Name ______________________

Student Number ______________________

HAND IN
answers recorded
on question paper

BISHOP'S UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE

CS 310

MID-TERM EXAMINATION

20 February 2026

Instructor: Stefan D. Bruda

Instructions

- This examination is *80 minutes in length* and is *open book*. You are allowed to use any kind of printed documentation. Electronic devices are not permitted. You are *not* allowed to share material with your colleagues. *Any violation of these rules will result in the complete forfeiture of the examination.*
- In particular, be aware that *any electronic device in your possession in the examination room will be assumed to have been used and so will result in zero marks being assigned to your exam.*
- There is no accident that the total number of marks add up to the length of the test in minutes. The number of marks awarded for each question should give you an estimate on how much time you are supposed to spend answering the question.
- *To obtain full marks provide all the pertinent details.* This being said, do not give unnecessarily long answers. In principle, all your answers should fit in the space provided for this purpose. If you need more space, use the back of the pages or attach extra sheets of paper. However, if your answer is not (completely) contained in the respective space, clearly mention within this space where I can find it.
- The number of marks for each question appears in square brackets right after the question number. If a question has sub-questions, then the number of marks for each sub-question is also provided.

When you are instructed to do so, turn the page to begin the test.

1		10	/	10			
2		5	/	5			
3		5	/	5			
4		10	/	10			
5		5	/	5			
6	a,b	15	/	15			
7	a,b,c,d	25	/	25			
8		5	/	5			
	Total:	80	/	80	=	30 /	30

The highest grade was 30 (100%), the lowest grade (excluding no-shows) was 7 (23%), and the average grade was 17.77 (60%).

1. [10] For each question below check *one* answer. Your response is a priori incorrect if more than one answer is checked.

   - Which of the following statements is true:
     - ☐ Some finite languages are not regular.
     - ☑ Some regular languages are not finite.
     - ☐ Some regular languages are not context-free.
     - ☐ The above three statements are all false.
   - Which of the following statements is true:
     - ☐ Some regular expressions do not have equivalent nondeterministic finite automata.
     - ☐ Some regular expressions do not have equivalent deterministic finite automata.
     - ☐ Some nondeterministic finite automata do not have equivalent deterministic finite automata.
     - ☑ The above three statements are all false.
   - Let $L = (a + ba^*b)^*$ and $R = (a^*ba^*ba^*)^*$. Which of the following statements is true:
     - ☐ $L \subsetneq R$
     - ☑ $R \subsetneq L$
     - ☐ $L = R$
     - ☐ The above three statements are all false.
   - Let A and B be languages over an alphabet Σ. Then the following is true:
     - ☐ If A is nonregular, then $A + B$ is nonregular.
     - ☐ If A and B are nonregular, then $A + B$ is nonregular.
     - ☑ If $A + B$ is nonregular, then A or B is nonregular.
     - ☐ The above three statements are all false.
   - The language $\{0^k 1^{2n} 0^{3k} : k, n \geq 0\}$:
     - ☐ is both regular and context-free.
     - ☑ is context-free but is not regular.
     - ☐ is regular but is not context-free.
     - ☐ is neither regular nor context-free.

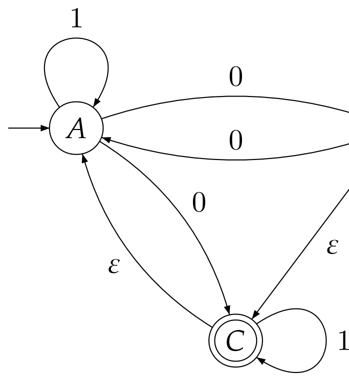
2. [5] Give a regular expression for the language over $\Sigma = \{a, b\}$ of all the strings that contain the substring aba and have an odd number of b symbols.

ANSWER:

$(a + ba^*b)^*aba(a + ba^*b)^*$

---

3. [5] Use the systematic method described in the course to convert the following state transition diagram into an equivalent state transition diagram without ε-transitions.

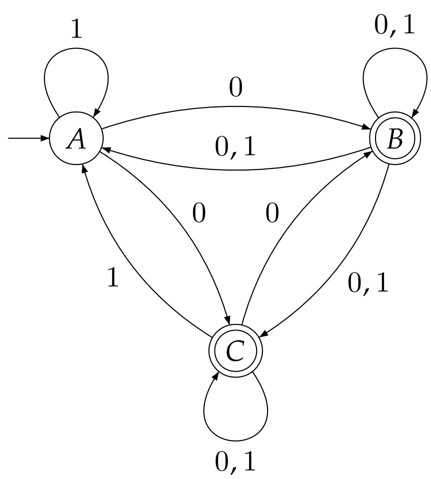


Answer:

