AFLL ISA-1 Practice Quiz

* Required

Answer All Questions (Time Limit : 45 min)

Choose the FALSE Statement * 1 point

- (●) Deterministic Finite Automata are strictly weaker class than Non-deterministic Finite Automata (NFAs), i.e., there exists a language that is accepted by an NFA but is not accepted by any DFA.
- ( ) Let Σ = (,) be an alphabet. Then the grammar G: S → (S) represents the empty language.
- ( ) There are regular languages that contain context-free languages as subsets.
- ( ) a^nb^m, where the alphabet is a, b and n ≥ 0, m ≥ 0, is a regular language.

The complement of an infinite language is necessarily finite. * 1 point

- ( ) True
- (●) False

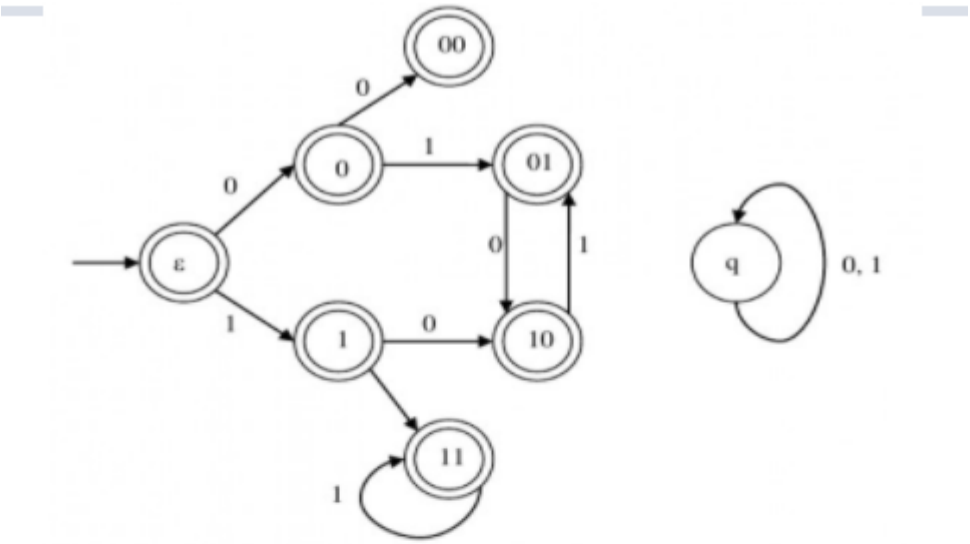
. Given an arbitrary non-deterministic finite automaton (NFA) with N states, the maximum number of states in an equivalent minimized DFA is at most. * 1 point

- ○ N^2
- ◉ 2^N
- ○ 2N
- ○ N!

The number of derivation steps required to derive a string of length x from a grammar in GNF are * 1 point

- ○ x - 1
- ◉ x
- ○ x + 1
- ○ 2x

Consider the set of strings on {0,1} in which, every substring of 3 symbols has atmost two zeros. For examples, 001110 and 011001 are in the language, but 100010 isnot. All strings of length less than 3 are also in the language. A partially completed DFA that accepts this language is shown below. * 2 points



(A)

	0	1
00	01	00
01		11
10	00	
11	10	
q		

(B)

	0	1
00	01	q
01		01
10	11	
11	01	
q		

(C)

	0	1
00	q	01
01		01
10	10	
11	01	
q		

(D)

