VICTORIAN CURRICULUM
AND ASSESSMENT AUTHORITY

Victorian Certificate of Education
2019

SUPERVISOR TO ATTACH PROCESSING LABEL HERE

STUDENT NUMBER | | | | | | | | | Letter: |

ALGORITHMICS (HESS)

Written examination

Tuesday 19 November 2019

Reading time: 3.00 pm to 3.15 pm (15 minutes)
Writing time: 3.15 pm to 5.15 pm (2 hours)

QUESTION AND ANSWER BOOK

Structure of book

Section	*Number of questions*	*Number of questions to be answered*	*Number of marks*
A	20	20	20
B	16	16	80
			Total 100

- Students are permitted to bring into the examination room: pens, pencils, highlighters, erasers, sharpeners, rulers and one scientific calculator.
- Students are NOT permitted to bring into the examination room: blank sheets of paper and/or correction fluid/tape.

Materials supplied
- Question and answer book of 29 pages
- Answer sheet for multiple-choice questions

Instructions
- Write your **student number** in the space provided above on this page.
- Check that your **name** and **student number** as printed on your answer sheet for multiple-choice questions are correct, **and** sign your name in the space provided to verify this.
- All written responses must be in English.

At the end of the examination
- Place the answer sheet for multiple-choice questions inside the front cover of this book.

Students are NOT permitted to bring mobile phones and/or any other unauthorised electronic devices into the examination room.

THIS PAGE IS BLANK

SECTION A – Multiple-choice questions

Instructions for Section A

Answer **all** questions in pencil on the answer sheet provided for multiple-choice questions.

Choose the response that is **correct** or that **best answers** the question.

A correct answer scores 1; an incorrect answer scores 0.

Marks will **not** be deducted for incorrect answers.

No marks will be given if more than one answer is completed for any question.

Use the Master Theorem to solve recurrence relations of the form shown below.

$$T(n) = \begin{cases} aT\left(\frac{n}{b}\right) + kn^c & \text{if } n > 1 \\ d & \text{if } n = 1 \end{cases} \quad \text{where } a > 0,\ b > 1,\ c \geq 0,\ d \geq 0,\ k > 0$$

and its solution $T(n) = \begin{cases} O(n^c) & \text{if } \log_b a < c \\ O(n^c \log n) & \text{if } \log_b a = c \\ O(n^{\log_b a}) & \text{if } \log_b a > c \end{cases}$

Question 1

A stack, S, contains the elements

S = [0, 8, 18, 12, 31, 77]

where the first element is the top of the stack.

S.push(75)
S.pop()
S.pop()
S.push(31)
S.pop()
S.push(8)

What does S look like once the operations above are executed in order?

A. S = [12, 31, 77, 75, 31, 8]

B. S = [8, 31, 75, 0, 8, 18]

C. S = [0, 8, 18, 12, 31, 8]

D. S = [8, 8, 18, 12, 31, 77]

Question 2

A Turing machine is run with a set of instructions designed to solve a problem. The machine is run multiple times using randomly selected inputs. On some inputs the machine halts and accepts the input and on all other inputs it halts and rejects the input. When the machine halts, the solution produced by the input can be quickly verified.

Which one of the following statements is definitely true?

A. The problem is decidable and in NP.

B. The problem is undecidable and in NP.

C. The problem is decidable and not in NP.

D. The problem is undecidable and not in NP.

Use the following information to answer Questions 3 and 4.

A country has six states. Each state has its own major city. Below is a table of the major cities (A–F) and their distances apart in kilometres (km).

City						
F						
E						720
D					550	1060
C				860	400	250
B			940	1800	1380	750
A		1600	960	1160	640	1160
	A	B	C	D	E	F

Question 3

Veronica has a free ticket that allows her to travel at most 2000 km starting from any city. She wants to visit as many cities as possible so she chooses to visit the cities that are nearest each other first. She does not need to return to the city where she starts her journey.

The three most probable cities that Veronica will visit, in order, are

A. A → B → C

B. B → C → D

C. F → C → E

D. F → C → D

Question 4

Eliana has free tickets that allow her to travel a maximum of 10 000 km. However, she wants to visit only cities A, B, C and D, and she wants to save the remaining tickets to maximise her future travels. She does not need to return to the city where she starts her journey.

What will be Eliana's best route?

A. A → B → C → D

B. A → D → C → B

C. C → D → B → A

D. D → A → C → B

Question 5

Stan is storing a series of timing observations for a function he is writing so he can calculate the average amount of time taken for an observation.

