SOLUTIONS in red

ALGORITHMICS UNIT 3 & 4

Trial Exam 2: 2019

Reading Time: 15 minutes
Writing time: 120 minutes (2 hours)

QUESTION AND ANSWER BOOK

Section	*Number of questions*	*Number of questions to be answered*	*Number of marks*
A	20	20	20
B	9	9	80

- 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 ?? 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 you name in the space provided to verify this.

- All written responses must be in English, point form is preferred.

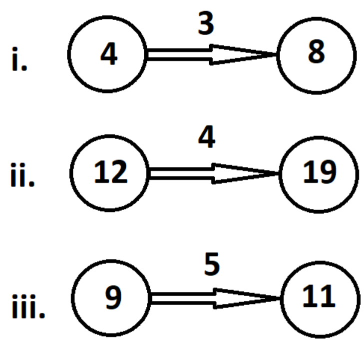
Students are NOT permitted to bring mobile phones and/or any other unauthorised electronic devices into the test room.

The Master Theorem as provided on VCAA exam papers is:	$T(n) = \begin{cases} aT\left(\frac{n}{b}\right) + kn^c, & if\ n > 1 \\ d, & if\ n = 1 \end{cases}$ $where\ a > 0, b > 0, c \geq 0, d \geq 0, k > 0$ and its solution $T(n) = \begin{cases} O(n^c), & if\ \log_b a < c \\ O(n^c \log n), & if\ \log_b a = c \\ O(n^{\log_b a}), & if\ \log_b a > c \end{cases}$
Georgia's note: Recall the log laws, for an alternate representation that shows the growth of the recursion in the layers. Also easier to work out!	$\begin{cases} \log_b a < c \Rightarrow \frac{a}{b^c} < 1 \\ \log_b a = c \Rightarrow \frac{a}{b^c} = 1 \\ \log_b a > c \Rightarrow \frac{a}{b^c} > 1 \end{cases}$

SECTION A – Multiple Choice – select one option only

Question 1

What is the result of relaxing each of the following weighted edges shown at right, where the node labels show the distances calculated so far?



A. 8 becomes 7, 19 becomes 16, 11 becomes 14

B. 8 becomes 8, 19 becomes 16, 11 becomes 14

C. 8 becomes 8, 19 becomes 19, 11 becomes 11

D. 8 becomes 7, 19 becomes 16, 11 becomes 11

Question 2

Will loop infinitely	Pseudocode
(i)	x = 0 while True print x endwhile
(ii)	x = 0 while x < 10 print x endwhile

(iii)	x = 0
	while x < 10
	    print x
	    x = x + 1
	endwhile

(iv)	y = 5 for x = 1 to y print x next

Which of the numbered pseudocode fragments will loop without terminating in reasonable time?

A. (i),(iii)

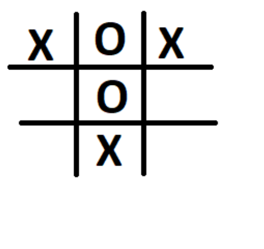
B. (ii),(iv)

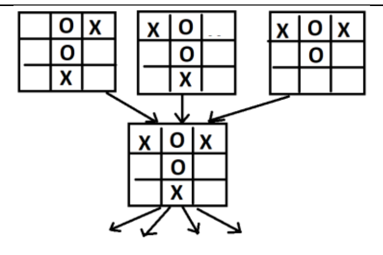
C. (i),(ii)

D. (i),(iv)

Question 3

Consider the directed graph where nodes are reachable tic-tac-toe board positions and edges represent valid moves. What are the in-degree and the out-degree respectively of the following game node when it is O's turn to play?





Answer:
There are three possible edges that could have gone into this node, and O has four spaces to move to.

A. In-degree is 3, out-degree is 4

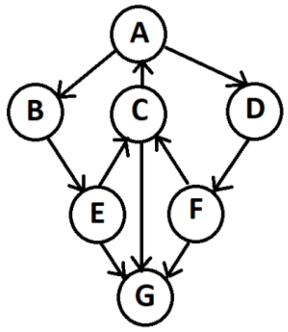
B. In-degree is 4, out-degree is 4

C. In-degree is 6, out-degree is 4

D. In-degree is 9, out-degree is 9

Question 4

If you perform a depth first search on the directed graph shown above, starting at node A and whenever faced with a decision of which node to pick from a set of nodes, choose the one that is earliest in the alphabet, the sequence of node traversal will be:



A. A B E G C D F

B. A B E C G D F

C. A B D E F C G

D. A B D C F E G

Consider the following pseudocode fragment to answer Question 5 and Question 6

```
01 for k = 1 to 10 do
02     for p = 1 to k do
03         print (k + p)
04     end do
05 end do
06 m = 7
07 print m * m
```

Question 5

The first three numbers that will be printed by this pseudocode fragment are:

A. 1,2,49

B. 2,6,8

C. 1,2,3

D. 2,3,4

Question 6

Line 03 is executed

A. 35 times

B. 55 times

C. 10 times

D. 100 times

Question 7

With regard to Artificial Intelligence, which of the following statements is FALSE?