	0	1
00	q	01
01		11
10	00	
11	10	
q		

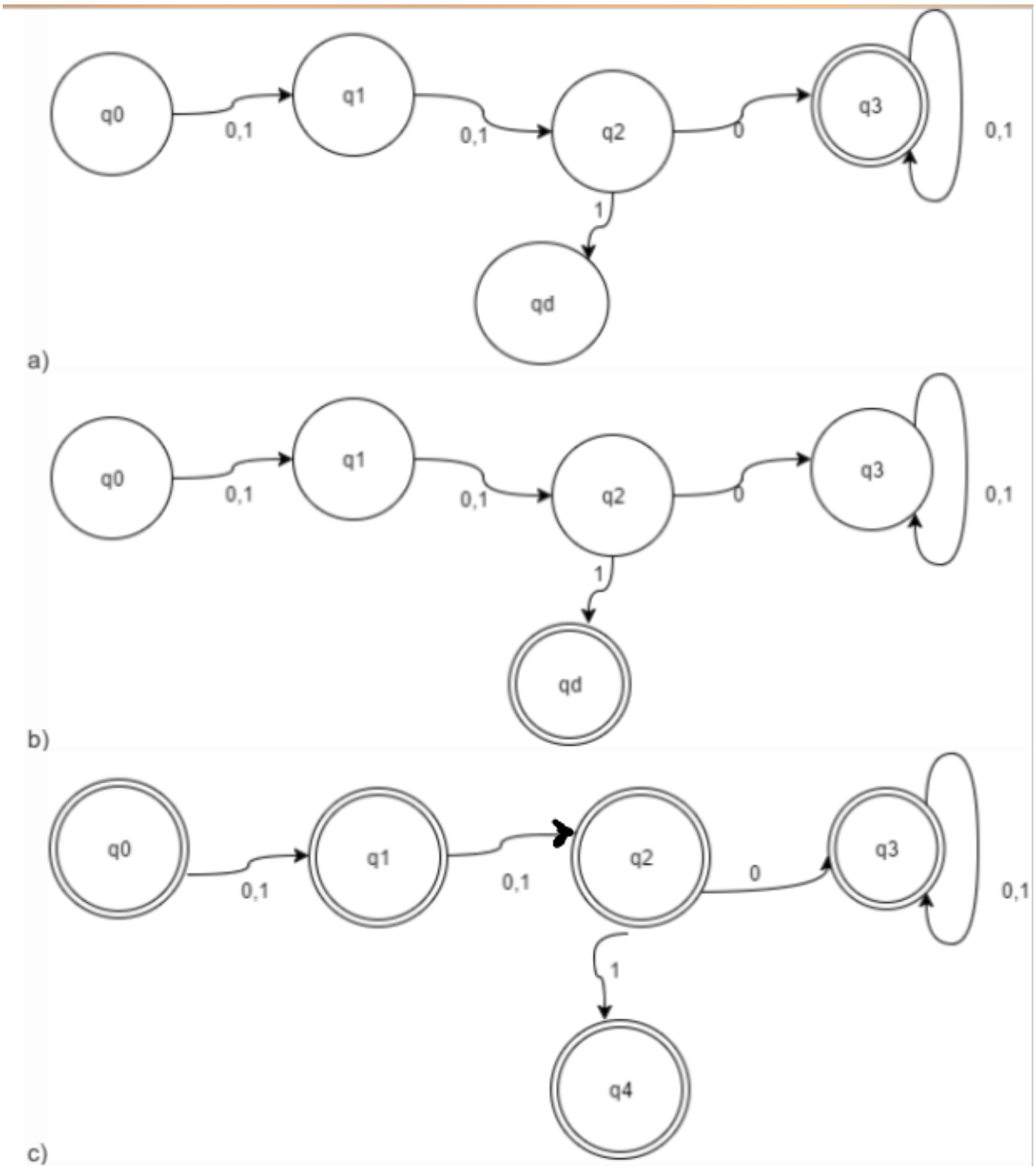
- ◯ A
- ◯ B
- ◯ C

- (●) D

Definition of a language L with alphabet {a} is given as following. L= { a^(n) | n > 0, and n is a positive integer constant} What is the minimum number of states needed in a DFA to recognize L? * 1 point

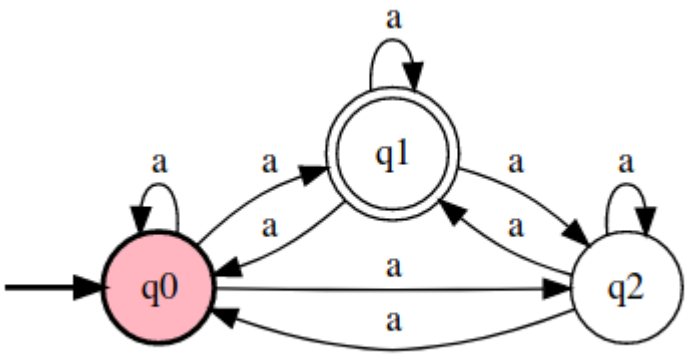
- ( ) n
- ( ) n + 1
- (●) n + 2
- ( ) 2^n
- ( ) 2^(n+1)

Which among the following NFA"s is correct corresponding to the given Language?L= {xϵ {0, 1} | 3rd bit from left is 0}. Please note q0 is the start state in each NFA. * 2 points