For this given problem, which abstract data type (ADT) would be the most suitable?

A. queue

B. graph

C. stack

D. array

Question 6

Sahil is testing his friend Julia's knowledge of recurrence relations. He writes the following recurrence relation for an algorithm.

$$T(n) = 2T\left(\frac{n}{2}\right) + O(1)$$

Which algorithm(s) has Sahil written the recurrence relation for?

A. quicksort only

B. mergesort only

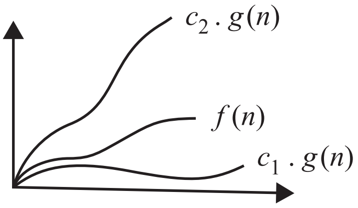
C. both quicksort and mergesort

D. neither quicksort nor mergesort

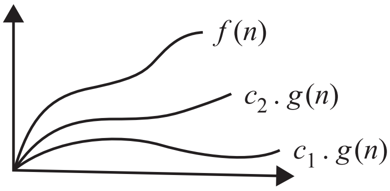
Question 7

Which one of the following graphs best represents Big-Θ when considering extremely large values of *n*?

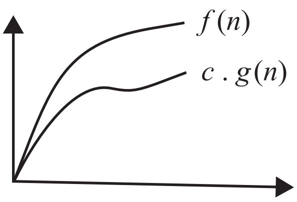
A.



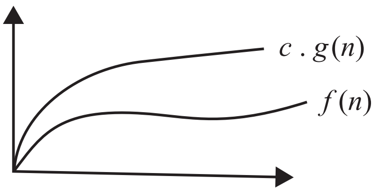
B.



C.



D.



Question 8

A quicksort algorithm with an unknown implementation will be used to sort a large array of elements where the pivot is chosen as the first element in the array.

What property must the input to this array have in order to minimise the chance of reaching the worst case runtime complexity?

A. The input must be random.

B. The input must be pre-sorted.

C. The input must be in ascending order.

D. The input must be in descending order.

Question 9

The main goal of David Hilbert's 1927 program was to

A. prove that a system with a computable set of axioms could never be complete.

B. remove all paradoxes and inconsistencies from the foundations of mathematics.

C. prove that it is not possible to formalise all mathematical statements axiomatically.

D. construct a statement that can be derived from formal axiomatic rules and can be shown to be true.

Question 10

Which one of the following descriptions of the Floyd-Warshall algorithm for transitive closure of a graph having V vertices and E edges is correct?

A. The time complexity of the algorithm is $\Theta(E^3)$.

B. The time complexity of the algorithm is $\Theta(V^2 \log V)$.

C. The algorithm finds the transitive closure of a graph whether it is directed or undirected.

D. The algorithm finds the transitive closure of a directed graph by using the weighted edges as part of constructing the adjacency matrix.

Question 11

Which one of the following descriptions of P, NP and NP-complete problems is **incorrect**?

A. It is a general belief that NP-complete problems are considered to be harder to solve than P problems.

B. Normally, heuristics are applied to solve NP-complete problems. The solution obtained may be exact.

C. If an NP-complete problem can be solved in P time, all NP-complete problems can also be solved in P time. In that case, P = NP.

D. If an NP-complete problem can be solved in P time, all NP-complete problems can also be solved in P time. In that case, it is still not known whether P = NP.

Question 12

Consider the following algorithm.

```
Algorithm myFunction(a)
Begin
    If (a < 1) Then
      Return 2
    Else
       b ← a - 3
       Return a + myFunction(b)
    EndIf
End
```

What is the value at the end of the algorithm if it is run with the input a = 4?

A. 5

B. 6

C. 7

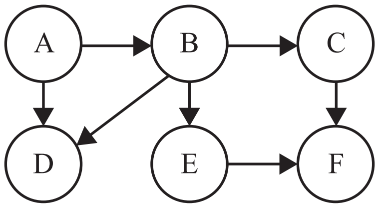
D. 2

Question 13

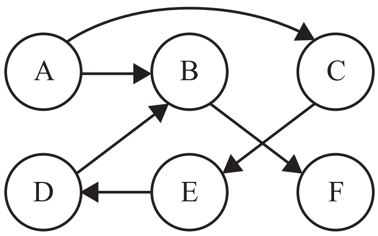
The string A B C F E D is obtained as a breadth-first search through a directed graph, beginning at A.

Which one of the following trees could represent the graph being searched?