A. In order to manipulate symbols, a computer needs to understand what they mean.

B. Solving a problem involves using actions to transition from a start state to a goal state.

C. Problems are solved through the manipulation of symbols.

D. Artificial Intelligence programs reduce problems to symbols.

Question 8

John Searle created a thought experiment where he sat in a room and was passed messages written in Chinese. He would look up the symbols in a book and write down the corresponding symbols as a reply. To an outside observer it would appear that he understood Chinese. Searle used this thought experiment to illustrate that computers were not intelligent. What was his **main** point?

A. Humans can understand Chinese symbols but computers can't.

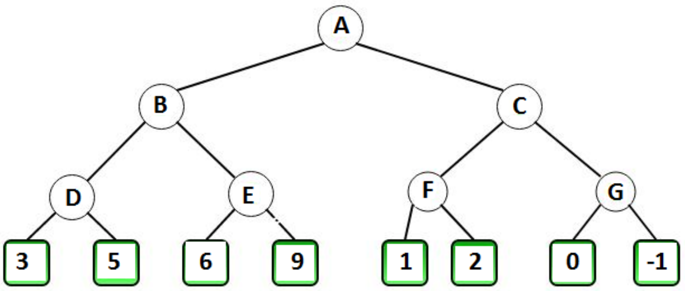
B. Manipulating symbols requires intelligence.

C. The replies were used to determine if Searle was a human or a computer.

D. Computers just manipulate symbols without understanding them.

Question 9

A game tree for the maximising player is shown at point A in the game for a turn taking game. According to the minimax algorithm, which is the most favourable path that the maximising player should choose against an equally capable opponent?



A. A-B-E

B. A-C-F

C. A-B-D

D. A-C-G

Question 10

Consider the following pseudocode snippet

```
y := 0 ;
while y < n do (
     y := y + 1 ;
)
```

The best and worst time complexity respectively is:

A. O(1), O(n)

B. Ω(1), O(n)

C. Ω(n), O(n)

D. Ω(n), O(1)

Question 11

Consider the following pseudocode snippet

```
y := 1 ;
while y < n do (
     x := n ;
     while x > 0 do (
         x := x - y ;
     );
     y := y + 1 ;
)
```

The number of actions performed in the worst case as a function of n, is most closely represented by:

A. $\frac{n(n+1)}{2}$

B. $\frac{n(n+1)(2n-1)}{6}$

C. $\frac{n}{1!}+\frac{n}{2!}+\frac{n}{3!}+\frac{n}{4!}+\frac{n}{5!}+.......+\frac{n}{n!}$

D. $n+\frac{n}{2}+\frac{n}{3}+\frac{n}{4}+\frac{n}{5}+.......+1$ (correct answer)

Question 12

A Turing machine with 6 states is defined as follows:

State A				State B			
Read	Write	Move	State	Read	Write	Move	State
0	0	R	A	0	0	R	C
1	0	R	B	1	1	R	B

State C				State D			
Read	Write	Move	State	Read	Write	Move	State
0	0	R	C	0	0	L	D
1	0	R	F	1	1	L	E

State E				State F			
Read	Write	Move	State	Read	Write	Move	State
0	0	R	A	0	0	Halt	
1	1	L	E	1	1	L	D

For the following tape in state A, at the position indicated by the arrow.

								State
0	1	1	1	0	1	1	0	A

After 4 operations of the TM, the symbols on the tape and current state will be:

A.

0	0	1	0	0	1	1	0	C

B.

0	0	1	1	1	1	1	0	B

C.

0	0	1	1	0	1	1	0	B

D.

0	0	1	1	0	0	1	0	C

(D is correct)

								State
0	1	1	1	0	1	1	0	A
0	0	1						B
0	0	1	1					B
0	0	1	1	0				B
0	0	1	1	0	0			C

Justification of D

Question 16

The recurrence relation $T(n) = 3T\left(\frac{n}{3}\right) + \sqrt{n}, where\ T(1) = c$ representing the time complexity of an algorithm, where c is a constant, is equivalent to:

A. $T(n) = O\left(\sqrt{n}\right)$

B. $T(n) = O\left(n^3\right)$

C. $T(n) = O(n)$ (is correct, *a=3,b=3,c=0.5* using the Master Theorem supplied by VCAA)

D. $T(n) = O(1)$

Question 17

Using dynamic programming, the output to stage 9 of a coin change problem to find the minimum number of coins to be given as change where the coins denominations are coins={1^c,3^c,4^c} and the target change is 10 cents, is formed by:

