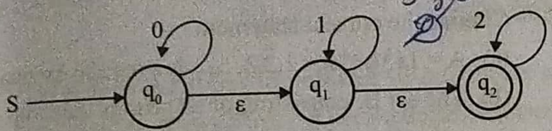
3 Theory of Computation

1. Regular Languages

01. What are final states of the DFA generated from the following NFA?



(a) q_0, q_1, q_2 (b) $[q_0, q_1], [q_0, q_2], []$
(c) $q_0, [q_1, q_2]$ (d) $[q_0, q_1], q_2$

[ISRO - 2013]

02. The number of states required by a Finite State Machine, to simulate the behaviour of a computer with a memory capable of storing 'm' words, with each word being 'n' bits long is

(a) $m \times 2^n$ (b) 2^{m+n}
(c) 2^{mn} (d) m+n

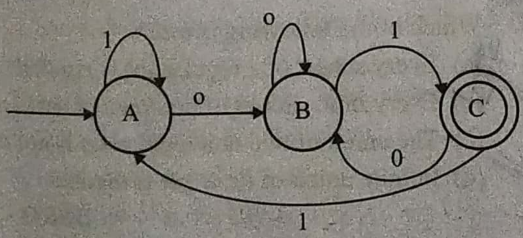
[ISRO - 2014]

03. How many states are there in a minimum state deterministic finite automaton accepting the language L = {w | w ∈ (0, 1)*, number of 0's is divisible by 2 and number of 1's is divisible by 5, respectively}?

(a) 7 (b) 9 (c) 10 (d) 11

[ISRO - 2014]

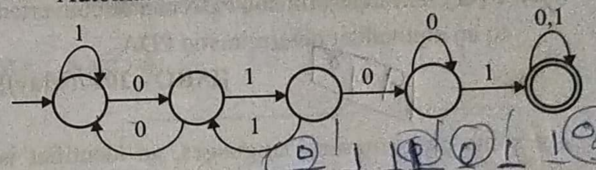
04. The following Finite Automaton recognizes which of the given languages?



(a) {1, 0} * {0 1} (b) {1, 0} * {1}
(c) {1}{1, 0} * {1} (d) 1 * 0* {0, 1}

[ISRO - 2014]

05. Consider the following Deterministic Finite Automaton M.



Let S denote the set of eight bit strings whose second, third,sixth and seventh bits are 1. The number of strings in S that are accepted by M is

(a) 0 (b) 1 (c) 2 (d) 3

[ISRO - 2014]

06. Let R_1 and R_2 be regular sets defined over the alphabet, then

(a) $R_1 \cap R_2$ is not regular
(b) $R_1 \cup R_2$ is not regular
(c) $\Sigma^* - R_1$ is regular
(d) R_1 * is not regular

[ISRO - 2015]

07. An FSM (Finite State Machine) can be considered to be a Turning Machine of finite tape length

(a) Without rewinding capability and unidirectional tape movement
(b) Rewinding capability and unidirectional tape movement.
(c) Without rewinding capability and bidirectional tape movement
(d) Rewinding capability and bidirectional tape movement.

[ISRO - 2016]

08. Let L = {w ∈ (0 +1)*| w has even number of 1s}, i.e. L is the set of all bit strings with even number of 1s. Which one of the regular expression below represents L?

(a) (0* 10*1)* (b) 0*(10*10*)*
(c) 0*(10*1*)*0* (d) 0*1(10*1)*10*

[ISRO - 2016]

09. Which one of the following is FALSE?
(a) There is a unique minimal DFA for every regular language
(b) Every NFA can be converted to an equivalent PDA
(c) Compliment of every context-free language is recursive
(d) Every non-deterministic PDA can be converted to an equivalent deterministic PDA
[ISRO - 2017(May)]

10. In some programming languages, an identifier is permitted to be a letter followed by any number of letters or digits. If L and D denotes the set of letters and digit respectively. Which of the following expression defines an identifier?
(a) (L + D)*
(b) (L.D)*
(c) L(L + D)*
(d) L(L.D)*
[ISRO - 2017(May)]

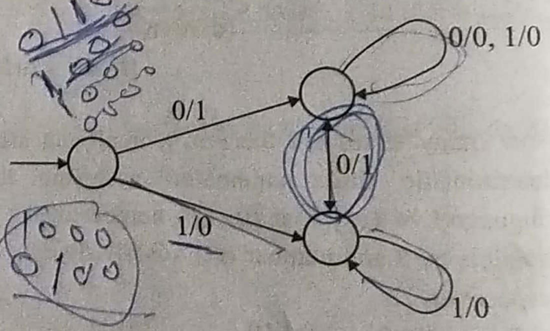
11. For $\Sigma = \{a, b\}$ the regular expression $r = (aa)^*(bb)^*b$ denotes
(a) Set of strings with 2 a's and 2 b's
(b) Set of strings with 2 a's 2 b's followed by b
(c) Set of strings with 2 a's followed by b's which is a multiple of 3
(d) Set of strings with even number of a's followed by odd number of b's
[ISRO - 2017(Dec)]

