Roll No.: ____________________

CS 208 : Automata Theory and Logic Spring 2024

Quiz 2

Time: 21:00 - 23:30 *Max Marks: 50*

Instructions:

- ***Please write your roll number on all pages in the space provided at the top.***
- *Be brief, complete, and stick to what has been asked.*
- ***You must write your answer for every question only in the space allocated for answering the question. Answers written outside the allocated space risk not being graded.***
- ***You can use an extra answer book for rough calculations. You must write your roll number on the extra answer book if you are using one.***
- ***You must submit this question+answer book in its entirey along with any extra answer book for rough calculations (if you used one).***
- *Untidy presentation of answers, and random ramblings will be penalized by negative marks.*
- *Unless asked for explicitly, you may cite results/proofs covered in class without reproducing them.*
- ***If you need to make any assumptions, state them clearly.***
- ***Do not copy solutions from others. All detected cases of copying will be reported to DADAC with names and roll nos. of all involved. The stakes are high if you get reported to DADAC, so you are strongly advised not to risk this.***

DO NOT TURN THIS PAGE UNTIL YOU ARE ASKED TO.

Marks:

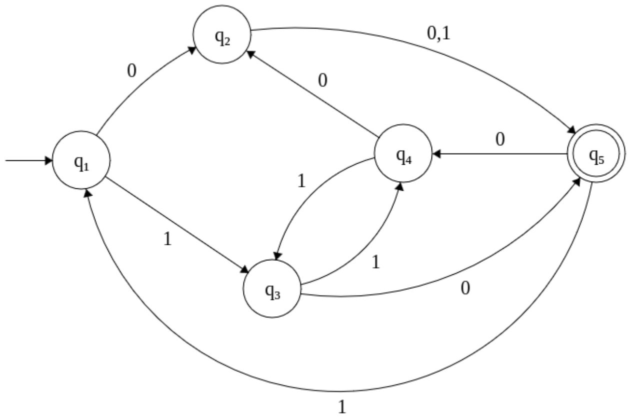
1a	1b	2a	2b	3a	3b	3c	4	5a	5b

THIS PAGE IS INTENTIONALLY LEFT BLANK

Roll No.: ____________________

1. **Compressed automata** [10]

(a) Consider the DFA shown below. [5]



We wish to compress this DFA to one that has the minimum number of states and yet accepts the same language. In order to do this, you are required to fill in the non-shaded cells of the table below for every pair of states (q_i, q_j) as follows:

- If q_i and q_j are indistinguishable, put an X in the row/column corresponding to q_i and row/column corresponding to q_j.
- If q_i and q_j are distinguishable, put a shortest length string (could even be ε) that distinguishes q_i from q_j in the row/column corresponding to q_i and row/column corresponding to q_j.

Please be mindful of the labels of the rows and columns. They may not be in the order you expect.

	q_2	q_5	q_1	q_3	q_4
q_2					
q_5					
q_1					
q_3					
q_4					

(b) The language $L = \{ww \mid w \in (\mathbf{0}+\mathbf{1})^*\}$ is known not to be regular. Show that the string ε is in a singleton Nerode equivalence class of its own, i.e. no other string in $(\mathbf{0}+\mathbf{1})^*$ can be Nerode equivalent to ε for the above language. You must clearly justify why for every string $w \neq \varepsilon$, you can find a string $x \in (\mathbf{0}+\mathbf{1})^*$ such that $\varepsilon.x \in L$ and $w.x \notin L$ or vice versa. Answers without justification will fetch no marks. 5

Roll No.: ______________________

2. **How's your grammar?** [10]

Consider the context-free grammar G shown below, where $\{S, A, B\}$ is the set of non-terminals, S is the start symbol and $\{0, 1\}$ is the set of terminals.

$$\begin{aligned} S &\rightarrow AB \mid SS \mid B \mid 1 \\ A &\rightarrow 0S \mid 1B1 \mid \varepsilon \\ B &\rightarrow 1S \mid 1B \end{aligned}$$

Let $L(G)$ denote the language generated by G.

(a) Construct a grammar G' in Chomsky Normal Form such that $L(G') = L(G)$. You must show every step of the transformation of G to G'. Simply giving the Chomsky Normal Form grammar without showing detailed steps will fetch no marks. [5]

(b) Give an infinite family of strings $\mathcal{F} = \{w_i \mid i \geq 1, w_i \in \{0,1\}^*\}$ such that each w_i satisfies the following properties: [5]

- $|w_i| = i + 1$
- $w_i \notin L(G)$
- There exists a prefix v_i of w_i such that $|v_i| \geq i - 1$ and $v_i \in L(G)$

For each $i \geq 1$, you must clearly describe what the string w_i is, what its prefix v_i is, ***and*** give justification why w_i and v_i satisfy the three properties given above. No marks will be awarded if any of the above parts is missing. In other words, even if you have given the correct w_i and v_i for all i, and shown that the one or two of the properties hold, you will not be awarded any marks unless you show all the three properties hold.