A. Using stage n-1 of the calculations.

B. Using stage 8 of the calculations.

C. Using stage 8,6,5 of the calculations.

D. Using stage 9,8,7 of the calculations.

Question 18

Consider recursive function `geo` defined as follows (for positive integers input only):

```
Function geo(n)
// Input n is a natural number
If (n≤1)
    Return n
Else
    Return geo(n-1)+2n
Endif

End function
```

According to this function, if geo is called with `geo(3)` the value returned by the function will be:

A. 8

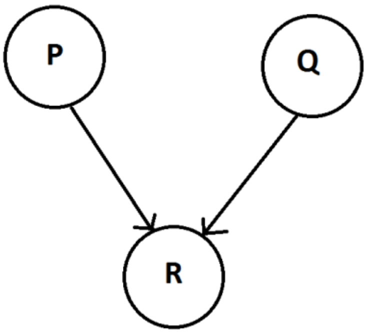
B. 9

C. 10

D. 11

Question 19

Suppose that P, Q, and R are different web pages in a system that are connected by directed edges as shown in the diagram.



Which possible Page Rank web page calculations matches the outcome **before** and **after** a directed link from P to Q is added?

A. Before: $\Pr(R_{i+1}) = \Pr(Q_i) + \Pr(P_i)$ After: $\Pr(R_{i+1}) = \Pr(Q_i) + \frac{1}{2}\Pr(P_i)$

B. Before: $\Pr(R_{i+1}) = \Pr(Q_i) + \Pr(P_i) + \frac{1}{3}\Pr(R_i)$ After: $\Pr(R_{i+1}) = \Pr(Q_i) + \frac{1}{2}\Pr(P_i) + \frac{1}{3}\Pr(R_i)$

C. Before: $\Pr(R_{i+1}) = \Pr(Q_i) + \frac{1}{2}\Pr(P_i) + \Pr(R_i)$ After: $\Pr(R_{i+1}) = \Pr(Q_i) + \frac{1}{3}\Pr(P_i)$

D. Before: $\Pr(R_{i+1}) = \Pr(Q_i) + \frac{1}{2}\Pr(P_i) + \frac{1}{3}\Pr(R_i)$ After: $\Pr(R_{i+1}) = \Pr(Q_i) + \Pr(P_i) + \frac{1}{2}\Pr(R_i)$

Question 20

'Push' and 'Pop' are appropriate operations for which data structure?

A. a list

B. a queue

C. a stack

D. an array

SECTION B – Extended Response Questions Answer all questions in the space provided.

Question 1 (7 marks)

Let F_1, F_2, ….F_n be the size of n computer files that need to be stored on a disk with capacity of D gigabytes. We cannot store all these files in this case as $\sum_{i=1}^{n} F_i > D$.

a. Does a greedy algorithm that selects files in order of non-decreasing F_i size maximise the possible **number of files** held on the disk? Justify your answer with a proof or counter example. (2 marks)

Yes a greedy approach will maximise the possible number of files it will as pack as many small files as possible onto the disk and we select sizes in ascending order until no more will fit onto D.

By contradiction say we can take two files Fj and Fk instead of Fi where sizes Fi<Fj<Fk, this will only make sense if Fk+Fj < Fi+Fj which means that Fk < Fi which isn't true.

b. Does a greedy algorithm that selects files in order of non-decreasing file size F_i, use as much of the capacity of the disk as possible? Justify your answer with a proof or counter example. (2 marks)

This is not possible to solve optimally with a greedy approach.

By counter example suppose that D=10 and the files sizes are {2,3,4,5} if you use a greedy approach we will select (2+3+4=9), where there is a better solution (2+3+5=10) that will be missed.

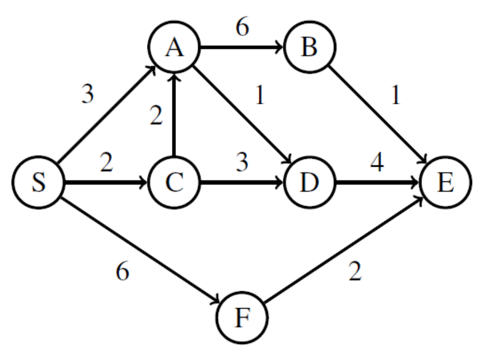
c. With reference to Cobham's thesis and the Church Turing thesis, what is the time complexity and computability of an approach to finding the **optimal** solution for parts a) and b)? (3 marks)

Part a) can be solved in polynomial time as the file sizes are sorted ascending, this is a tractable problem according to Cobham's thesis and belongs to class P as it can be solved in polynomial time on a Turing Machine satisfying the Church Turing thesis.

Part b) is similar to the knapsack problem where there is a combinatorial explosion in the selections That can be made to maximise the usage of available disk space which is equivalent to minimising the Unused space, there is an exponential growth in the decision tree for fitting files, making this problem Non-polynomial and intractable according to Cobham's thesis. Since the optimal solution can neither be found or verified in polynomial time on a turing machine for part b), this problem is NP-Hard according to the Church-Turing thesis of computability.

Question 2 (6 marks)

Consider the following directed and weighted graph.