12. A particular BNF definition for a "word" is given by the following rules:
<word> :: = <letter> | <letter><charpair> | <letter><intpair>
<charpair> :: = <letter><letter> |<charpair><letter> <letter>
<intpair> :: = <integer><integer>|<intpair><integer><integer>
<letter> :: = a | b | c | ..... | y | z
<integer> :: = 0 | 1 | 2 | .... | 9

Which of the following lexical entries can be derived from <word>?
I. pick   II. picks   III. c44
(a) I, II and III
(b) I and II only
(c) I and III only
(d) II and III only
[ISRO - 2018]

13. Choose the correct statement
(a) $A = \{a^n b^n \mid n = 1,2,3, ...\}$ is a regular language
(b) The set B, consisting of all strings made up of only a's and b's having equal number of a's and b's defines a regular language
(c) $L(A^* B) \cap B$ gives the set A
(d) None of the above
[ISRO - 2018]

14. The FSM (Finite State Machine) machine pictured in the figure below



(a) Complements a given bit pattern
(b) Finds 2's complement of a given bit pattern
(c) Increments a given bit pattern by 1
(d) Changes the sign bit
[ISRO - 2019]

15. Which of the following is true?
(a) Every subset of a regular set is regular
(b) Every finite subset of non regular set is regular
(c) The union of two non regular set is not regular
(d) Infinite union of finite set is regular
[ISRO - 2020]

16. The language which is generated by the grammar $S \rightarrow aSa \mid bSb \mid a \mid b$ over the alphabet {a, b} is the set of
(a) Strings that begin and end with the same symbol
(b) All odd and even length palindromes
(c) All odd length palindromes
(d) All even length palindromes
[ISRO - 2020]

17. Which of the following classes of languages can validate an IPv4 address in dotted decimal format? It is to be ensured that the decimal values lie between 0 and 255.
(a) RE and higher finite
(b) CFG and higher
(c) CSG and higher
(d) Recursively enumerable language
[ISRO - 2020]

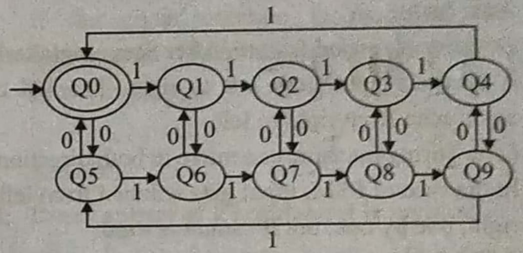
18. Minimum number of states required in DFA accepting binary strings not ending in "101" is
(a) 3 (b) 4
(c) 5 (d) 6
[ISRO - 2020]

KEY & Detailed Solutions				
01. (a)	02. (c)	03. (c)	04. (a)	05. (c)
06. (c)	07. (a)	08. (b)	09. (d)	10. (c)
11. (d)	12. (d)	13. (d)	14. (*)	15. (b)
16. (c)	17. (a)	18. (b)		

01. Ans: (a)
Sol: q_2 has a double rounded circle. Hence it is a final state.
The states which contain final states in their $\in$ - closure are also Final states.
$\in$ - closure of $q_0 = \{q_0, q_1, q_2\}$
$\in$ - closure of $q_1 = \{q_1, q_2\}$
So q_0, q_1, q_2 all are final states

02. Ans: (c)
Sol: If we have k bits, we can have 2^k different patterns. (each position can be filled in 2 ways, 0 or 1)
Given
Number of words = m
Length of the word in bits = n
Total Number of bits = (number of words)*(bits per word) = mn
So, number of states required = $2^{\text{Total number of bits}} = 2^{mn}$

03. Ans: (c)



04. Ans: (a)
Sol: The given automata accepts the strings that are ending with 01.
The regular expression for above Language is (0+1)*01 or {0,1}*01

05. Ans: (c)
Sol: The number of 8 bit strings in which 4 bits (2^{nd}, 3^{rd}, 6^{th}, 7^{th}) are set is $2^4 = 16$

1111 0110	We can see that strings starting with 1 & having 6th, 7th bits as 1 cannot be accepted by DFA given.
1111 0111	
1111 1110	
1111 1111	
1110 0110	
1110 0111	
1110 1110	
1110 1111	
0111 0110	These two strings can be accepted by the given DFA.
0111 0111	
0111 1110	Not accepted
0111 1111	Not accepted
0110 0110	Not accepted
0110 0111	Not accepted
0110 1110	Not accepted
0110 1111	Not accepted

06. Ans: (c)

Sol: $R_1 \cap R_2$ is regular,because Regular languages are closed under Intersection.