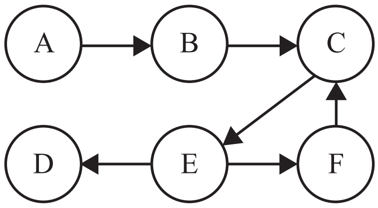
A.



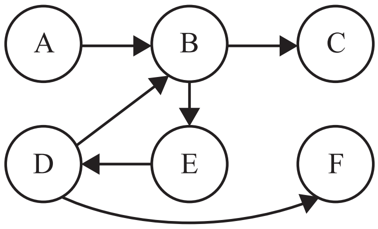
B.



C.



D.



Question 14

(X or Y) and (not (X) or Z) and (not (Z) or Y)

Under which circumstances is the conditional expression above true?

A. when X and Y are true, or X is false and at least one of Y and Z is true

B. when all of X, Y and Z are true, or X is false and Y is true

C. when exactly two of X, Y and Z are true

D. when all of X, Y and Z are false

Question 15

A program with a nested loop iterates n times for the outer loop and $n - 1$ times for the inner loop.

What is the Big-O complexity of this program?

A. $O(n - 1)$

B. $O(n^2)$

C. $O(1)$

D. $O(n)$

Question 16

Sebastian is working with a graph that contains edges with negative weights. He needs to implement an algorithm to calculate the shortest path from a particular node.

Which graph algorithm should Sebastian implement and for what reason?

A. the Bellman-Ford algorithm as Dijkstra's algorithm does not always provide a correct solution

B. Dijkstra's algorithm as the Bellman-Ford algorithm does not always provide a correct solution

C. the Bellman-Ford algorithm due to the negative weights

D. Dijkstra's algorithm due to the negative weights

Question 17

Which of the following properties, where intensification narrows the search to a local region and diversification considers other regions of the search space, is more likely to cause convergence towards global optimality when assessing meta-heuristic algorithms?

A. intensification by itself

B. diversification by itself

C. both intensification and diversification

D. neither intensification nor diversification

Question 18

Which one of the following statements about Big-O, Big-Ω and Big-Θ notation is correct?

A. An algorithm having a best case complexity of $\Omega(n \log n)$ must also have an average case complexity of $\Theta(n \log n)$.

B. An algorithm having a worst case complexity of $\Theta(n \log n)$ will have a worst case complexity of $O(n \log n)$.

C. An algorithm having a worst case complexity of $O(n^2)$ must also have a worst case complexity of $\Theta(n^2)$.

D. An algorithm having a best case complexity of $\Omega(n)$ will also have a worst case complexity of $O(n)$.

Question 19

The following pseudocode for Floyd's all-pair shortest path algorithm is incomplete.

```
Let D be a |V| × |V| array of minimum distances initialised to ∞
For each edge (u,v) Do
      D[u][v] ← w(u,v)  // the edge weight (u,v)
  For each vertex v Do
      D[v][v] ← 0
  EndFor
  For k from 1 to |V| Do
      For i from 1 to |V| Do
         For j from 1 to |V| Do
         // this section is incomplete
         EndFor
      EndFor
  EndFor
EndFor
```

Which one of the following pseudocode extracts will complete the algorithm?

A.
```
If D[i][j] > D[i][k] + D[j][k] Then
      D[i][j] ← D[i][k] + D[k][j]
EndIf
```

B.
```
If D[i][j] < D[i][k] + D[j][k] Then
      D[i][j] ← D[i][k] + D[k][j]
EndIf
```

C.
```
If D[i][j] < D[i][k] + D[k][j] Then
      D[i][j] ← D[i][k] + D[k][j]
EndIf
```

D.
```
If D[i][j] > D[i][k] + D[k][j] Then
      D[i][j] ← D[i][k] + D[k][j]
EndIf
```

Question 20

Which one of the following descriptions of dynamic programming and divide and conquer is correct?

A. Dynamic programming aims to solve the sub-problems once, whether they are overlapping or not, whereas divide and conquer does not care about how many times it needs to solve a sub-problem.

B. Divide and conquer aims to solve the sub-problems once, whether they are overlapping or not, whereas dynamic programming does not care about how many times it needs to solve a sub-problem.

C. Dynamic programming aims to find an optimal solution for a problem by splitting it into non-overlapping sub-problems, finding the optimal solutions for the sub-problems, and combining the optimal solutions for the sub-problems to form the optimal solution for the original problem.

D. Divide and conquer aims to find an optimal solution for a problem by splitting it into overlapping sub-problems, finding the optimal solutions for the sub-problems with the intention of solving the overlapping sub-problems only once, and combining the optimal solutions for the sub-problems to form the optimal solution for the original problem.

