1. Which one of the following pairs have different expressive power?

a) Deterministic Finite Automata AND Non Deterministic Finite Automata

b) Deterministic Push Down Automata AND Non deterministic Push Down Automata

c) Deterministic single tape turing machine and Non-Deterministic single tape turing machine

d) single tape turing machine and multiple tape turing machine

2. The lexical analysis for a modern computer language such as java needs the power of which one of the following models in a necessary and sufficient sense?

a) Finite state automata

b) Deterministic pushdown automata

c) Non-deterministic pushdown automata

d) Turing machine

3. Which one of the following problem is undecidable?

a) Membership problem of CFG

b) Ambiguity problem for CFG

c) Finiteness problem for FSA

d) Equivalence problem for FSA

4. The recognizing capability of Non Deterministic Finite State Machine and Deterministic Finite State Machine

a) is different b) sometimes different

c) is the same d) none of these

5. Finite State Machine can recognize

a) any grammar b) only CFG

c) any unambiguous grammar **d) only regular grammar**

6. Pumping lemma is generally used for proving

a) a given grammar is regular

b) a given grammar is not regular

c) weather two given regular expressions are not equivalent

d) none of above

7. Which one of following is regular?

a) Strings of 0's whose length is a perfect square

b) Set of all palindromes made up of 0's and 1's

c) String of 0's, whose length is a prime number

d) String of odd numbers of zeros

8. Which one of the following pairs of regular expression are not equivalent?

a) 1 (01) * and (10)*1

b) x (xx)* and (xx)*x

c) (ab)* and a*b*

d) x^+ and x*x^+

9. Assuming P!=NP, which of time is TRUE?

a) NP-complete=NP

b) NP-complete intersection P= Ø

c) NP-hard=NP

d) P=NP-complete

10. The lexical analysis for a modern language such as Java needs the power of which one of the following machine models in a necessary and sufficient sense?
a) Finite state automata
b) Deterministic pushdown automata
c) Non-deterministic pushdown automata
d) Turing machine

11. Let L denotes the language generated by the grammar S – >OSO/00. Which of the following is true?
(a) L = O
(b) L is regular but not O
(c) L is context free but not regular
(d) L is not context free

12. Consider the following two statements:
S1: { 0^2n |n >= l} is a regu1ar language
S2: { 0^m 0^n 0^(m+n) l m >= 1 and n >= 2} is a regu1ar language
Which of the following statements is correct?
a) Only S1 is correct
b) Only S2 is correct
c) Both S1 and S2 are correct
d) None of S1 and S2 is correct

13. Which of the following statements in true? (GATE CS 2001)
(a) If a language is context free it can always be accepted by a deterministic push-down

automaton
(b) The union of two context free languages is context free
(c) The intersection of two context free languages is context free
(d) The complement of a context free language is context free

14. Given an arbitrary non-deterministic finite automaton (NFA) with N states, the maximum number of states in an equivalent minimized DFA is at least. (GATE CS 2001)
(a) N^2
(b) 2^N
(c) 2N
(d) N!

15.Let S and T be language over ={a,b} represented by the regular expressions (a+b*)* and (a+b)*, respectively. Which of the following is true? (GATE CS 2000)

(a) ScT (S is a subset of T)
(b) TcS (T is a subset of S)
(c) S=T
(d) SnT=Ø

16. What is the complement of the language accepted by the NFA shown below? Assume ∑ = {a} and ε is the empty string

(A) Φ
(B) ε
(C) a
(D) {a, ε}

17. Given the language L = {ab, aa, baa}, whih of the following strings are in L*?
1) abaabaaabaa
2) aaaabaaaa
3) baaaaabaaaab
4) baaaaabaa
(A) 1, 2 and 3 (B) 2, 3 and 4 **(C) 1, 2 and 4** (D) 1, 3 and 4

18. Which of the following problems are decidable?
1) Does a given program ever produce an output?
2) If L is a context-free language, then is L’ (complement of L) also context-free?
3) If L is a regular language, then is L’ also regular?
4) If L is a recursive language, then, is L’ also recursive?
(A) 1, 2, 3, 4
(B) 1, 2,
(C) 2, 3, 4
(D) 3, 4

19. Which of the following strings do not belong the given regular expression?
(a)*(a+cba)
a) aa
b) aaa
c) acba
d) acbacba