3. **Dr. Turing comes to town** [10]

   (a) In this question, we will consider a single-tape non-deterministic Turing machine M with at most four states q_0, q_1, q_2 and q_3, and with tape alphabet $\Gamma = \{0, 1, b\}$, where b stands for the blank symbol. The input alphabet of our machine is $\Sigma = \{0, 1\} \subset \Gamma$. The tape may be assumed to be doubly infinite. The transitions of the machine M, using the notation discussed in class, are given in the following two tables (split to fit in the page). [3]

Current state	Symbol read	Next state	Symbol written	Head movement
q_0	0	q_1	0	Right
q_0	0	q_1	1	Right
q_0	1	q_2	0	Left
q_0	b	q_0	1	Right
q_2	1	q_0	1	Left
q_2	0	q_1	1	Left
q_2	b	q_3	0	Right

Current state	Symbol read	Next state	Symbol written	Head movement
q_1	0	q_1	1	Right
q_1	1	q_2	1	Left
q_1	b	q_1	0	Right
q_3	1	q_0	0	Left
q_3	b	q_1	0	Left

   Using the notation for configurations or instantaneous descriptions of Turing machines studied in class, show the first ten configurations/instantaneous descriptions of M starting from $q_0 01101$. Consider $q_0 01101$ as the first configuration in this sequence of configurations.
   [Note: The answer to this question may not be unique]

(b) A student defines the language $H(M)$ of the above Turing machine as the set of all strings $w \in \{0,1\}^*$ such that the configuration/instantaneous description q_0w of M ***always*** (i.e. regardless of the way non-deterministic choices are resolved) leads to a configuration in which the machine M has no moves to make. [4]

Show that for every $i \geq 1$, there is a string $w_i \in \{0,1\}^*$ such that $|w_i| = i$ and $w \in H(M)$. You must clearly describe what w_i is for each i, and give justification for why each such $w_i \in H(M)$.

(c) Give an example of a string $w \in \{0,1\}^*$ such that the Turing machine M when started in the configuration/instantaneous description q_0w doesn't halt. You must clearly show that the sequence of configurations reached by M is such that the Turing machine always has a move to make. Answers without this justification will fetch 0 marks. [3]

4. **Let's build a machine** 10

In this question, you are required to design a single-tape (doubly infinite tape) deterministic Turing machine M with tape alphabet $\Gamma = \{0, 1, *, b\}$, where b represents the blank symbol. The set of states of the machine includes a starting state q_0 and a halting state q_h that is distinct from q_0. The machine starts in state q_0 and if at any time during its execution it enters state q_h, it halts, i.e. it can't make any move regardless of what tape symbol it reads. *In no state other than q_h should the machine halt when reading any symbol on the tape.*

We want the machine M to have the following behaviour: If we start the machine in the initial configuration/instantaneous description $q_0 0^i$, then the machine should halt in the configuration/instantaneous description $q_h 0^i 1^i$, for every $i \geq 1$.

Describe the machine M by listing its set of states and transitions. Use as few states as possible. You should describe the transitions in the form of a table, as was done in the previous question. You must also justify why your machine has the desired behaviour stated above. Simply giving the machine wihtout justification will fetch no marks.

5. **Frugal PDAs** 10

We have studied how to construct a PDA from a given CFG. Did you notice that every PDA constructed in this manner always has a single state in its finite state control? This effectively proves that for every PDA, there is an equivalent one that has a single state: convert the PDA to a CFG, and then convert the CFG back to a PDA. Hence, state minimization for PDAs is an uninteresting question.

What happens if we restrict the count of stack symbols? In other words, suppose you are allowed to have as many (but finite, of course!) states in the finite state control of your PDA, but you can have only one stack symbol, say X. Clearly, X must also be the initial symbol on the stack. Let's also agree that our PDA accepts by emptying its stack. We will call such a PDA a 1-*counter machine* (because a stack with only one symbol is nothing but a counter).

(a) Show that there exist context-free languages that cannot be accepted by any 1-counter machine. Simply giving a CFL and claiming without proof that it cannot be accepted by any 1-counter machine will fetch no marks. Similarly, simply giving a CFL and claiming (with our without proof) that it cannot be accepted by a specific 1-counter machine will also fetch no marks. You must give complete reasoning why your CFL can't be accepted by ***any*** 1-counter machine. 7

(b) Now suppose we say that our PDA can have only 1 state *and* also only 1 stack symbol. Clearly, such a PDA can accept (by empty stack) languages that are non-regular, e.g. we can accept $\{0^n 1^n \mid n \geq 0\}$ by such a PDA. Prove that there are regular languages that cannot be accepted by such a PDA. Once again, simply giving a regular language and claiming without proof that it can't be accepted by a 1-state, 1-stack symbol PDA will fetch no marks. Similarly, showing that a regular language is not accepted by a specific 1-state 1-stack symbol PDA will also fetch no marks. You must show that your regular language cannot be accepted by ***any*** 1-state, 1-stack symbol PDA. [3]