- [x] a
- [ ] b
- [ ] c
- [ ] None

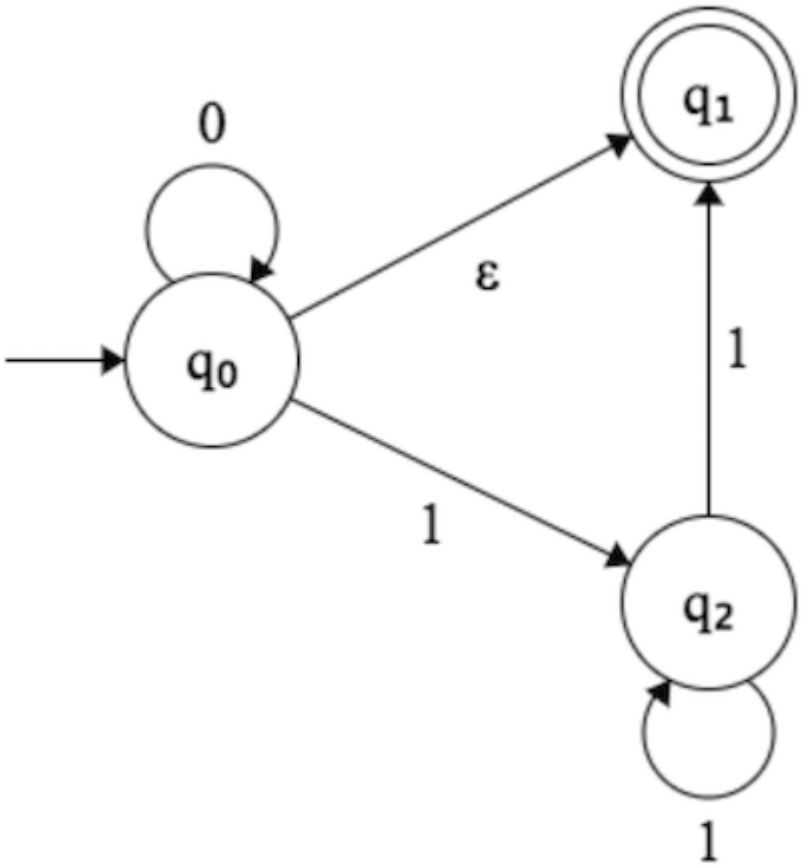
What is the complement of the language accepted by the NFA shown below?Assume ∑ = {a} * 1 point



- We cannot take compliment of NFA
- a+
- λ
- Φ

The language represented by the given NFA is (here ε is the empty string) 2 points

*



- 0* + 11*1
- 0*11*1
- 11*1
- 0*1*
- 0* (ε+ 11*1)

A PDA behaves like a FSM when the amount of auxiliary memory it has is * 1 point

- (●) 0
- ( ) 1
- ( ) 2
- ( ) Infinite

Consider the following regular expression: (a + ab + abc)∗ Which of the following regular expressions denote the same language as the above regular expression? * 1 point

- ( ) (ε + a + ab + abc)
- ( ) a(ε + b + bc)
- ( ) a*(ε + b + bc)*
- (●) (a(ε + b + bc))∗
- ( ) a∗ + (ab)∗ + (abc)*

Pick the right option to complete the given statement S, where S = "Any string of terminals that can be generated by the following CFG * 2 points

$$S \rightarrow XY$$
$$X \rightarrow aX \mid bX \mid a$$
$$Y \rightarrow Ya \mid Yb \mid a$$

- ○ Has atleast one b
- ○ Should end in an 'a'
- ○ Has no consecutive a's and b's
- ◉ Has atleast two a's

*

2 points

Consider the following Context-Free Grammar (CFG) G:

$$\begin{aligned} S &\rightarrow X \mid XY \\ X &\rightarrow aXb \mid aYb \\ Y &\rightarrow bYc \mid \epsilon \end{aligned}$$

where S, X, Y are nonterminal symbols, S is the start symbol, and a, b, c are terminal symbols.

Which of the following statements about the language $L(G)$ generated by G are correct?

(i) $\epsilon \in L(G)$

(ii) $aaabbbcc \in L(G)$

(iii) $aabbbbcc \in L(G)$

(iv) $\{a^i b^i b^j c^j \mid i, j \in \mathbb{N},\ i > 0\} = L(G)$

(v) The following CFG G' is equivalent to G above, i.e. $L(G') = L(G)$:

$$\begin{aligned} S &\rightarrow XY \\ X &\rightarrow aXb \mid ab \\ Y &\rightarrow bYc \mid \epsilon \end{aligned}$$

- ○ (i)
- ○ (ii)
- ◉ (iii)
- ○ (iv)
- ○ (v)

Identify all the nullable variables * 1 point

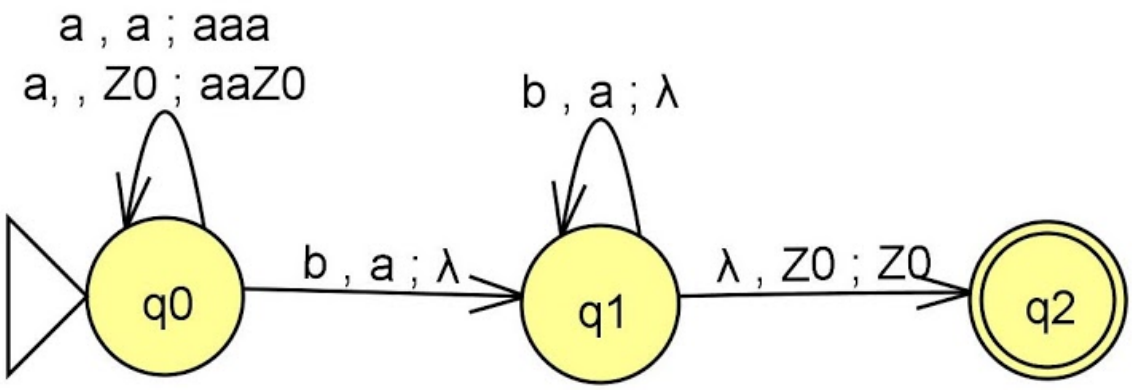
Consider the following Context-Free Grammar (CFG):