Answer (d)

20. Which of the following regular expression allows strings on {a,b}* with length n where n is a multiple of 4.
a) (a+b+ab+ba+aa+bb+aba+bab+abab+baba)*
b) (bbbb+aaaa)*
c) ((a+b)(a+b)(a+b)(a+b))*

d) ((a+b)(a+b)(a+b)(a+b))

Answer (c)

Part- B

1. For $\Sigma = \{a, b\}$, let us consider the regular language $L = \{x \mid x = a2+3k \text{ or } x = b10 + 12k, k \geq 0\}$. Which one of the following can be a pumping length (the constant guaranteed by the pumping lemma) for L?

A. 3 B. 5 C. 9 **D. 24**

Answer D

2. Which one of the following statements is FALSE?
A. Context-free grammar can be used to specify both lexical and syntax rules.
B. Type checking is done before parsing.
C. High-level language programs can be translated to different Intermediate Representations.
D. Arguments to a function can be passed using the program stack.
Answer (B)

3. Consider the grammar: S-> aSa|bSb|a|b

The language generated by the above grammar over the alphabet {a,b} is the set of:
(A) All palindromes
(B) All odd length palindromes.
(C) Strings that begin and end with the same symbol
(D) All even length palindromes

4. Which one of the following grammar generates the language $L = \{a^i b^j \mid i \neq j\}$?

(A)

$$S \to AC \mid CB$$
$$C \to aCb \mid a \mid b$$
$$A \to aA \mid \epsilon$$
$$B \to Bb \mid \epsilon$$

(B) $S \to aS \mid Sb \mid a \mid b$

(C)

$$S \to AC \mid CB$$
$$C \to aCb \mid \epsilon$$
$$A \to aA \mid \epsilon$$
$$B \to Bb \mid \epsilon$$

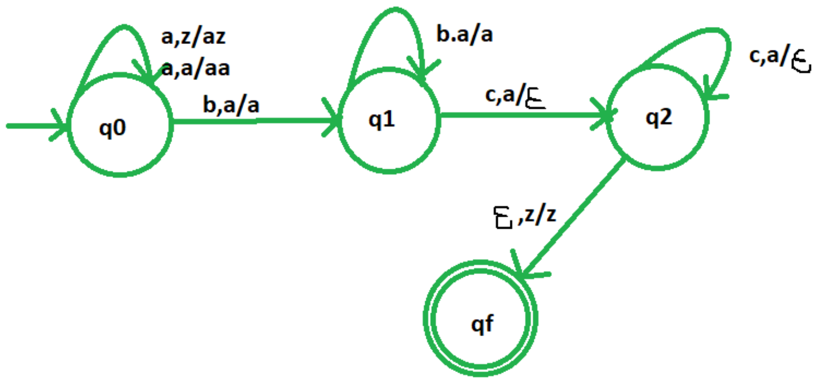
(D)

$$S \to AC \mid CB$$
$$C \to aCb \mid \epsilon$$
$$A \to aA \mid a$$
$$B \to Bb \mid b$$

Answer : D

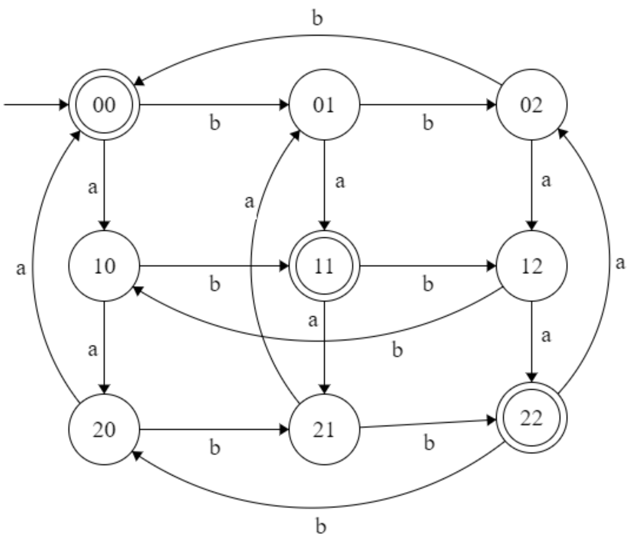
5. Design a non deterministic PDA for accepting the language L = {$a^n b^m c^n$ | m, n>=1}, i.e. L = { abc, abbc, abbbc, aabbcc, aaabccc, aaaabbbcccc, ... }

Solution:



Required PDA

6. Construct a deterministic finite automata (DFA) for accepting the language L = {w | w ∈ {a,b}* and Na(w) mod 3 = Nb (w) mod 3}.