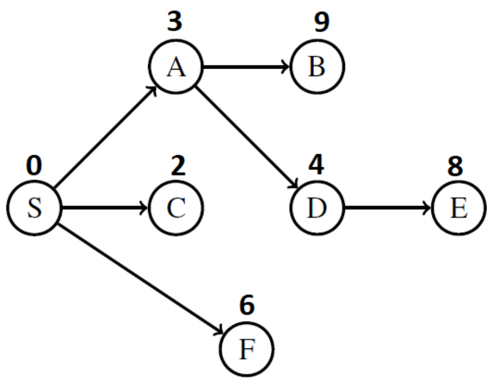


S, A, B, C, D, E, F	nodes
0, ∞, ∞, ∞, ∞, ∞, ∞	initialise distances
0, 3, ∞, 2, ∞, ∞, 6	update S outgoing edges
0, 3, ∞, 2, 5, ∞, 6	relax C outgoing edges
0, 3, 9, 2, 4, ∞, 6	relax A outgoing edges
0, 3, 9, 2, 4, 8, 6	relax D outgoing edges
0, 3, 9, 2, 4, 8, 6	relax F outgoing edges
0, 3, 9, 2, 4, 8, 6	relax E outgoing edges
0, 3, 9, 2, 4, 8, 6	relax B outgoing edges

a. Show the distance information in each main step of the execution of Dijkstra's algorithm starting at node S, and state the order that the nodes are removed from the priority queue. (2 marks)

Dijkstra will visit the vertices in the order S, C, A, D, F. E, B

b. Show the resulting shortest path tree starting at node S with distances annotated on the nodes. (2 marks)



c. If the edge C-D is reversed on the original graph, and its weight multiplied by negative 1, explain the properties of the updated graph and discuss how or if Dijkstra's can be implemented. (2 marks)

Dijkstra's is not recommended for graphs that have negative edges, a cycle is formed A-D-C-A with a net sum of edges equal to 0.

If nodes are only processed once, Dijkstra's can be run on the resulting graph, although due to the cycle formed the distance of node C is updated from (2,S) to (1,D), causing no further change to A, and in this case Dijkstra's will work for the remainder of the nodes as before.

Question 3 (11 marks)

Suppose that you are given 3 words in variables X, Y and Z, respectively, where length(X)=n, length(Y)=m and length(Z)=n+m. The word Z is said to be a shuffle of the letters in words X and Y formed by interleaving the letters from X and Y in such a way that the left-to-right ordering is maintained.

a. If X=chocolate and Y=chips show that Z=cchocohilaptes is a shuffle of X and Y, but chocochilatspe is not. (2 marks)

Z=cchocohilaptes => C choco HI la P te S can be split into CHIPS and chocolate going forward or
going backwards. Z=chocochilatspe =>The only s and p in the shuffle is in CHIPS
and this Z has them in the wrong order, so this shuffle doesn't follow the restriction.

Here is an algorithm to work out if Z is a shuffle of X and Y.

```
Function IsShuffle (X, Y, Z)
// Inputs words X, Y, Z, treated as list of letters
// Outputs Boolean TRUE/FALSE if shuffle exists
If length(Z) is not equal to (length(X)+length(Y)) then
          Return FALSE
Else
     For i=1 to length(Z) do
        If ith element of Z is not in X or Y then
              Return FALSE
        End If
     End do
End if

If ((length(Z)=0) and length(X)=0 and length(Y)=0 then
         Return TRUE
Else If (1st element of Z equals 1st element of X) then
         Remove first element of X
         Remove first element of Z
         isShuffle(X,Y,Z)
Else if (1st element of Z equals 1st element of Y) then
         Remove first element of Y
         Remove first element of Z
         isShuffle(X,Y,Z)
Else
         Return FALSE
End if
```

To call the function `IsShuffle` we prepare by

Setting the global variables

```
Create list X
Set X=[c|h|o|c|o|l|a|t|e]
Create list Y
Set Y=[c|h|i|p|s]
Create list Z
Set Z=[c|c|h|o|c|o|h|i|l|a|p|t|e|s]
```

Then the algorithm is called

```
Print IsShuffle(X,Y,Z)
```

a. What are the main design patterns used in the algorithm? Justify your response using the modules defined and identify the lines of pseudocode for each design pattern described. (2 marks)

Brute force is used to check that all elements of Z exist in X and Y, first if and first for loop.
Decrease and conquer recursion is performed in the second if condition as each match of the current
front of Z is found in X or in Y in the correct sequence, and removing the matching elements before
calling the function again recursively with a smaller input.

Question 3 (continued)

b. Prove by induction that IsShuffle will return the correct answer for any three words X, Y, Z where each variable is input as a list of letters.. (3 marks)

IsShuffle checks the length(Z)=length(X)+length(Y) and correctly exit with FALSE if lengths in error.

