The Knowledge Hub Universities

partnered with Coventry University

KH5002CEM-Theory of Computation

Mock-up Exam

School of Computing

Question 1

a. Find the recurrence relation for the number of basic operations for the following algorithm, then solve it using backward substitution.

ALGORITHM *A(n)*
//Input: A positive integer *n*
for (int n = 0, n < 100; n++)
y = y / 3;
if *n* = 0 **return** 1
else return *A(n* − 1*)* + (*n* / 2)

T(n) = T(n-1) + 101 T(0) = 100
T(n) = T(n-2) + 101 + 101
T(n) = T(n-3) + 101 + 101 + 101
T(n) = T(n-n) + n*(101)
T(n) = 100 + n*(101)

O(n)

b. Find the order of growth of the following algorithm that is described by the following recurrence relation:

T(n) = 4T(n/3) + 4, T(1) = 1

Let $n = 3^k$
$T(3^k) = 4T(3^k /3) + 4$, $T(3^0) = 1$
$T(3^k) = 4T(3^{k-1}) + 4$
$T(3^k) = 4(T(3^{k-2})+4) + 4$
$T(3^k) = 4(4T(3^{k-2})+4) + 4$
$T(3^k) = 4(4T(3^{k-2}))+4.4 + 4$
$T(3^k) = 4(4(4T(3^{k-3})+ 4)))+4.4 + 4$
$T(3^k) = 4(4(4T(3^{k-3}))))+ 4.4.4 +4.4 + 4$
$T(3^k) = 4^kT(3^{k-k})+ 4^{k-1}.4 +4^{k-2}.4 \ \dots + 4^0.4$
$T(3^k) = 4^k(1)+ (4^k +4^{k-1} \ \dots + 4^1+ 4^0- 4^0)$
$T(3^k) = 4^k+ (4^k +4^{k-1} \ \dots + 4^1+ 4^0- 4^0)$
$T(3^k) = 4^k+ (4^{k-1} -1)/(4-1) - 1$
$O(4^k+ (4^{k-1} -1)/(4-1) - 1)$
$O(4^k)$
$O(4^{\log_3 n})$

c. For the following pair of functions, indicate whether the first function has a lower, same, or higher order of growth than the second function.

$log_{10}n^2$ and $log_5 3n$

$$\lim_{n\to\infty} \frac{log_{10}n^2}{log_5 3n}$$
$$\lim_{n\to\infty} \frac{3.\,log_{10}n}{log_5 3 +\ log_5 n}$$
$$\lim_{n\to\infty} \frac{3.\,log_{10}n}{log_5 3 +\ log_{10}n/log_{10}5}$$
$$3.\,log_{10}5$$

Hence, both functions have the same order of growth.

Question 2

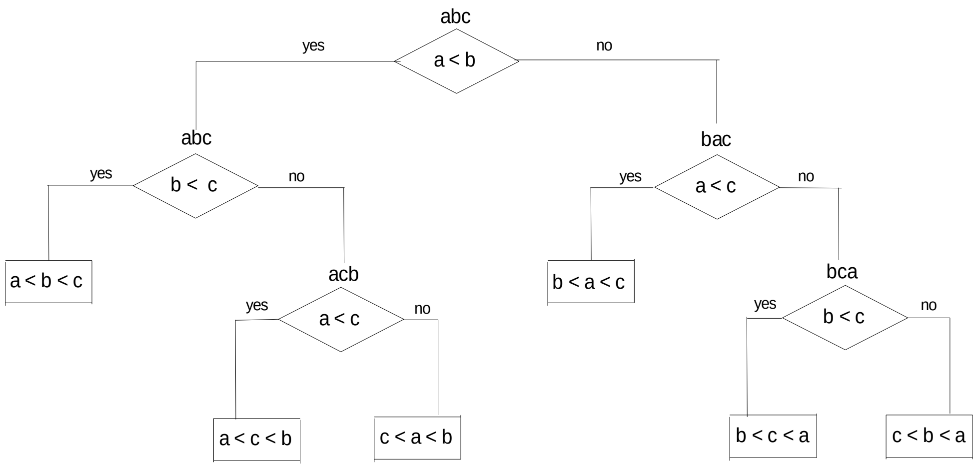
Differentiate with the aid of example each of the following pairs:

a. Optimization problem and decision problem

In optimization problem, we have an objective function that we need to maximize or minimize it, while decision problem is the kind of problems that is solved using a series of Yes/No questions.

For example:
Travelling salesperson problem is an optimization problem, where the objective function is to minimize the overall cost, with the constraints of passing through each city once and returning to the initial city.

On the other hand, the sorting of a set of numbers can be considered as a decision problem, for example if we try to sort three numbers, we can do that using the shown set of questions:



b. Tractable and NP-complete problems

Tractable problem is the kind of problems that can be solved in polynomial time. For example, sorting a set of numbers is a tractable problem.

NP-Complete problem is a decision problem where it is as hard as any problem in NP, for example CNF satisfiable problem is an NP-complete problem.

c. Decidable and undecidable problems

Decidable problem if there exists a corresponding Turing machine which halts on every input with an answer- yes or no. while, undecidable Problems: The problems for which we can't construct an algorithm that can answer the problem correctly in finite time.

For example, Post's correspondence problem is an undecidable problem, while searching for a key in a set of numbers is a decidable problem.

Question 3

For each of the following regular expressions, indicate if the given strings can be generated from the regular expression or not:

a. R = (a + ab)* and the following strings {ε, aba, aab, aba, aaa}

All given strings can be generated.

b. R = (00)* and the following strings {{ε, 00, 000}

The first two strings can be generated, while the third one cannot.