Required DFA

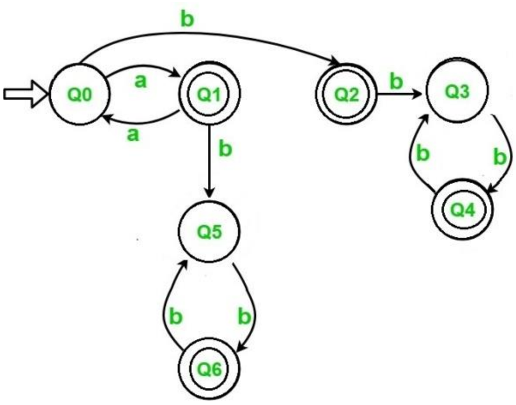
7. Design a deterministic finite automata (DFA) for accepting the language L = $\{a^n b^m \mid n+m=odd\}$

Approaches:

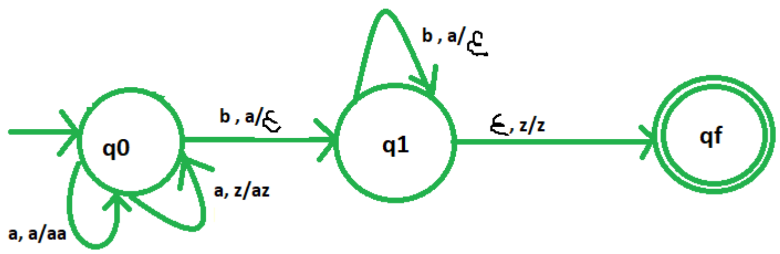
There is 2 cases which results in acceptance of string:

1. If n is odd and m is even then their sum will be odd
2. If n is even and m is odd then their sum will be odd

Any other combination result is the rejection of the input string.

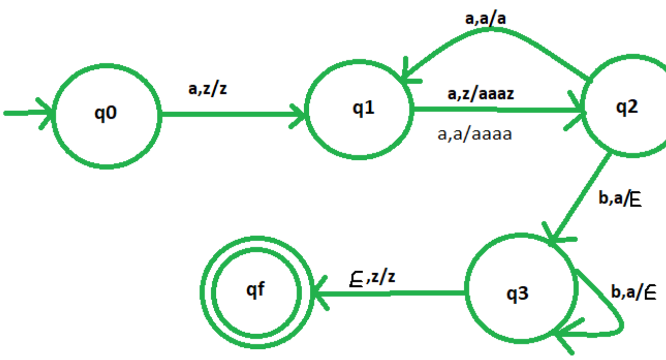


8. Design a non deterministic PDA for accepting the language L = $\{a^n b^n \mid n>=1\}$



Required PDA

9. Design a non deterministic PDA for accepting the language L = {$a^{2m}b^{3m}$ | m ≥ 1}



Required PDA

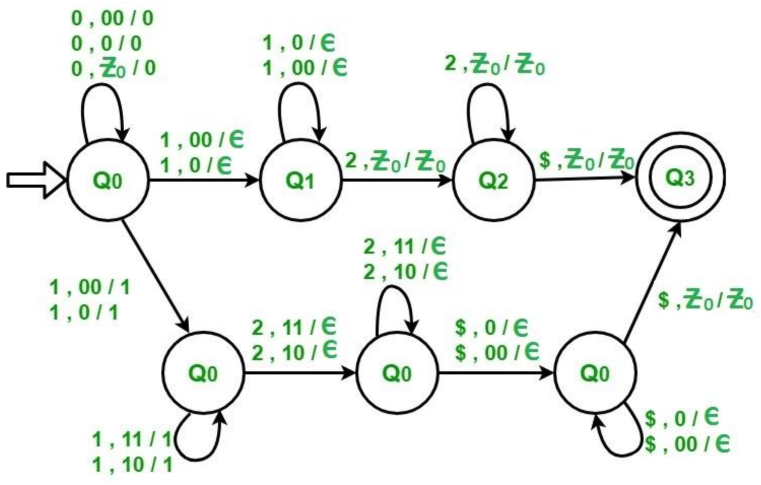
10. Design NPDA for L = {$0^i1^j2^k$ | i==j or j==k ; i , j , k >= 1}
There are 2 approaches for the solution. First is for i==j and second is for j==k. These are:

Steps for i == j :

1. Input all 0's in the stack
2. When we get 1 as input pop a 0 from stack and goto next state.
3. If input is 1 then pop 0 from stack.
4. If stack becomes empty (i.e., every 0 corresponding to a 1 has been popped so i = j) and input is 2 then ignore it and goto next state.
5. If input is 2 then ignore it . If input is finished and $ is received then goto final state.

Steps for j == k :

1. Input all 0's in the stack
2. When we get 1 as input push it onto stack and goto next state.
3. If input is 1 then push it onto stack.
4. If input is 2 pop a 1 from stack and goto next state.
5. If input is 2 then pop 1 from stack. If input is finished and $ is received then pop a 0 from stack.



Part-C

1. Design a non deterministic PDA for accepting the language L = {$a^ib^jc^kd^l$: i==k or j==l, i>=1}, ie.,

L = {abcd, aabccd, aaabcccd, abbcdd, aabbccdd, aabbbccddd, ...} In each string, the number of a's are followed by any number of b's and b's are followed by the number of c's equal to the number of a's and c's are followed by number of d's equal number of b's.

Explanation –

Here, we need to maintain the order of a's, b's, c's and d's.That is, all the a's are coming first then all the b's are coming and then all the c's are coming then all the d's are coming . Thus, we need a stack along with the state diagram. The count of a's and b's is maintained by the stack.We will take 2 stack alphabets:

Approach used in the construction of PDA –

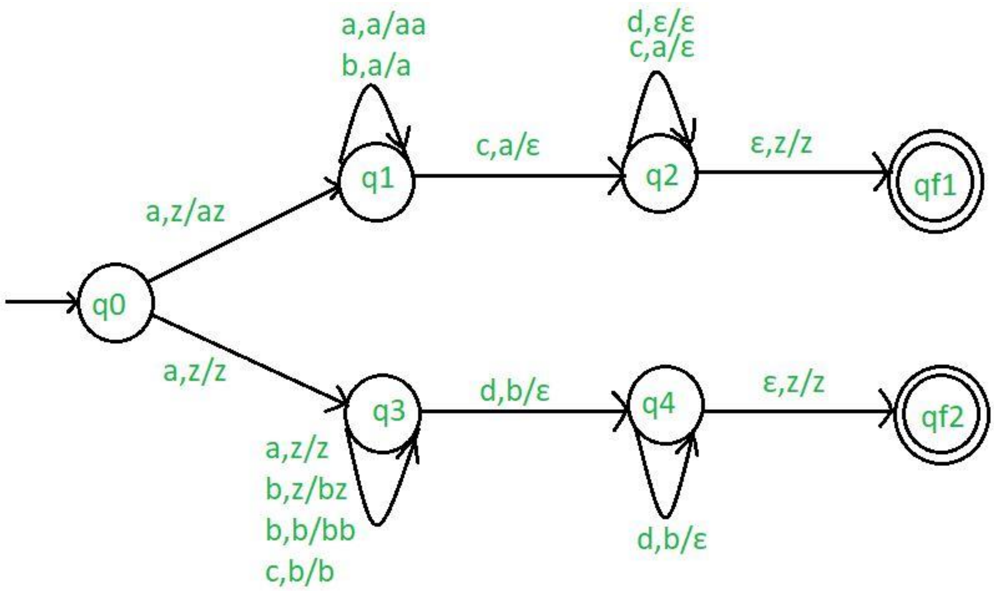
In designing a NPDA, for every a', 'b', 'c' and 'd' will comes in proper order.

- For i==k : Whenever 'a' comes, push it in stack and if 'a' comes again then also push it in the stack.After that, if 'b' comes not do any operation. After that, when 'c' comes then pop 'a' from the stack each time.After that, if 'd' comes not do any operation.
- For j==l : Whenever 'a' comes, not do any operation.After that, if 'b' comes push it in stack and if 'b' comes again then also push it in the stack. After that, when 'c' comes not do any operation.After that, if 'd' comes then pop 'b' from the stack each time.