END OF SECTION A

SECTION B

Instructions for Section B

Answer **all** questions in the spaces provided.

Use the Master Theorem to solve recurrence relations of the form shown below.

$$T(n) = \begin{cases} aT\left(\dfrac{n}{b}\right) + kn^c & \text{if } n > 1 \\ d & \text{if } n = 1 \end{cases} \qquad \text{where } a > 0,\ b > 1,\ c \geq 0,\ d \geq 0,\ k > 0$$

and its solution

$$T(n) = \begin{cases} O(n^c) & \text{if } \log_b a < c \\ O(n^c \log n) & \text{if } \log_b a = c \\ O(n^{\log_b a}) & \text{if } \log_b a > c \end{cases}$$

Question 1 (3 marks)

Explain, using an example, the role of the tape in a Turing machine.

Question 2 (2 marks)

A mergesort algorithm runs on an initial array input, x, with a best case running time. It returns the sorted array [0, 2, 4, 6, 8, 10, 12, 14].

What is the initial array input x? Explain your answer.

Question 3 (8 marks)

A special calculator is designed to accept an input stream of symbols. The symbols can be either a number or an arithmetic operator such as +, −, × and ÷.

The calculator works in the following way:

- Whenever it encounters a number, the calculator will append the number to a special location designated for storing the numbers and preserve their order from the input.
- Whenever it encounters an arithmetic operator, the calculator will do the following:
  - fetch the last two numbers stored in the special location with *x* being the second-last number and *y* being the last number
  - perform the arithmetic operation with *x* being the first operand and *y* being the second operand
  - put the result back into the special location as the last entry

  For example, if the numbers in the special location are 8, 1, 6, 2 and it encounters a ÷, the numbers in the special location after the operation will be 8, 1, 3 because 6 ÷ 2 = 3 and 3 is put back into the special location as the last entry.
- When the input stream is used up, the calculator will fetch the last number in the special location and then display it as the final result of the calculation.
- Whenever it cannot perform its operations (for example, it cannot fetch two numbers from the special location to perform the arithmetic), the calculator will display 'Error'.

An example of an input stream may be

22	3	−	100	20	×	+

For this example of an input stream, the calculator will display 2019 as the final result.

a.

8	3	+	18	2	÷	×

What will be the final display if the input stream is as shown above? Explain your answer. 3 marks

b. **i.** Select the most appropriate abstract data type (ADT) to model the numbers stored in the special location of the calculator. Explain your answer. 2 marks

ii. Write the ADT specification for the ADT selected in **part b.i.** 3 marks

DO NOT WRITE IN THIS AREA

Question 4 (4 marks)

Ada is currently studying array data structures. She comes up with the following way of comparing two numeric arrays of the same size. A numeric array is one where all of its entries are numbers.

Let A and B be two numeric arrays of size n. The array A is said to be greater than or equal to the array B, denoted as $A \geq B$, if for at least half of the values of i, the condition $A[i] \geq B[i]$ holds where $i = 1, \ldots, n$.

Ada wants to write an algorithm to determine whether A is greater than or equal to B. She has already implemented the following two algorithms:

1. an algorithm called `sortAscending` that will sort a numeric array of size n in ascending order with a worst case time complexity of $O(n^2)$
2. an algorithm called `median` that returns the median value of a numeric array with a time complexity of $O(1)$

Ada writes the following pseudocode to determine whether a numeric array A is greater than or equal to B, both of size n.

```
Algorithm isGreaterOrEqual(A, B, n)
Begin
  A ← sortAscending(A, n)
  B ← sortAscending(B, n)
  mA ← median(A, n)
  mB ← median(B, n)
  If mA >= mB Then
       Return true
  Else
       Return false
  EndIf
End
```

a. What is the worst case time complexity of `isGreaterOrEqual`? Explain your answer. 2 marks

b. Is Ada's pseudocode for `isGreaterOrEqual` correct? Explain your answer using an example. 2 marks

Question 5 (4 marks)