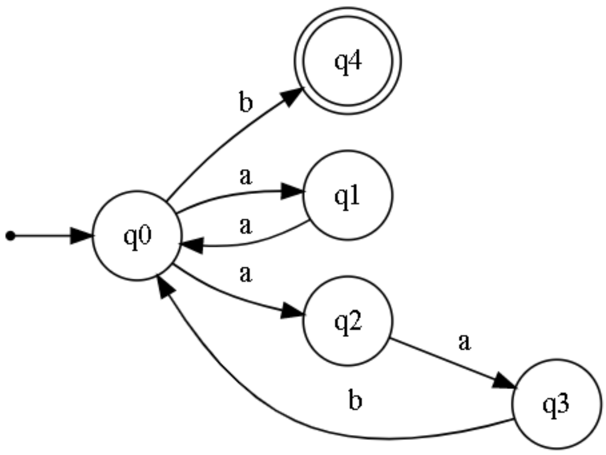
c. R = [(0 + 1)* 0 (0 + 1)* 1 (0 + 1)*] + [(0 + 1)* 1 (0 + 1)* 0 (0 + 1)*]

and the following strings {01, 10, 1100, 00, 11}

The last two strings cannot be generated while the rest can.

Question 4

Convert the following Non-Deterministic Finite Automata (NFA) into Deterministic Finite Automata (DFA).



For the given NFA, the transition table is as follows:

state	a	b
q_0	q_1, q_2	q_4
q_1	q_0	-
q_2	q_3	-
q_3	-	q_0
q_4	-	-

Step 1: Q' is empty
Checks if Q' is empty.

Step 2: Add the first state to Q'
Add the first state to Q' such that it now has *q*0.

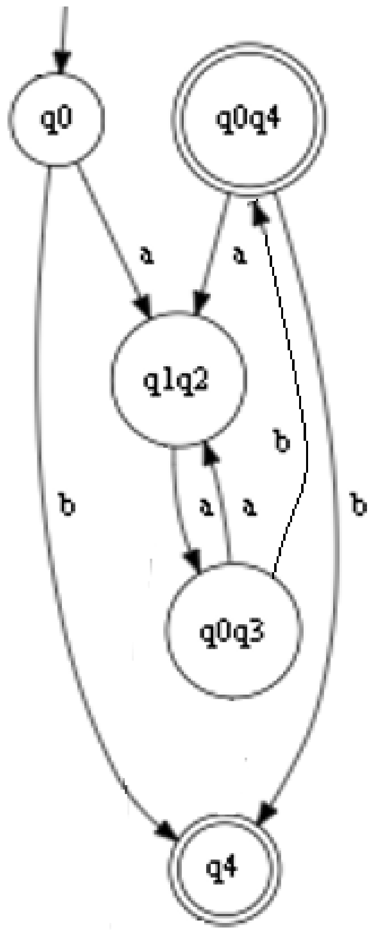
Step 3: Find states for each input symbol
For each state in Q', we will find the states for each of the input symbols. Q' has *q*0 presently. Check transitions for 'a*a*' and 'b*b*' from the transition table for *q*0. Update the table as new states are reached.

state	a	b
q_0	q_1, q_2	q_4

state	a	b
q_0	q_1, q_2	q_4
q_1, q_2	q_0, q_3	-
q_4	-	-

state	a	b
q_0	q_1, q_2	q_4
q_1, q_2	q_0, q_3	-
q_4	-	-
q_0, q_3	q_1, q_2	q_0, q_4

state	a	b
q_0	q_1, q_2	q_4
q_1, q_2	q_0, q_3	-
q_4	-	-
q_0, q_3	q_1, q_2	q_0, q_4
q_0, q_4	q_1, q_2	q_4



Question 5

1. Use pumping Lemma to prove that the following language is not regular:

   $L = \{(10)^p 1^q : p,q \in N, p \geq q\}$

We prove that L is not regular by contradiction, suppose L is regular.

- **Let $x = (10)^n 1^n$. Then $x \in L$ and $|x| = 3n \geq n$.**
- **By Pumping Lemma, there are strings u,v,w such that x =uvw, v not empty, $|uv| \leq n$, $uv^k w \in L$ for all $k \in N$.**

- **Let y be the prefix of x with length n. By our choice of x, $y = (10)^{n/2}$ if n is even, and $y = (10)^{(n-1)/2}$, hence, uv is a prefix of y, and $uv =(10)^j$ for some $j \in N$ with $0 \le j \le n/2$, or $uv =(10)^j1$ for some $j \in N$ with $0 \le j < n/2$.**
- **Depending on whether |uv| is even or odd, v is some nonempty substring of $(10)^j$ for some j where $0 \le j \le n/2$, or v is some nonempty substring of $(10)^j1$ for some j where $0 \le j < n/2$.**
- **There are 3 cases to consider: (a) v starts with 0 and ends with 0. (b) v starts with 1 and ends with 1. (c) v starts and ends with different symbols. For case (a), $uv^0w = uw$ contains 110 as a substring. Thus $uv^0w \in L$, [110 is not a substring of any string in L] which makes contradiction. Similarly for case (b), uv^0w contains 00 as a substring. For case (c), $v = (10)^i$ or $v = (01)^i$, where $0 < i$. So $|v| = 2i$. Thus $uv^0w = uw =(10)^{n-i}1^n \notin L$, which makes contradiction. We reach a contradiction in all cases. Therefore, L is not regular.**