If X=[x1,x2,x3,….xn] and Y=[y1,y2,y3,…..ym] and Z=[z1,z2,z3,….z(m+n)] the IsShuffle function

will check all elements of Z are either in X or in Y and correctly exit with FALSE if element not found.

By induction we can show the decrease and conquer method of the algorithm works correctly.

Base case if X=[x_1] and Y=[y_1] and Z=[y_1,x_1], IsShuffle is called IsShuffle([x_1],[y_1],[y_1,x_1])

then statement If 1^{st} element Z is not 1^{st} element X or 1^{st} Y then correctly return FALSE and exit.

If 1^{st} Z= 1^{st} of X or 1^{st} Y will be true, these elements removed and IsShuffle([x_1],[],[x_1]) will be called

this call the 1^{st} Z= 1^{st} of X or 1^{st} of Y will be true, these elements removed and IsShuffle([],[],[]) will

be called, this will reach the if (length(Z)=0 and length(X)=0 and length(Y)=0) statement and correctly

return TRUE for a correct shuffle. Assume that this works for checking k elements correctly then we

show that checking the (k+1)th item is also correct as the lists are always reduced when matches are

correctly X=[xk,x(k+1)] and Y=[y(k+1),y(k+2)] and Z=[y(k+1),xk,y(k+2),x(k+1)] will be correctly

detected as matching in the correct order. And the case of incorrect shuffle order will be detected

X=[xk,x(k+1)] and Y=[y(k+2),y(k+1)] and Z=[y(k+1),xk,y(k+2),x(k+1)] correctly as FALSE.

c. What is the time complexity of IsShuffle for the best and worst case? Justify your response. (2 marks)

The best case is $\Omega(1)$ and will occur if lengths are inconsistent or first letters of X or Y are not the

first letter of Z, resulting in constant time complexity.

The worst case is O(m+n) and this will occur as a match is found in X or Y in sequence in Z for the

entire length of Z which is (m+n), resulting in linear time complexity.

d. Is it possible to improve the efficiency of IsShuffle. Describe how this could be done and outline the new time complexity for the improvement. (2 marks)

The first part of the function that checks lengths and whether symbols in Z match those in X or Y need

only be done once and could be a separate function that is called once only to determine if IsShuffle

should be run. Removing this part of the algorithm will not change the overall time complexity of the

solution to the problem.

Question 4 (8 marks)

a. Write a function called mapSquares in structured pseudocode that takes a list of integers called keys as a parameter and creates and fills a dictionary mapping each of the elements in keys to their squares. For example, if 5 is an element of keys, the dictionary returned by the function would contain a mapping from the key 5 to the value 25. (3 marks)

```
Function mapSquares(keys)
CreateDictionary(d_square)
For each k in keys do
        insertDictionary(d_square, key=k, value=k*k)
End do
Return d_square
End Function
```

b. Write an algorithm FindAnySymbol in structured pseudocode to find the first position in an array of symbols stored in variable **s** that holds any symbol from another array called **d**, or -1 if there is no such position. For example, if **s=[M|o|n|d|a|y]** and **d=[a|e|i|o|u]**, it will find the first vowel in **s** which is at position 2, if **s=[t|i|g|e|r]** and **d=[z|o|o]**, it will return -1 since none of the letters in **d** are in **s.**(5 marks)

```
Algorithm FindAnySymbol (s, d)
// Array s, d
Set position:=MAXVALUE
For i=1 to length of s do
    For j=1 to length of d do
        If (s[i] equals d[j]) then
            // found matching symbols
            If (i < position) then
                position:=i
            End if
        End if
    End do
End do
If (position < MAXVALUE) then
        return position
else
        return -1
End if
End Algorithm
```

Question 5 (10 marks)

a. Show the steps that the simplest Mergesort algorithm would take to put the following list of codes into ascending alphabetical order (from A to Z). P , B , F , J , T , A , Z , H (2 marks)

Split to base case single symbol, merge in order at each level.	
1. [P, B, F, J] [T, A, Z, H]	5. [B, F, J, P] [A, H, T, Z] merges
2. [P, B] [F, J] [T, A] [Z, H]	6. [A, B, F, J, P, T, Z] sorted
3. [P] [B] [F] [J] [T] [A]..[Z]..[H] base case	
4. [B, P] [F, J] [A, T] [H, Z] merges	

b. Describe the design pattern used by the Mergesort algorithm. Explain how it works and situations where it is appropriate to use this design pattern to make algorithms more efficient. (3 marks)

Divide and Conquer design pattern is used by Mergesort.
Works by recursively breaking down a problem into independent sub-problems of similar type,
until the base case is reached which is a simple problem that is easy to solve, solutions to the
Sub-problems are combined until the overall solution is found. Divide and Conquer is useful to apply
to any problem that can be broken down into non-overlapping sub-problems, such as sorting or
closest points, or multiplication.

c. In general how is the time complexity worked out for this type of design pattern? Describe the mathematical constructions used? Demonstrate the technique for working out the time complexity of the simplest Mergesort implementation. (3 marks)

The master theorem can be used to find the time complexity for a Divide & Conquer algorithm.
To use the master theorem you need to work out the recurrence relation for the Divide & Conquer
How many recursive calls, how many data splits and how much work is done as a function of the input
Size n outside of the recursion. $T(n)=aT(n/b)+n^k$, a = number of recursive calls, b = data splits,
And k is the order of the polynomial measuring actions done outside of the recursion.
For Mergesort $T(n)=2T(n/2)+n$, $T(1)=O(1)$, indicates a=2 recursive calls, b=2 data split into two
and linear amount of work done $(k=1)$ at each recursive level to merge the data.