The following paths need to be "contracted":

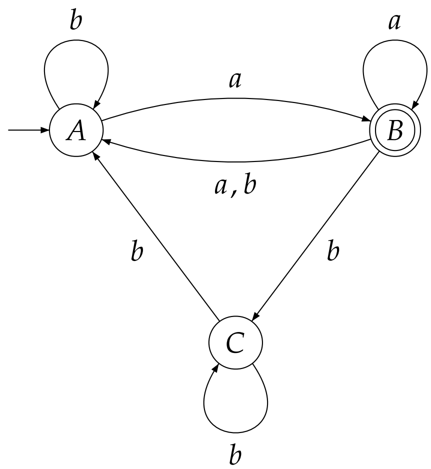
$$
\begin{array}{ll}
B \xrightarrow{\varepsilon} C \xrightarrow{1} C & \text{becomes} \quad B \xrightarrow{1} C \\
C \xrightarrow{\varepsilon} A \xrightarrow{1} A & \text{becomes} \quad C \xrightarrow{1} A \\
C \xrightarrow{\varepsilon} A \xrightarrow{0} B & \text{becomes} \quad C \xrightarrow{0} B \\
C \xrightarrow{\varepsilon} A \xrightarrow{0} C & \text{becomes} \quad C \xrightarrow{0} C \\
B \xrightarrow{\varepsilon} C \xrightarrow{\varepsilon} A \xrightarrow{0} B & \text{becomes} \quad B \xrightarrow{0} B \\
B \xrightarrow{\varepsilon} C \xrightarrow{\varepsilon} A \xrightarrow{0} C & \text{becomes} \quad B \xrightarrow{0} C \\
B \xrightarrow{\varepsilon} C \xrightarrow{\varepsilon} A \xrightarrow{1} A & \text{becomes} \quad B \xrightarrow{1} A
\end{array}
$$

B becomes accepting since it is connected to the accepting state C through an ε-transition.



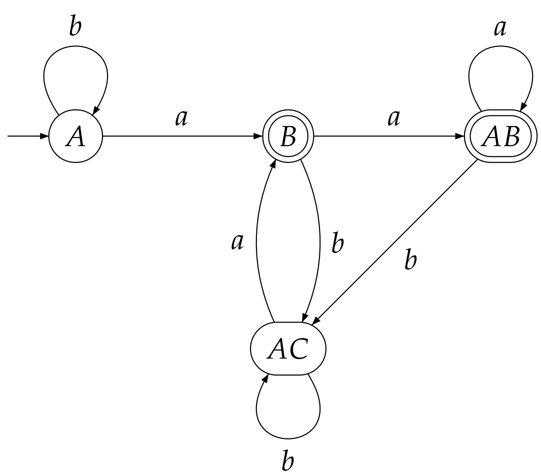
---

4. [10] Convert the following nondeterministic state transition diagram into an equivalent deterministic state transition diagram.



Label the states of the deterministic diagram by sets of states of the nondeterministic diagram (like we did in class).

ANSWER:

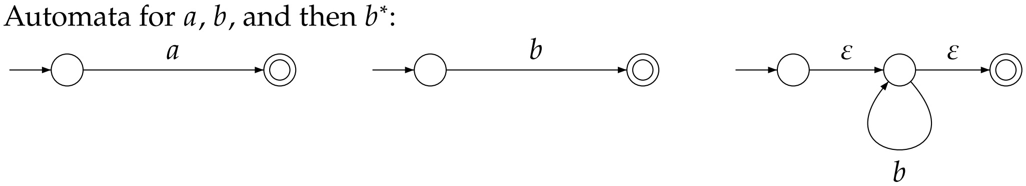


---

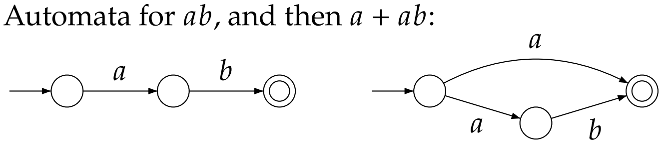
5. [5] Draw a state transition diagram that recognizes the language $(a + ab)^*b^*$. Justify your answer by showing all the intermediate finite automata.

Answer:

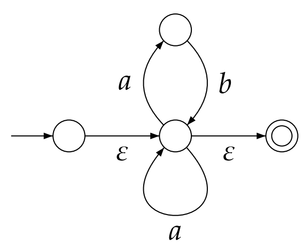
- Automata for a, b, and then b^*:



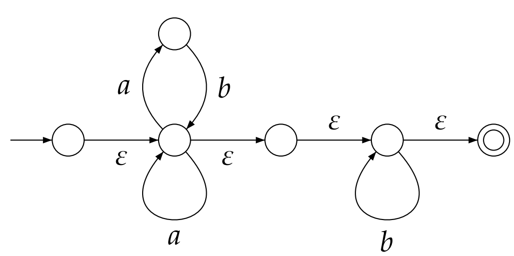
- Automata for ab, and then $a + ab$:



- Automaton for $(a + ab)^*$:



- Automaton for $(a + ab)^*b^*$:



---

6. [15] Are the languages L_1 and L_2 below over the alphabet $\Sigma = \{0, 1\}$ regular or non-regular? Justify your answer carefully.

   (a) [10] $L_1 = \{0^i 1^j : j \geq i \geq 0\}$

   ANSWER:

   The language is not regular. We assume that it is, so the pumping lemma applies. Let n be the threshold from the lemma and let $w = 0^n 1^{n+1}$. Clearly $|w| \geq n$ and $w \in L_1$. Therefore $w = xyz$ such that $|xy| \leq n$, $y \neq \varepsilon$, and $xy^i z \in L_1$ for all $i \geq 0$ which implies that $xy^3 z \in L_1$. Since $|xy| \leq n$ and the first n symbols of w are all 0 it must be that $y = 0^k$ for some $k > 0$ and therefore $xy^3 z = 0^{i+2k} 1^{i+1}$. Since $xy^3 z \in L_1$ it must be that $i + 1 \geq i + 2k$ and therefore $2k \leq 1$, which cannot be true unless $k = 0$, a contradiction.

   (b) [5] $L_2 = L_1 + 0^*$

   ANSWER:

   Again we assume that L_2 is regular and so the pumping lemma applies. We choose again $w = 0^n 1^{n+1}$ (we can choose any long-enough string). We already know that $xy^3 z \notin L_1$ (see previous question) and we also notice that $xy^3 z \notin 0^*$ (since it contains $n + 1 > 0$ occurrences of 1). Thus $xy^3 z \notin L_2$, again a contradiction.

   ---

7. [25] Consider the language $L = \{a^{2i}b^{2i}c^{2j} : i \geq 1, j \geq 0\}$.

(a) [5] Give a context-free grammar that generates L.

ANSWER:

$S \rightarrow AC$
$A \rightarrow aaAbb$
$A \rightarrow aabb$
$C \rightarrow ccC$
$C \rightarrow \varepsilon$

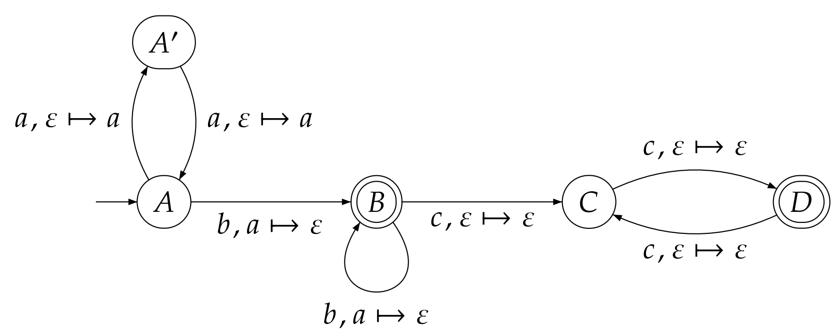
(b) [5] Give a derivation for the string $aaaabbbbcccc$ using your grammar from Question 7a.

ANSWER:

$S \Rightarrow AC \Rightarrow aaAbbC \Rightarrow aaaabbbbC \Rightarrow aaaabbbbccC \Rightarrow aaaabbcccc$

(c) [10] Draw a *deterministic* push-down automaton that recognizes the language L. A nondeterministic push-down automaton for L will receive partial marks.

ANSWER:



(d) [5] Draw a table that traces the run of your push-down automaton from Question 7c on input $aabbc$, listing the current state, the remaining input, and the stack at each step. Explain why the input is accepted or rejected (as the case might be).

ANSWER:

State	Input	Stack
A	$aabbc$	ε
A'	$abbc$	a
A	bbc	aa
B	bc	a
B	c	ε
C	ε	ε

At the end of the run we have an empty stack but we are not in an accepting state, so the input is rejected.

8. [5] Explain formally why the following grammar is not suitable for recursive descent parsing:

$$S \to aSa \quad S \to bSc \quad S \to \varepsilon$$

ANSWER:

We have first$(aSa) = \{a\}$ and follow$(S) = \{a, c\}$. Since $S \Rightarrow^* \varepsilon$ and first$(aSa) \cap$ follow$(S) = \emptyset$ the grammar is not suitable for recursive descent parsing.