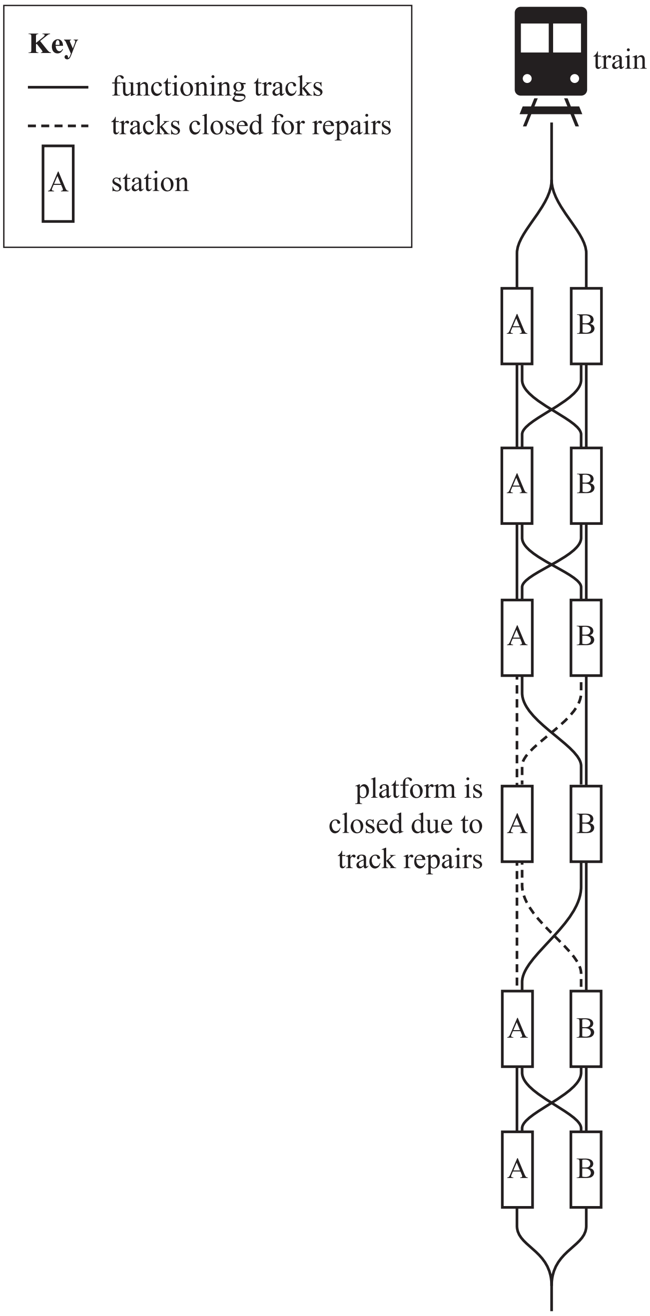
Wei has just finished reading about the Halting Problem and is still confused about what the Halting Problem is.

Write pseudocode to demonstrate the Halting Problem to Wei. Include relevant annotations.

Question 6 (7 marks)

A metropolitan train company has asked Maia to assist with scheduling trains travelling along a train network. Each station along the network has two platforms and interconnecting tracks. The expected wait time at each platform and the time taken to travel along that track depend on the number of staff allocated to assist with boarding and signalling. At times, platforms or tracks may be closed for repairs.

The proposed network is modelled below.



DO NOT WRITE IN THIS AREA

a. One approach to help with scheduling is to use a brute-force algorithm to reduce congestion for each train travelling through the network.

Explain whether or not this is feasible. Include the time complexity in your explanation. 4 marks

b. Maia suggests that a dynamic programming approach should be used for scheduling as the train network is likely to expand.

What properties of this problem make it suitable for a dynamic programming approach? 3 marks

Question 7 (10 marks)

Eric has been employed by a chemistry laboratory to test the purity of a material it manufactures called strongsheet. Strongsheet is a material made of pure carbon, although the laboratory is still perfecting its manufacturing process and there are some impurities present.

The following images have been produced using an electron microscope. Figure 1 is an ideal sample of a strongsheet structure, while Figure 2 has various impurities creating unwanted links in the lattice.

