CMSC330 Fall 2023 Quiz 2

Proctoring TA: _____ **Name:** _____

Section Number: _____ **UID:** _____

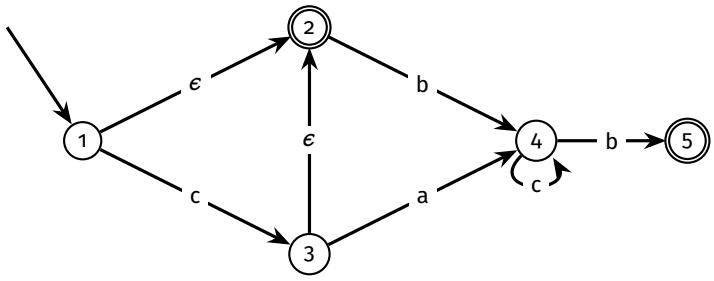
Problem 1: Basics

[Total 4 pts]

	True	False
Checking to see if an **arbitrary string of size 5** is a palindrome can be calculated with a FSM. If you can make a regex that describes all palindromic strigs of size 5, you can make an FSM	T	F
Checking to see if an **arbitrary string of any size** is a palindrome can be calculated with a FSM. Since you cannot make a regex to describe palindromes of arbitrary size, you cannot make an FSM	T	F
FSMs can represent regular languages By definition, a regular language can be made from a regular expresssion, thus also a FSM	T	F
Every regular expression has **exactly one** corresponding DFA. Multiple DFAs could describe the same regex.	T	F
On average, compared to a DFA, checking acceptance with an NFA is more computationally expensive NFA acceptance has to check every single path, DFA acceptance has only 1 path to check	T	F
NFAs have exactly one path during a graph traversal for any given input NFAs could have more than 1 path (which is why they are non-deterministic)	T	F
All DFAs are NFAs. DFAs are a subset of NFAs	T	F
A DFA can have a **only one** start state and final state They could have only 1 yes, but they don't have to	T	F

Problem 2: Finite State Machine Analysis

[Total 4 pts]



Which strings would the above Finite State Machine accept? Select all that apply.

(A) bb (B) cab (C) caccccab (D) baccccb (E) cb
(F) c (G) cabb (H) the empty string (I) cbcb (J) cbb

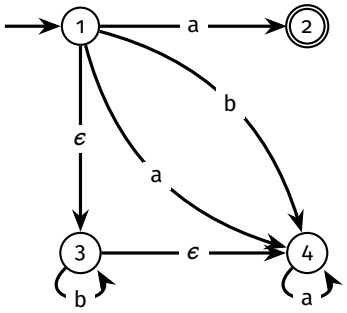
Write a regular expression that is equivalent to the above Finite State Machine:

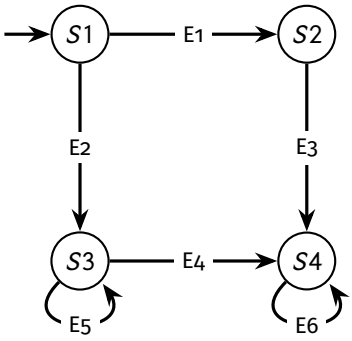
c?|(ca|cb|b)c*b

Problem 3: NFA to DFA

[Total 12 pts]

Consider the NFA and fill in the blanks of the equivalent DFA. Use the subset construction (on-demand) algorithm we gave in lecture/discussion. We will only be checking state names for partial credit.





S1: 1,3,4 S2: 2,4 S3: 3,4 S4: 4

E1: a E2: b E3: a

E4: a E5: b E6: a

What state(s) are final states? Select all that apply:

(A) $S1$ (B) $S2$ (C) $S3$ (D) $S4$

Scratch Space: