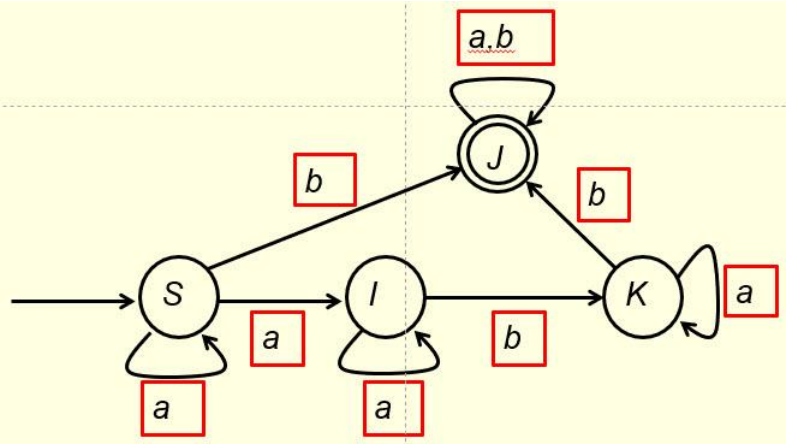
CS375 Homework Assignment 4-2 Solution Set (19 points)

Due date: 10/09/2026

1. Given the following regular grammar:

$$S \rightarrow aS \mid aI \mid bJ, \quad I \rightarrow bK \mid aI, \quad K \rightarrow aK \mid bJ, \quad J \rightarrow aJ \mid bJ \mid \Lambda$$

one can use the algorithm given in slide 39 of the notes "Regular Languages and Finite Automata- IV" to construct an NFA to recognize the language of that grammar. Fill out blanks in the following figure so that the resulting NFA would fulfill such a task, i.e., would recognize the language of the given grammar. (4 points)



2. Fill out the following blanks to make it a regular grammar with S being the start symbol for the following given regular expression: c*d*a* + b (5 points)

S → b | C | D | A | Λ
C → cC | D | A | Λ
D → dD | A | Λ
A → aA | Λ

or

S → a | cC | dD | aA | Λ
C → cC | dD | aA | Λ
D → dD | aA | Λ
A → aA | Λ

3. Using **Pumping Lemma** (slides 46-52 of the notes 'Regular Languages & Finite Automata-IV') one can show the language $L = \{a^n b^n \mid n \in \mathbf{N}\}$ is not regular (We need this property in the notes 'Context-free Languages and Pushdown Automata I). This is done by way of contradiction. We assume L is regular. Since L is infinite, Pumping Lemma applies. We then consider the string $s = a^m b^m$ where m is the number of states in the DFA that recognizes L. Since the length of s is bigger than m, by Pumping Lemma, there exists strings $x, y \; and \; z$ such that $s = xyz$, $y \neq \Lambda$, $|xy| \leq 2m$ and $xy^k z \in L$ for all $k \in N$. If $|xy| < 2m$ then the first repeated state on the acceptance path cannot be a final state. This statement is not true, why? (2 points)

The condition that |xy|< 2m implies the substring z is not an empty string. So xyz != xy. If the first repeated state on the acceptance path, I, is a final state, the substring xy would be accepted by the DFA. However, since the execution of the entire string is not completed yet, the DFA would continue executing the remaining portion of the string, z, until the last symbol of z is executed and the DFA would end at a final state. Hence, the first repeated state on the acceptance path could be a final state if the length of xy is smaller than 2m.

4. Fill out the following blanks to make it a context-free grammar for the given language over the alphabet {a, b}: $\{ a^{2n+3}b^{3n} \mid n >= 0\}$ (2 points)

S → aaSbbb | aaa

5. Fill out the following blanks to make it a context-free grammar for the given language:
$\{ a^{n+3} b^{2n} \mid n >= 0 \}$ $\{ a^{2n}b^{n+3} \mid n >= 0 \}$ (2 points)

S → AB, A → aAbb | aaa, B → aaBb | bbb

6. For the PDA's shown in Fig. 1 and Fig. 2, which one/ones or no one would accept the language $L = \{x \in \{a, b\}^* \mid N_a(x) = 1 + N_b(x)\}$ by empty stack? Use the execution of the string babaa to justify your answer. (4 points)

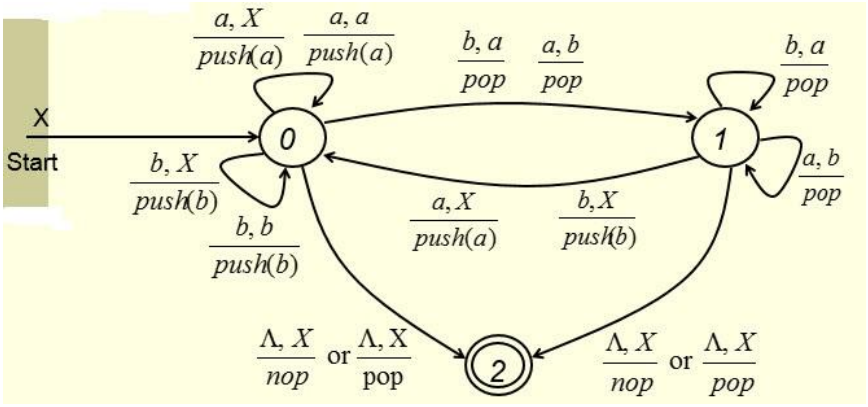


Fig. 1

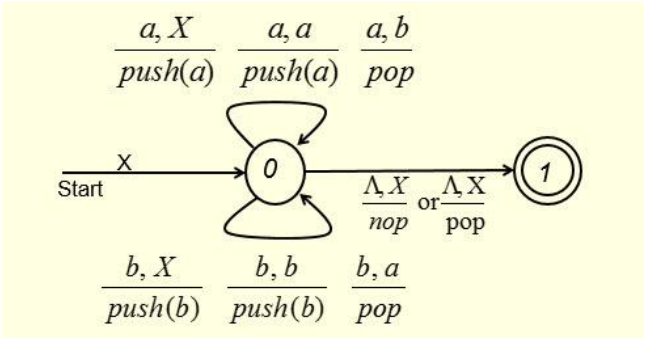


Fig. 2

Sol.
Option 1: use FA instructions

For the PDA in Fig. 1, we have

(0, babaa, X)
(0, b, X, push(b), 0)
(0, a, b, pop, 1)
(1, b, X, push(b), 0)
(0, a, b, pop, 1)
(1, a, X, push(a), 0)
(0, Λ, a, ?, ?)

We got stuck here (no instruction to use for the next step). The stack is not empty, so the PDA in Fig.1 does not accept L.

For the PDA in Fig. 2, we have

(0, babaa, X)
(0, b, X, push(b), 0)
(0, a, b, pop, 0)
(0, b, X, push(b), 0)
(0, a, b, pop, 0)
(0, a, X, push(a), 0)
(0, Λ, a, ?, ?)

We got stuck here (no instruction to use for the next step). The stack is not empty, so the PDA in Fig.2 does not accept L either.

Option 2: use instantaneous descriptions

For the PDA in Fig. 1, we have

(0, babaa, X)
(0, abaa, bX)
(1, baa, X)
(0, aa, bX)
(1, a, X)
(0, Λ, aX)

We got stuck here. The stack is not empty, so the PDA in Fig.1 does not accept L.

For the PDA in Fig. 2, we have

(0, babaa, X)
(0, abaa, bX)
(0, baa, X)
(0, aa, bX)
(0, a, X)
(0, Λ, aX)

We got stuck here. The stack is not empty, so the PDA in Fig.2 does not accept L either.