Question 6 (10 marks)
Happytown are holding an election with three candidates (A, B and C). The voting system is preferential and voters must mark each candidate in order of preference 1, 2, 3.

a. Using the graph abstract data type as a main building block, describe a model composed of abstract data types that can hold the information for the following:
- keeps track of how many first, second, and third preferences are given to each candidate
- keeps track for each first preference candidate where the second and third preferences have been allocated
- keeps track for each second preference candidate where the third preference has been allocated

(3 marks)

A directed graph is created where each candidate is represented by a node.
An attribute is added to each node to indicate how many first, second and third preferences they have received. The total can be used to scale the size of the node to show relative popularity.
A directed edge going out of each candidate node to second and third preferenced nodes can be created.
Each edge can have extra attributes showing the count of second and the count of third preferences going in that direction from the first and second preference candidates. The thickness of the edge Can be used to visually represent the allocation of preferences.

b. Show the main operations required to construct and maintain the model you have described in part a. for holding the information for this problem. (3 marks)

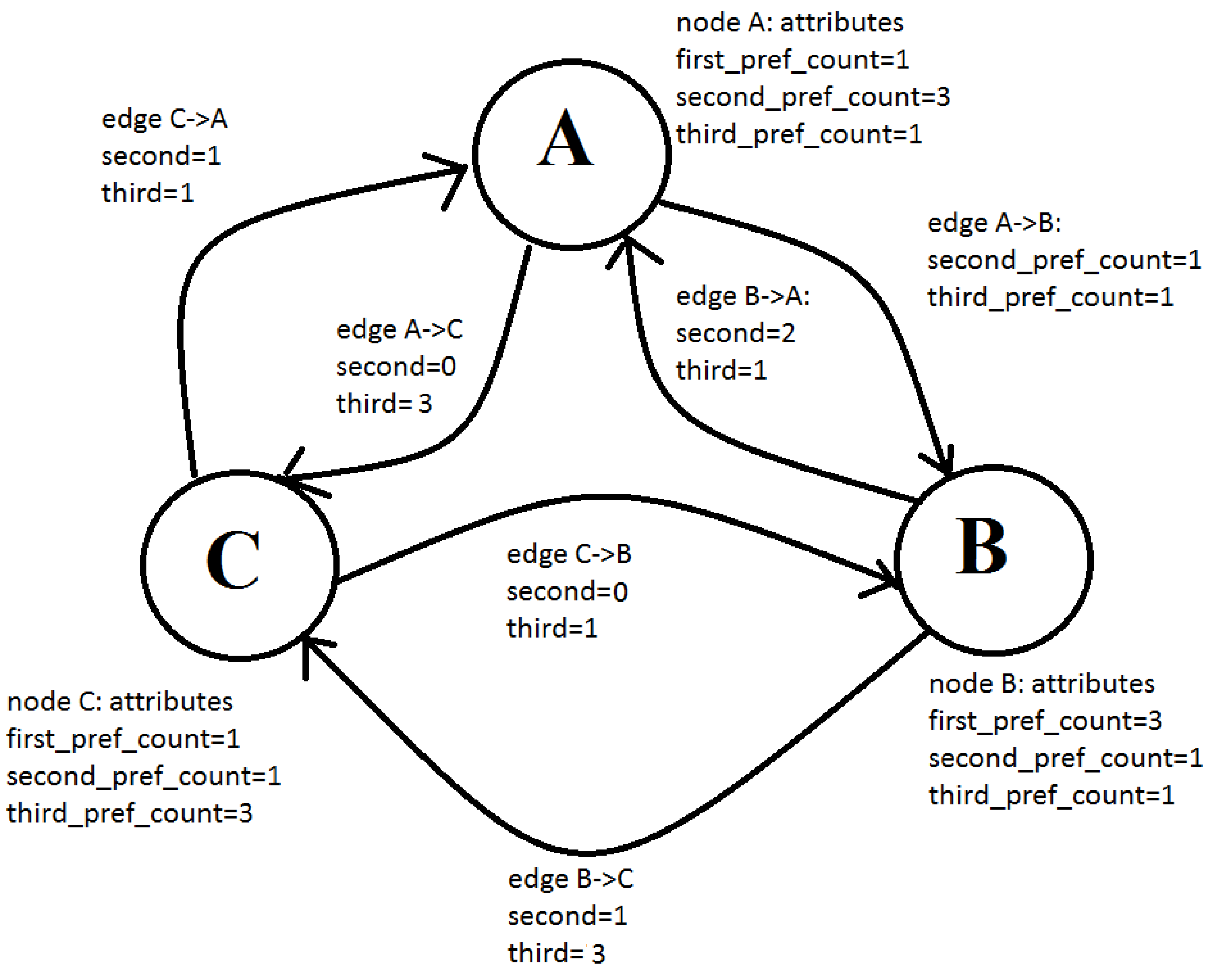
Create directed graph G (V=candidates, E=preferences)
Create a node for each candidate in the election (A, B, C)
Add attribute to each node of first_pref_count, second_pref_count, third_pref_count
Add attribute to edge of second_pref_count, third_pref_count
As each vote is processed, update the count for the first preferenced candidate
For the second preference, if needed create an edge from first candidate to second preference
Update the outgoing edge and the incident node with the second_pref_count.
For the third preference, if needed create an edge from first candidate to third preference
Update the outgoing edge and the incident with the third_pref_count.

