1

Subhojit Ghimire
1912160

UG END SEMESTER EXAMINATION, 2021

BRANCH : CSE

SCH ID: 1912160

SUBJECT: Theory of Computation

SUBJECT CODE: CS-204

SEMESTER: IVth

DATE : 3rd May, 2021

Q.1.

Ans: Option (a): Yes, NFAs really help in representing languages exponentially more compactly than minimised DFAs.

Explanation:

Let $L_n = \{ w \in \{0,1\}^* \}$

Here, the n^{th} to last symbol in w is 1.

For all n, there exists NFA with $n+1$ states.

For fixed n, $N_n = (\{q_0, \dots, q_n\}, \{0,1\}, \delta, q_0, \{q_n\})$

Here, $\delta : \delta(q_0, 0) = \{q_0\}$

$\delta(q_0, 1) = \{q_0, q_1\}$

$\delta(q_i, a) = \{q_{i+1}\} \ \forall \ i \in \{1, 2, \dots, n-1\}$
$a \in \{0,1\}$

$\delta(q_n, a) = \phi \quad \forall \ a \in \{0,1\}$

This proves that N_n recognises the language L_n.
and for any $n \in N$, if there exists a DFA recognising the language L_n, it must have 2^n states.

This proves that there exists a language L_n where the number of states in the least possible DFA is 2^n, but there exists an NFA for the same language with $n+1$ states.

(2)

Subhojit Ghimire
1912160

Q.30

Ans: Option (c): $L = \{uawb : u, w \in (a+b)^*, |u| = |w|\}$

Explanation:

Consider a language, $L' = \{uaw : u, w \in \{a,b\}^*, |u| = |w|\}$. Here, u and w can be generated in parallel, keeping something in the middle that will turn into a.

L is L' with b tacked on the end.

So, the grammar for L is:

$G = (\{S, T, a, b\}, \{a, b\}, R, S)$

And, $R = \{S \to Tb,\ T \to aTa \mid aTb \mid bTa \mid bTb \mid a\}$

[options were bit confusing]

Q.20

Ans: Option (c) $L_1 \cap L_2$ is neither necessarily non-regular nor regular. Can't say for sure

Explanation:

Given L_1 is regular

L_2 is not regular

$L_2 \not\subset L_1$

$L_1 \not\subset L_2$

To check regularity: $(L_1 \cup \overline{L_2}) \cap (L_2 \cup \overline{L_1}) = \phi$

for L_1 and L_2 both regular.

E.g.: $L_1 = \{a^n b^n : n \in N\}$

$L_2 = \{a^n : n \in \text{Prime}\}$

$L_1 \cap L_2 = \phi$ is regular.

To check irregularity: Let $A = a^* b^*$

$B = a^n b^n$

for infinite sets, $A \cap B = B$ (i.e., not regular)

Subhojit Ghimire
1912160

Q. 40

Ans Option (b): There exists no membership algorithm for L.

Explanation:

Let M be universal turing machine
Let A and B be two states on input 0 and 1.

When A is entered, it stays in A forever.
When B is entered, it goes through some sequence that goes through all the states, ending in A.

If we make D(x) that analyses, whether X(x) enters each state atleast once.

If each state is entered at least once, then it goes to state A, if it does not, then it enters state B.

We see that D(x) is entering each state, if and only if X(x) does not enter each state. Then it leads to contradict D = X.

This proves that L is undecidable, i.e, there exists no membership algorithm for L.

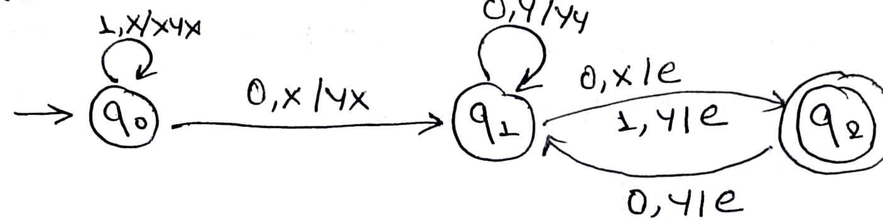
Subhojit Ghimire
1912160

Q.5.

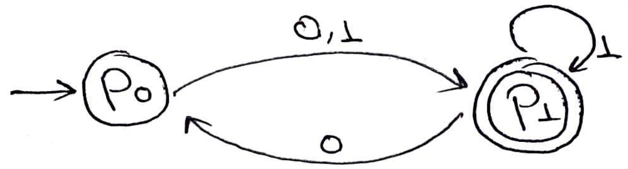
Ans: Option (c): Not Context-free but a Context-sensitive language.

Explanation:

PDA:



DFA:



Case (i): if $i = 2$, if 0 is added to string.

Number of 0's $= n+1$

$\therefore$ Not language.

case (ii): when $i = 2$,

Number of 0's >= Number of 1's

Number of 1's >= Number of 2's

case (iii): if $i = 0$,

Number of 1's <= Number of 0's

Number of 2's <= Number of 1's

case (iv): Number of 1's = Number of 2's

Case (v): if $i = 0$,

Number of 2's <= Number of 1's

$\therefore$ It is not context free, rather context-sensitive language.