Ideal structure

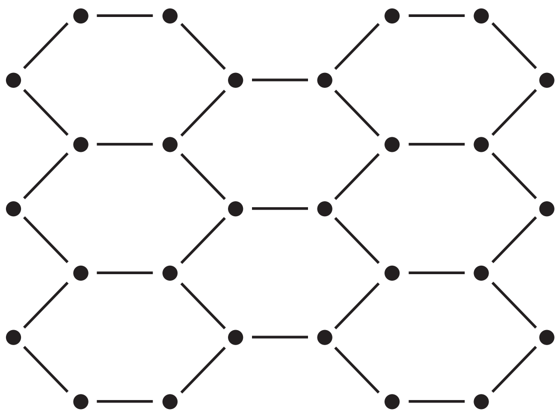


Figure 1

Current structure

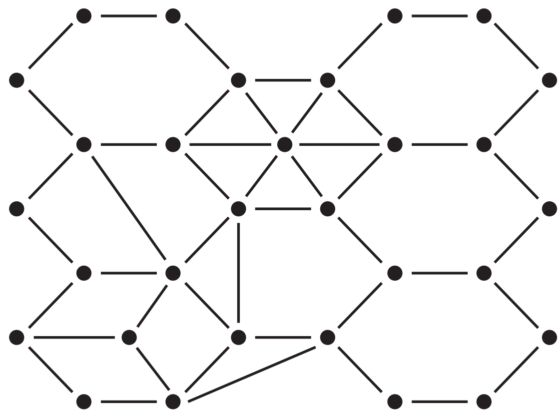


Figure 2

a. Explain how graph colouring can be used to test the purity of the strongsheet sample. 2 marks

b. As the laboratory increases the size of its strongsheets, will it still be able to use graph colouring to test for purity? Explain your answer. 3 marks

c. Write a greedy algorithm that can be used to compute over the selected samples, Figure 1 and Figure 2, shown on page 18. 5 marks

Question 8 (4 marks)

Explain the relationship between Cobham's thesis and the Church-Turing thesis. As part of your explanation, include a definition of both theses.

Question 9 (4 marks)

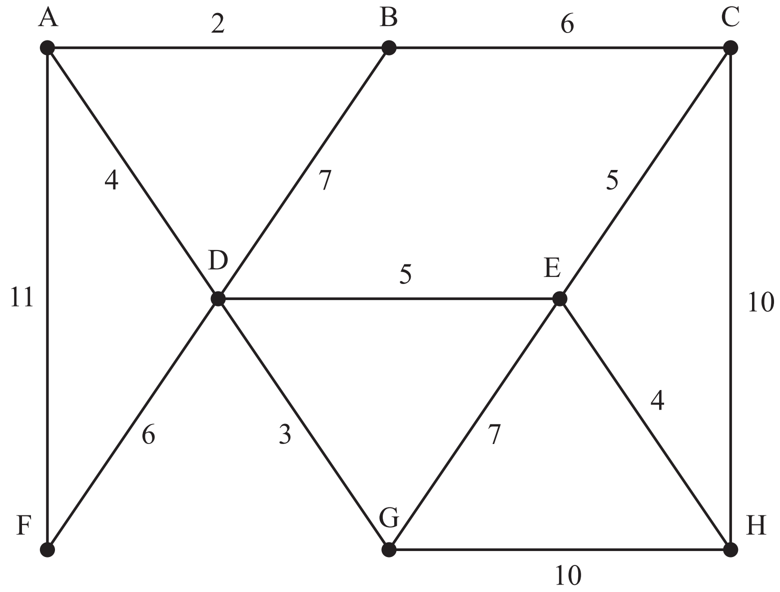
Describe how DNA computing works and explain how it can be used to overcome the current limits of computation.

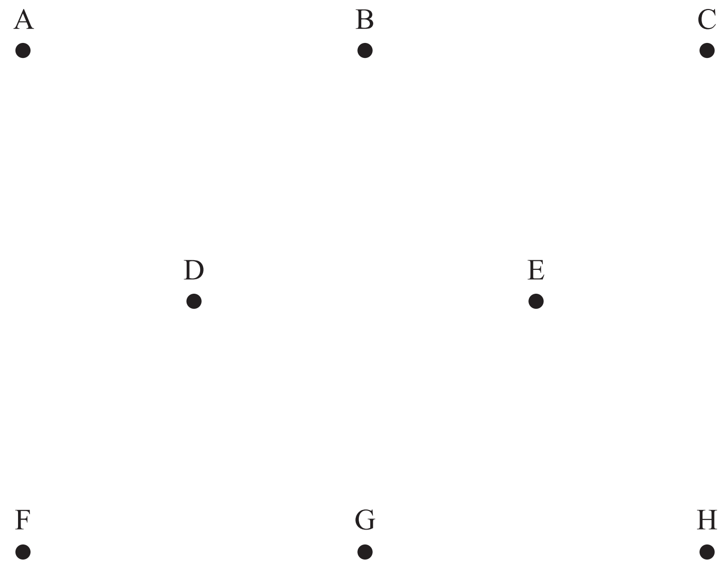
Question 10 (5 marks)

Outline how induction can be used to show that a tree with n vertices has $n - 1$ edges. You may draw and annotate a diagram as part of your answer.