So that the stack becomes empty.If stack is empty then we can say that the string is accepted by the PDA.

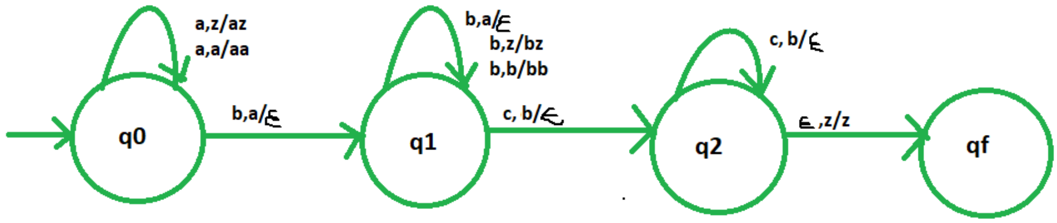
Construct Stack transition functions

Required NPDA



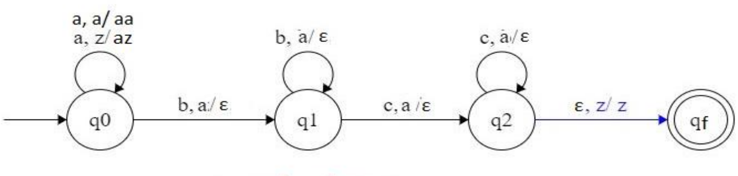
Required PDA

3.Design a non deterministic PDA for accepting the language $L = \{a^m b^{(m+n)} c^n \mid m,n \geq 1\}$



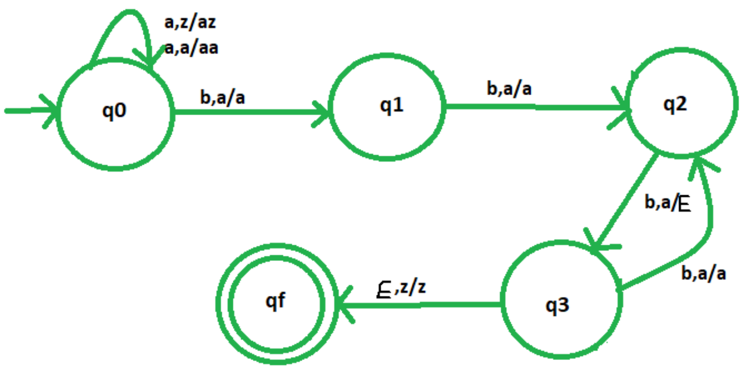
Required PDA

4. Design a non deterministic PDA for accepting the language $L=a^{(m+n)}b^m c^n$,m,n>=1



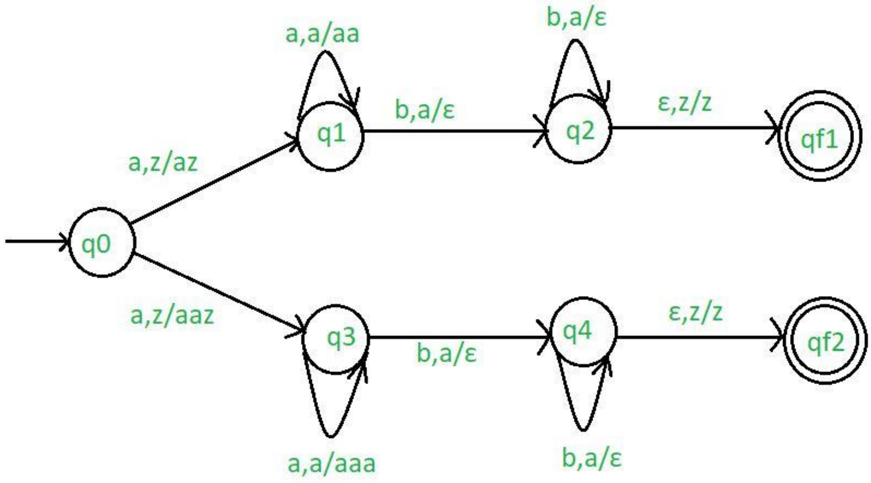
Required PDA

4. NPDA for accepting the language $L = \{a^m b^{(2m+1)} \mid m \geq 1\}$ Design a non deterministic PDA for accepting the language