Question 6 (continued)

The first 5 voting papers counted for this election are as follows:

Paper 1	Paper 2	Paper 3	Paper 4	Paper 5
1. A 2. B 3. C	1. B 2. C 3. A	1. B 2. A 3. C	1. B 2. A 3. C	1. C 2. A 3. B

c. Show how the model you have defined in parts a. and b. will hold this information, by using an annotated diagram or similar. (4 marks)



Question 7 (10 marks)

a. Briefly explain the following concepts and give an example problem to which it relates to:

- uncomputability
- undecidability (2 marks)

Uncomputability indicates there is no known algorithm to solve this problem, an example is the
Busy beaver problem. The decimal places of π are not countable or computable.
Undecidability indicates the decision problem requiring a yes or no answer that is uncomputable,
an example is the halting problem.

b. What is the Church-Turing Thesis? Provide an argument that testing whether a number is a prime number is a decidable or undecidable function. (3 marks)

The Church-Turing thesis defines computability as a problem that has an algorithm to solve it that
can be run on a Turing Machine in finite time with unlimited resources.
Testing for a prime number n is a decidable function as we can progressively see if we can divide the
number using integer factors from 2 until k, where n/k >1 and check whether any remainders exist.
If the remainder is zero then the number is prime, therefore this is a decidable problem and function,
since there is a known algorithm to solve this problem in finite time.

In Mathematics the theory of addition of natural numbers is complete and decidable (Presburger 1929), as is the theory of the multiplication of positive integers (Skolem 1930).
These proofs of completeness and decidability for addition and multiplication of natural numbers resulted from Hilbert's program of formalisation of Mathematics.

c. Explain what Hilbert's program was trying to achieve? What was the motivation for the program? Explain the limitations and implications for Mathematics and Computer Science, that were subsequently realised or discovered. (3 marks)

Hilbert's program was intended to be a complete, consistent and finitary representation of Mathematics
based on axioms and proven theorems.
Challenges posed by non-Euclidean geometry and the use of algebra and other methods to prove
inconsistent theories was the catalyst for this attempt at formalism.
Godel's incompleteness theorem demonstrated that it was not possible to prove the consistency of all
of number theory of Mathematics by using itself, thereby demonstrating that it is not possible for
Mathematics to be complete and decidable, as also demonstrated by Turing's Halting problem.

Question 8 (8 marks)

a. . Describe the Travelling Salesman Problem (TSP) and explain, informally, why it is classed as a very hard, indeed, intractable, problem. (2 marks)

A salesperston travels between N cities, each city is visited once only during the trip and he/she finishes
where they began. Each link/edge connected two cities has a cost, which must be kept at a minimum for
the trip. This problem is classified as intractable, hard and NP-Hard since it can be neither solved or
have candidate solutions verified in polynomial time.

b. Explain what is meant by the term problem size. What is used as the measure of problem size for the TSP? (2 marks)

Since solutions to the TSP are Hamiltonian paths of the lowest weight they are dependent on the size
of the graph and weighted edges modelling the cities and cost for roads respectively.
The input to this problem is the number of cities $\|V\|$ as nodes and the number of roads $\|E\|$ as edges.
The higher the degree of each node/city the more options are available for traversal from this city to the
next city.

c. Explain a "brute force" method for obtaining a solution to the problem, then give and informally justify its worst case time performance measure in terms of the problem size (given in answer to part 8b above)? (2 marks)

To search for all the Hamiltonian paths by brute force requires checking a huge number of paths,
In the worst case the road network is a complete graph there will be $\|V\|$ nodes and $\|V\|\|V-1\|/2$ edges.
If $N=\|V\|$ for the worst case there will be $(N-1)!/2$ possibilities to check.
Once all Hamiltonian paths are found they can be sorted to find the cheapest one to answer the TSP.