Question 11 (2 marks)

Using Prim's algorithm, find the minimal spanning tree for the weighted graph below, starting from A. Show the order of the edges added to the tree.





Order ____

Question 12 (8 marks)

A cellular automaton is a system in which each row is generated based on the row before it, in particular the cell above, the cell above to the left and the cell above to the right. The rules can vary, but for this question the rule is given as the following.

Rule

1 1 1	1 1 0	1 0 1	1 0 0	0 1 1	0 1 0	0 0 1	0 0 0
0	1	1	1	0	0	0	0

Assume the edges are considered 0, that is, the cells on the edge do not consider the cells on the other edge. For example, given the rule above with a row containing a single 1, the next few rows will be

0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	1

a. Given the input row | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 |, determine the next row. 1 mark

___

b. Draw a decision tree to implement this cellular automata rule. 3 marks

c. Write pseudocode that takes an input array containing a combination of eight 0s and 1s, and generates *n*, the number of rows. Row 0 should contain the input row. 4 marks

Question 13 (4 marks)

Donna has a number of forms, numbered from 1 to 5, that need to be delivered to classrooms at a school. She has made a list of classroom names and, for each classroom, she has created a table of how many of each form need to be delivered. For example

Forms / Classrooms	1	2	3	4	5
1A	0	1	3	2	0
1B	1	4	5	2	1

Donna would like to get these forms delivered in the quickest way possible. At the moment she intends to proceed in classroom order.

Advise Donna on an alternative way of delivering the forms that will be more efficient.

Question 14 (9 marks)

Joe finds it very time-consuming to perform the multiplication of two two-dimensional numeric arrays of size $n \times n$. He asks Alex, Betty and Chloe to help him write a program to perform the multiplication.

Alex first attempts to implement the multiplication according to the following pseudocode.

```
Algorithm multiply(A, B, n)
Begin
    For i = 1 to n Do
        For j = 1 to n Do
            Product[i][j] ← 0
            For k = 1 to n Do
                Product[i][j] ← Product[i][j] + A[i][k] × B[k][j]
            EndFor
        EndFor
    EndFor
    Return Product
End
```

Assume the multiplication and addition of two numbers can be performed in O(1) time.

a. What is the time complexity of Alex's pseudocode? Justify your answer. 2 marks

The pseudocode to add A and B, two $n \times n$ numeric arrays, is given below.

```
Algorithm add(A, B, n)
Begin
    For i = 1 to n Do
        For j = 1 to n Do
            Sum[i][j] ← A[i][j] + B[i][j]
        EndFor
    EndFor
    Return Sum
End
```

Betty comes up with the following recursive method of multiplying the arrays **when *n* is 1 or *n* can be divided by 2**:

- Step 1 – When n is 1, that is $A = A[1][1]$ and $B = B[1][1]$, just multiply the two numbers together to obtain the product, that is $C[1][1] = A[1][1] \times B[1][1]$, and return C.
- Step 2 – Otherwise, do the following:

I. Split each two-dimensional array into four smaller two-dimensional arrays of size $(n/2) \times (n/2)$. Then, the two-dimensional arrays A and B will be denoted as

$$A = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} B_{1,1} & B_{1,2} \\ B_{2,1} & B_{2,2} \end{bmatrix}$$

where $A_{1,1}$, $A_{1,2}$, $A_{2,1}$ and $A_{2,2}$ are the two-dimensional arrays of size $(n/2) \times (n/2)$ split from A, and $B_{1,1}$, $B_{1,2}$, $B_{2,1}$ and $B_{2,2}$ are those split from B.

II. Perform the following multiplications and additions.

$$C_{1,1} = A_{1,1} \times B_{1,1} + A_{1,2} \times B_{2,1}$$
$$C_{1,2} = A_{1,1} \times B_{1,2} + A_{1,2} \times B_{2,2}$$
$$C_{2,1} = A_{2,1} \times B_{1,1} + A_{2,2} \times B_{2,1}$$
$$C_{2,2} = A_{2,1} \times B_{1,2} + A_{2,2} \times B_{2,2}$$

III. Form the resultant two-dimensional array C using the following format and return it.

$$C = \begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \end{bmatrix}$$

b. Assume both the multiplication and addition of two numbers can be performed in O(1) time.

$$T(n) = \begin{cases} 8T\left(\frac{n}{2}\right) + n^2 & \text{if } n > 1 \text{ and } n \text{ is even} \\ 1 & \text{if } n = 1 \end{cases}$$