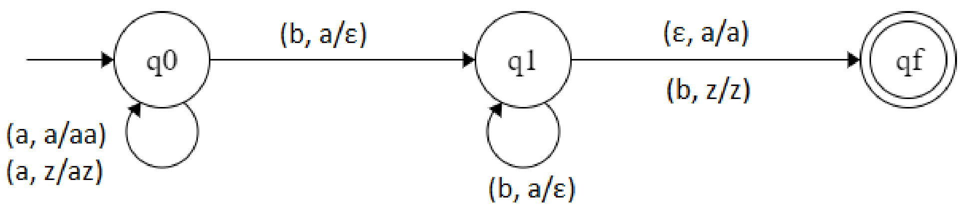
Required PDA

5. Design a non deterministic PDA for accepting the language NPDA for accepting the language L = {$a^n b^{(2n)}$ | n>=1} U {$a^n b^n$ | n>=1}



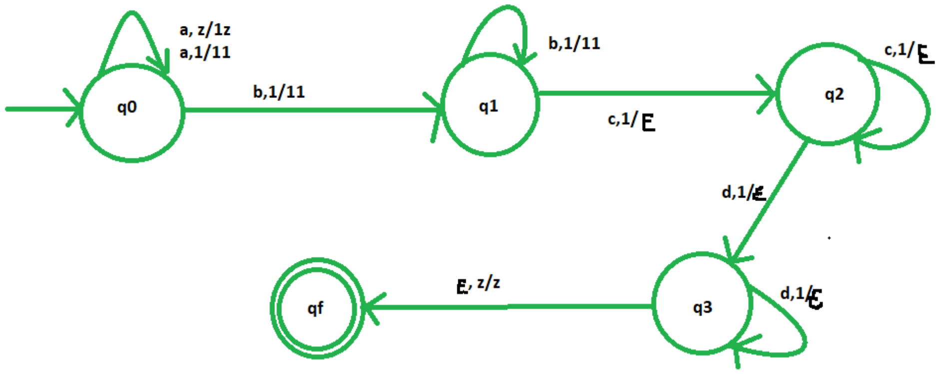
Required PDA

6. Design a non deterministic PDA for accepting the language L = {$a^n b^m$ | n,m ≥ 1 and n ≠ m}



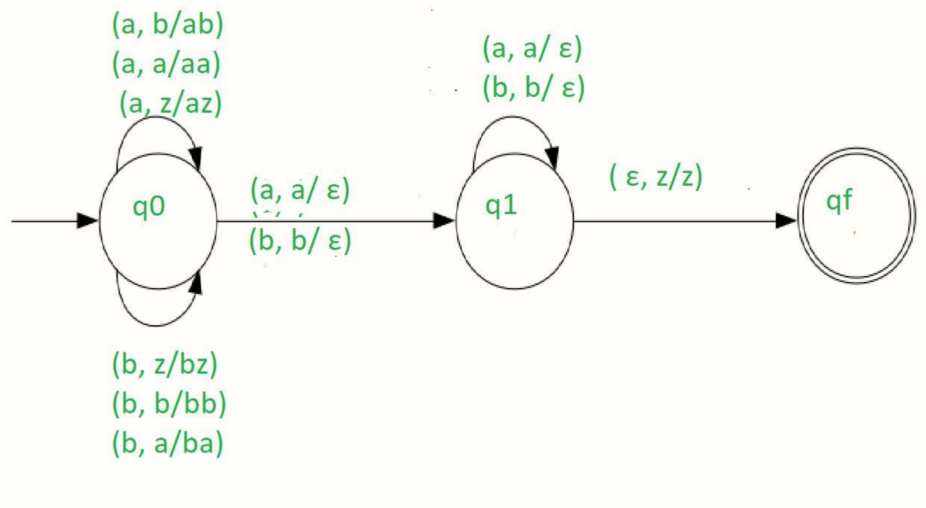
Required PDA

7. Design a non deterministic PDA for accepting the language L = {$a^m b^n c^p d^q$ | m+n=p+q ; m,n,p,q>=1}



Required PDA

8. Design a non deterministic PDA for accepting the language L = {wwR | w ∈ (a,b)*}



Required NPDA