence (Weak AI) and are never truly intelligent (Strong AI). That is to say that they create the perception that they have understanding when in actual fact they are simply following instructions.	A2 + A3

Question 7 (6 marks)

The Church-Turing Thesis and Cobham's Thesis both attempt to define the nature of computability.

a. Outline how the Church-Turing Thesis attempts to define computability and how it was influenced by Hilbert's 1927 Program. 3 marks

Marks	**0**	**1**	**2**	**3**	**Average**
%	5	35	35	18	**1.58**

The Church-Turing Thesis defines computability as being anything that is capable of being computed by a Turing machine that has been given infinite time and resources.	A1
Both Church and Turing came up with their equivalent ideas (Turing Machines and Lambda Calculus respectively) in an attempt to find a way of approaching Hilbert's idea of creating a program from which it would be possible to determine if all of mathematics was decideable.	A2
The Church-Turing Thesis was a step in defining an ordered way of creating the mathematics from logical principles.	A3

b. Outline how Cobham's Thesis attempts to define computability. As part of your answer, discuss some of the limitations of Cobham's Thesis in defining computability in this manner. 3 marks

Marks	**0**	**1**	**2**	**3**	**Average**
%	8	8	20	55	**1.58**

Cobham's Thesis attempts to define computability as those problems that are feasibly computable are those that can be computed in polynomial time.	A1
A limitation of this theory is that it neglects to take into account that some polynomial time complexity classes should still be considered intractable as problems such as O(n^100) for example quickly become difficult to find solutions to in reasonable amounts of time as n grows large.	A2
In contrast, some non-polynomial time complexity classes are tractable for significantly large values of n but would not be considered computable by Cobham's definition.	A3

Question 8 (10 marks)

Daniel has decided to take on Google and write his very own search engine which will return the best results for how to write English essays.

One of the first steps he has identified to write is a web 'crawler' which is a program that will, given a starting web page, follow all of the links from that page and save them on a separate disk.

a. What ADTs would you recommend Daniel utilise when writing this web crawler. 2 marks

Marks	0	1	2	Average
%	5	65	25	**1.15**

A graph ADT to represent the pages and their links to other pages.	A1
A list or a queue with which to store the links as they are followed.	A2

As part of his research into designing his very own search engine, Daniel takes a look at the PageRank algorithm and how it works. Thanks to a rather mysterious online acquaintance that Daniel met (who goes by the pseudonym Mr Treyment), Daniel comes across the following equation for PageRank that would act upon a series of 5 webpages, *A*, *B*, *C*, *D* and *E*.

$$\mathrm{PR}(A) = \frac{(1-d)}{N} + d\left(\frac{\mathrm{PR}(B)}{\mathrm{L}(B)} + \frac{\mathrm{PR}(C)}{\mathrm{L}(C)} + \frac{\mathrm{PR}(D)}{\mathrm{L}(D)} + \frac{\mathrm{PR}(E)}{\mathrm{L}(E)}\right)$$

b. **i.** What does the $\mathrm{PR}(A)$ represent? 1 mark

Marks	0	1	Average
%	13	80	**0.80**

The page rank value assigned to page *A*, also considered the probability of a surfer reaching the web page whilst surfing.	A1

ii. What does the d term represent? 2 marks

Marks	0	1	2	Average
%	30	25	33	**0.90**

The damping factor	A1
which represents the probability of a surfer following a link from this page to another.	A2

iii. What does the $\frac{(1-d)}{N}$ term represent? 1 mark

Marks	0	1	Average
%	40	38	**0.38**

The probability that a surfer will randomly click on a page rather than following a link.	A1

iv. What does the $L(B)$ term represent? 1 mark

Marks	0	1	Average
%	33	53	**0.53**

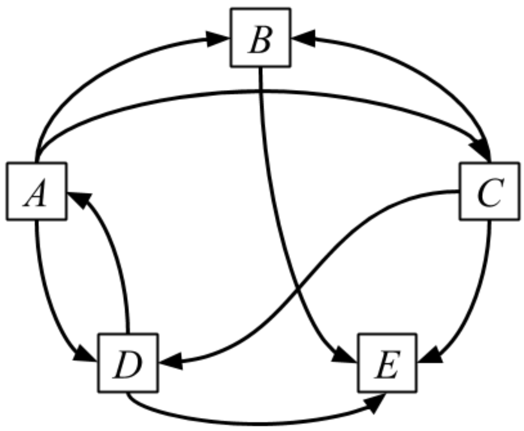
The number of outgoing links from page *B*	A1

v. What does the $d\left(\frac{\mathrm{PR}(B)}{\mathrm{L}(B)}+\frac{\mathrm{PR}(C)}{\mathrm{L}(C)}+\frac{\mathrm{PR}(D)}{\mathrm{L}(D)}+\frac{\mathrm{PR}(E)}{\mathrm{L}(E)}\right)$ term represent? 1 mark

Marks	0	1	Average
%	40	40	**0.40**

The probability of a surfer reaching the web page through clicking on a link from one of the pages linking to page *A*	A1

The series of webpages from **part b** can be represented by the following network graph:



c. The webpage *E* has no outgoing links. Describe how PageRank would deal with the values attributed to webpage *E*. 2 marks

Marks	0	1	2	Average
%	48	5	28	**0.60**

It would treat this page as a sink and redistribute all of the ranking from this page evenly amongst all of the other pages in the network.	A1
This could be achieved by adding imaginary links from node *E* to all other nodes.	A2

Question 9 (4 marks)

Traveen and Lleyton are designing a program that uses a variety of different algorithms of different time complexities. Lleyton wishes to run the algorithms sequentially, one after the other, whereas Traveen wishes to 'nest' the algorithms with each algorithm calling another as it runs.