Explain why the time complexity of Betty's algorithm can be obtained using the recurrence relation above. 3 marks

c. What is the time complexity of Betty's recursive algorithm? Explain your answer. 2 marks

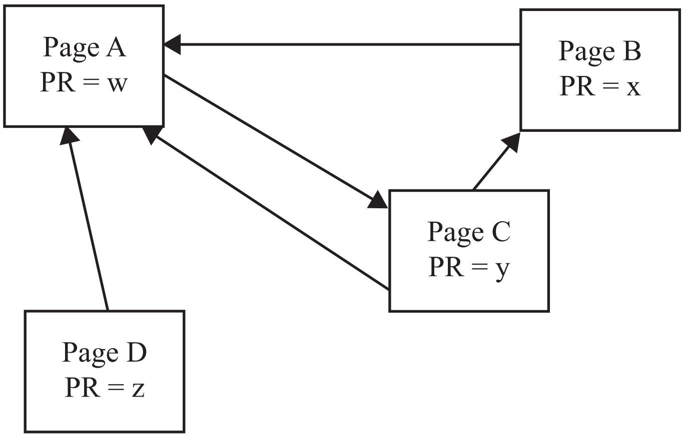
d. Chloe says that she knows another recursive method for the multiplication that gives the following recurrence relationship.

$$T(n) = \begin{cases} 7T\left(\frac{n}{2}\right) + 18\left(\frac{n}{2}\right)^2 & \text{if } n > 1 \\ 1 & \text{if } n = 1 \end{cases}$$

Is this new method faster than the previous two? Justify your answer. 2 marks

Question 15 (2 marks)

The following PageRank (PR) has been calculated using the PageRank algorithm for four web pages. All PR values are greater than zero.



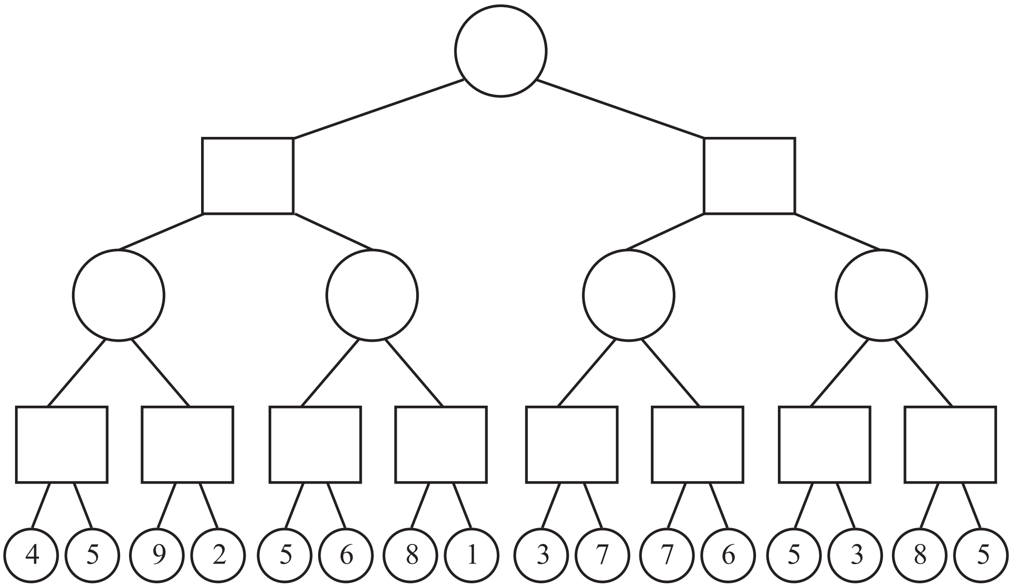
Explain why Page D has a PR value greater than zero despite having no incoming links. Use mathematical calculations as part of your explanation.

Question 16 (4 marks)

Stella and Cameron are playing a turn-based game that allows each player to collect points based on a heuristic value of special cards dealt face-up on the table. At each turn, a player chooses one card from a choice of two cards. The goal of the game is for a player to score the highest number of points.

Below is an incomplete minimax game tree of Stella and Cameron's game, where the circles represent the moves of Stella, the maximising player, and the squares represent the moves of Cameron, the minimising player.

Explain how the minimax algorithm can be used by Stella to give her the best chance of winning the game. Complete the game tree as part of your explanation.



DO NOT WRITE IN THIS AREA

END OF QUESTION AND ANSWER BOOK