$$
\begin{aligned}
S &\rightarrow ABC \mid BC \\
A &\rightarrow aA \mid a \\
B &\rightarrow b \mid C \\
C &\rightarrow cc \mid dd \mid \epsilon
\end{aligned}
$$

S, A, B, and C are nonterminals, a, b, c, and d are terminals, and S is the start symbol.

- ○ C only
- ○ B and C Only
- ◉ S, B and C only
- ○ All variables are nullable

Identify the language of the given PDA * 2 points



- L={a^n b^n | n>=0}
- L={a^n b^2n | n>=0}
- L={a^n b^m | n>=0,m>=0}
- L={a^n b^2m | n>=0,m>=0}

Find the missing entry in the triangular table for the parsing of the string "110100" in the given grammar: * 1 point

S->AB|BC
A->BB|0
B->BA|1
C->AC|AA|0

S,B					
S,B	A,C				
S,B		S			
A,C	A	S	S,B		
A	S,B	S	S,B	C	
B	B	A,C	B	A,C	A,C
1	1	0	1	0	0

- A
- A,C
- C
- S,B

Given is the regular grammar for the regular expression (a+b+)+ with terminals {a,b} and non-terminals {S, P, Q}, S being the initial state and Q, Choose the production missing in the below grammar for the given regular expression. * 1 point

S->aS|aP
P->bQ
Q->aP|aS|lamda

- S->bS
- P->bP
- Q->bQ
- S->bP

Pick the string for which the given grammar is ambiguous. G=({S,B}, {a,b,c,d},{P},{S}), * 1 point

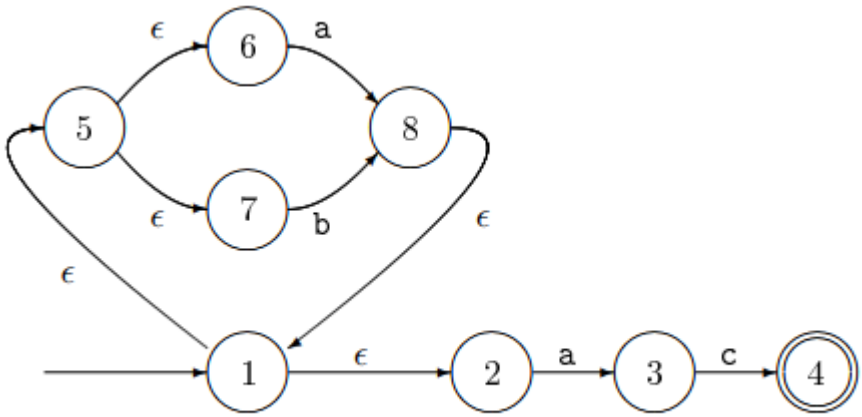
S->aS|SB|d
B->Bb|c

- aaad
- adc
- dcbb
- None of the mentioned options

Which of the following statement is false? * 1 point

- If we add a finite set of strings to a regular language, the result is a regular language
- If we remove a finite set of strings from a regular language, the result is a regular language
- If A is a nonregular language and B is a language such that B ⊆ A, then B must be regular
- If language A is recognized by an NFA , and language B is recognized by some DFA then A ◦ B is a regular language.

Find the λ-Closure of state 8 from the given NFA. * 1 point



- 8
- 8,1,2
- 7, 6, 5, 2, 1
- 8, 7, 6, 5, 2, 1

If language L={0,1}*, then the reversed language L^R = * 1 point

- (●) {0,1}*
- ( ) Φ
- ( ) Cannot be determined
- ( ) {1,0}

For which of the following Languages a FA can be constructed ? * 1 point

- ( ) { 0^i 1^i 0^j 1^j | i , j ≥ 0 }
- ( ) { 0^m 1^n | m ≠ n }
- ( ) { xx | x ∈ { 0 , 1 } * }
- (●) { x | x ∈ { 0 , 1 } * }

Identify the unambiguous grammar * 2 points

- ( ) S → a | aAb | abSb, A → aAAb | bS
- ( ) S → F | H, F → p | c, H → d | c
- (●) S -> baS | b
- ( ) S -> aSb | bSa | SS | λ

Back Next Page 2 of 4