Discuss how the variation in complexity can affect the overall running time of the program if they are nested (Traveen's idea) compared with run sequentially (Lleyton's idea). In your response, use examples of time complexity.

Marks	**0**	**1**	**2**	**3**	**4**	**Average**
%	10	5	8	13	53	**2.68**

By running the algorithms sequentially, the overall running time of each algorithm will be added together and the largest complexity class algorithm will dictate the overall complexity.	A1 + A2
If the algorithms are nested together, they can result in the time complexities multiplying together to create a significant increase in the overall time complexity.	A3 + A4

Question 10 (8 marks)

Emerging technologies may help to overcome some of the limits of current computing systems.

a. Describe **one** limit that exists in current models of computing. 2 marks

Marks	0	1	2	Average
%	25	13	58	**1.28**

Many problems are simply not solveable due to the demand on resources (time or space) required by them	A1
NP problems are such examples as we have ways of solving these problems but the length of time required to do so often exceeds the time that we have available in which to find a solution.	A2

b. Describe **one** way a neural network could be used to overcome current limits of computation. 3 marks

Marks	0	1	2	3	Average
%	23	30	23	20	**1.35**

Neural networks allow us to train a network to do something that is often very difficult to write down in traditional algorithmic ways.	A1
By creating an initial structure and repeatedly tweaking it through the use of training data it is possible to get a neural network to perform tasks that would otherwise be very difficult to achieve.	A2
Neural networks are very good at classification and recognition problems.	A3

c. Describe **one** way DNA computing could be used to overcome current limits of computation. 3 marks

Marks	0	1	2	3	Average
%	10	33	28	23	**1.55**

DNA computing can be considered to be mass parallel processing.	A1
What this enables us to do is find solutions to large problems faster by having such large amounts of DNA working simultaneously to generate a solution.	A2
While this does speed up the process, it does not change the big O complexity class of a problem.	A3

Question 11 (6 marks)

Ryan and Raj have been asked by the JMSS IT department to help lay out the network cabling for a new supercomputing lab! There will be 7 new PCs (creatively named **a**, **b**, **c**, **d**, **e**, **f**, **g**) in the lab in addition to the supercomputer.

Below is the adjacency matrix that describes the distances, in meters, between the location of each of the computers in the supercomputing lab.

	a	**b**	**c**	**d**	**e**	**f**	**g**
a		4	8				
b	4		9	8	10		
c	8	9		2		1	
d		8	2		7	9	
e		10		7		5	6
f			1	9	5		2
g					6	2	

a. Draw a graphical representation of the connections between the computers in the supercomputing lab. As part of your answer, include a legend explaining what each aspect of the graph represents. 3 marks

Marks	**0**	**1**	**2**	**3**	**Average**
%	0	3	8	88	**2.80**

Inclusion of a legend describing each aspect of the graph utilised.	A1
An attempt to draw a graphical representation of the data has been made perhaps with some inaccuracies or omissions.	A2
Accurate graphical representation of the data has been made without any error.	A3

b. Ryan and Raj wish to determine what the minimum amount of cabling would be in order to ensure that each of the seven computers, **a**, **b**, **c**, **d**, **e**, **f**, **g**, are connected to the same network.

What algorithm would you recommend for Ryan and Raj use? As part of your answer, provide a reason for why this algorithm would be **most** appropriate. 2 marks

Marks	**0**	**1**	**2**	**Average**
%	10	3	85	**1.73**

Prim's Algorithm would be the most effective algorithm to use.	A1
Prim's allows you to find the minimal spanning tree for the graph which, in turn guarantees that each computer is connected to the same network using the least amount of cable necessary.	A2

c. What is the minimum amount of cabling required to ensure that each of the seven computers, **a**, **b**, **c**, **d**, **e**, **f**, **g**, are connected to the same network? 1 mark

Marks	**0**	**1**	**Average**
%	25	73	**0.73**

22 metres	A1

Question 12 (5 marks)

Josh, while unwell in hospital was thinking up ways to torment those who were responsible for putting him there. Patrick was at the top of his list.

The problem that Josh came up with was based off the following grid.

3	6	8
9	1	4
7	5	2

Patrick was asked to find the maximum length path, starting from any cell in the grid, such that all of the cells along the path are in increasing order with a difference of 1.

a. Describe an algorithm design pattern that Patrick could use to solve Josh's problem. As part of your answer, make sure that you discuss the features of this problem that make this algorithm design pattern appropriate. 3 marks

Marks	**0**	**1**	**2**	**3**	**Average**
%	33	25	5	20	**0.95**

Dynamic Programming	A1
Patrick could calculate small portions of the problem that he then stores for later recall.	A2
Then, when considering a solution that brings him to a space where he has already calculated the maximum value, he can recall that value instead and save time.	A3

b. Josh considered using a larger size grid to torment Patrick but was convinced by Matt that the 3x3 grid would be difficult enough for Patrick. Describe how the difficulty of the problem would be affected by increasing the size of the dimensions of the grid. 2 marks

Marks	**0**	**1**	**2**	**Average**
%	18	28	40	**1.08**

The problem would get much harder.	A1
With each increase in the size of each dimension, the number of combinations that need to be considered grows combinatorially.	A2