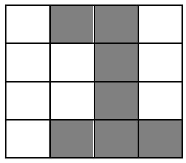
d. A particular computer, can process 1000 tours a second. Give an estimate, with a justification for the time taken (clearly stating your units of time measure) to solve by your brute force method, a TSP with problem size 21. (You may assume that 20! is approximately 2.43×10^{18}.)? (2 marks)

21 cities in a complete graph network is $(21-1)!/2$ distinct possible paths to check
$(2.43\times10^{18})/2=1.215\times10^{18}$ tours, at 1000 tours per second to check requires 1.215×10^{15} seconds
(Just out of interest only: There are 31,566,952 seconds in a year (approximately 3.157×10^{7}), which means that on this computer, this problem takes 0.38×10^{8} (380,000,000) years to solve, intractable is an understatement.)

Question 9 (10 marks)

Consider the case of recognising text and the use of Neural Networks.

A digital image of the number "1" represented by 16 shaded cells on a 4x4 matrix.

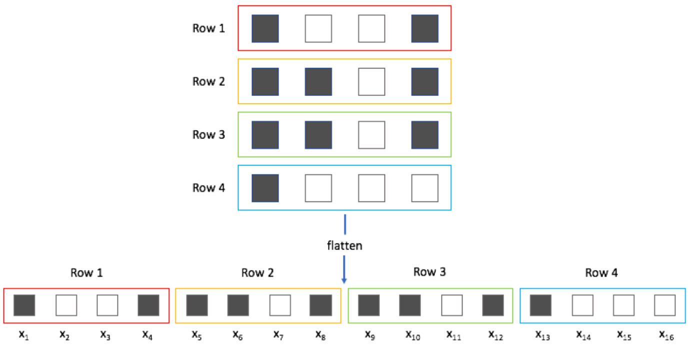


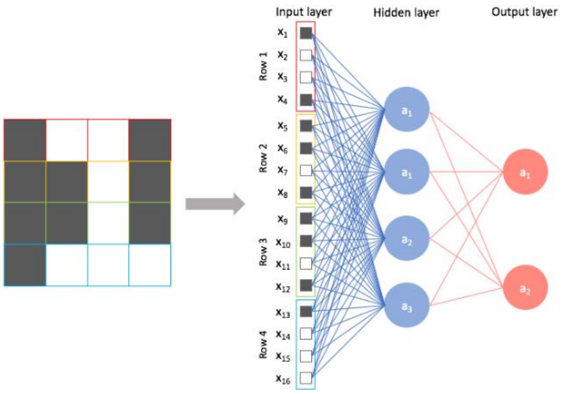
a. Describe in detail a method that can be used to input a digital image, like the one shown into the Neural Network based on the 4x4 matrix of 16 shaded cells. (1 mark)

Each cell of the image input can correspond to an input to the input layer of the Neural Network.

There would be 16 separate inputs, say x_1 to x_{16} representing each cell.

b. Show a pictorial representation of part b) of your answer. (2 marks)





c. After the input step has completed, describe briefly the next 3 main stages of processing performed by the Neural Network. (3 marks)

Each of the inputs is weighted and fed forward to a layer of neurons called the hidden layer where a mathematical function is applied to the weighted inputs from the input layer of neurons.

From the hidden layer values are sent to an output layer representing the result of the recognition.

As the Neural network will need to be trained to recognise text, during the training phase the output is Known and errors in detection are propagated back through the network from the errors to adjust the Weightings and functions on the hidden layers.

Question 9 (continued)

	A neural network has been set up that recognises handwriting and detects mistakes in spelling and grammar, as well as making suggestions about how to restructure sentences to improve clarity and style. This is every English teacher's dream neural network.

After trialling this neural network for several months it appears so intelligent and so effective that Ms. Took, a very new English teacher starts using it to "read" and grade her students' essay submissions.

d. Using the **Robot** and **Brain Simulator** standard responses for and against the Chinese Room argument outline an argument for Ms. Took to either continue, or stop using this neural network to help her grade her students. (4 marks)

The Robot argument is the view that given sensory information, words can be connected to meaning. While a good Neural network will find a mathematical correlation between inputs and outputs, one can argue that nothing is actually understood from the written essays. The NN has worked well in detecting spelling and errors in grammar but will not be able to assess the ideas expressed by the student and be able to determine any relevance to the essay topic. The NN has no understanding of the relevance or intentions, it is merely manipulating symbols. Cognition is not merely symbol manipulation.

The Brain simulator argument argues that the if the NN simulated the actual firing of neurons of a regular English teacher such as Ms. Took, we could actually say that the NN understands the stories. Searle argues that the problem with the Brain Simulator argument is that it is a simulation and is insufficient for producing meaning and mental states as it is not conscious, lacks awareness, and has no conscience, lacks sense of wrong and right.

END OF TRIAL EXAM 2