$R_1 \cup R_2$ is regular, because Regular languages are closed under Union.

$\Sigma^* - R_1$ is regular,because Regular languages are closed under Complementation.

R_1^* is regular, because Regular languages are closed under Kleene Closure.

07. Ans: (a)

Sol: Because of rewind feature, After Scanning(altering if needed) input from left to right, The TM can again scan from right to left.

(The Turing machine can move in both directions).

In the case of FSM, input gets scanned from left to right, one by one, but It cannot rewind.

A FSM(finite state machine) can be considered to be a Turing machine of finite tape length, which has no rewinding capability and with unidirectional tape movement.

08. Ans: (b)

Sol: Option (a) is wrong

$(0^*10^*1)^*$

All the strings generated by the above regular expression are ending with 1.

So, It cannot cover strings that have an even number of 1s and ending with 0.

Option (c) is wrong

$0^*(10^*1^*)^*0^*$

It can accept a string with an odd number of 1's.

Example: 01011.

Option (d) is wrong

$0^*1(10^*1)^*10^*$

11 is the minimum length string that can be generated by the above regular expression.

It cannot generate any string that has no 1s. (0, 00, 000….).

But all such strings are allowed in Language.

$0^*(10^*10^*)^*$ is the appropriate regular expression for the given Language.

09. Ans: (d)

Sol: There is a unique minimal DFA for every regul[ar] language, True.

For a given regular language, though Multiple NF[A] are possible, after converting and minimizing, [w]e get a unique minimal DFA.

Every NFA can be converted to an equivale[nt] PDA, True.

Because,The set of Languages that can be accept[ed] by a NFA are a proper subset of Languages that a[re] accepted by a PDA.

Complement of every context-free language [is] recursive,True

CFL is a proper subset of recursive languages a[nd] recursive languages are closed under complemen[t].

Every non-deterministic PDA can be converte[d] to an equivalent deterministic PDA,False

NDPA is more powerful than DPDA, so they a[re] not equivalent. The set of Languages that ca[n] be accepted by a DPDA are a proper subset [of] Languages that are accepted by NDPDA.

10. Ans: (c)

Sol: $(L + D)^*$

It is not satisfying the constraint that an identifi[er] should start with a letter.

$(L.D)^*$

It is not satisfying the constraint that an identifi[er] should start with a letter.

$L(L + D)^*$

It is the appropriate regular expression for the give[n] constraints.

$L(L.D)^*$

From the second letter onwards,there can be a lett[er] or digit, but this expression forces the strings t[o] have an equal number of letters and digits after th[e] first letter.

11. Ans: (d)

Sol: Set of strings with 2 a's and 2 b's

The given regular expression can represent aaaabbb, which is not allowed in the above Language.

Set of strings with 2 a's 2 b's followed by b

The given regular expression can represent aaaabbb, which is not allowed in the above Language.

Set of strings with 2 a's followed by b's which is a multiple of 3

The given regular expression can represent aaaabbb, which is not allowed in the above Language.

Set of strings with even number of a's followed by odd number of b is the language represented by given regular expression

12. Ans: (d)

Sol: picks

<word> = <letter> **<charpair>**

= <letter> <charpair> <letter> **<letter>**

= <letter> <charpair> **<letter>**s

= <letter> **<charpair>**ks

= <letter> <letter>**<letter>**ks

= <letter>**<letter>**cks

= **<letter>**icks

= picks

c44

<word> = <letter>**<intpair>**

= <letter> <integer>**<integer>**

=<letter> **<integer>**4

=**<letter>** 44

=c44

pick cannot be generated with given Rules.

13. Ans: (d)

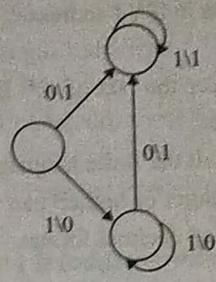
Sol: $\{a^n b^n \mid n = 1,2,3, ...\}$ is a DCFL but not regular, hence A is False.

The set B, consisting of all strings made up of only a's and b's having equal number of a's and b's defines a DCFL but not regular, hence (b) is False.

L(A * B) $\cap$ B gives the set B(not A), hence (c) is false.

14. Ans: None of the options

Sol:



If the given diagram is modified like this, **Increments given bit pattern by 1** would have been the answer.

15. Ans: (b)

Sol: Every subset of a regular set is regular

False. (a+b)* is regular, but its subset $\{a^n b^n \mid n>1\}$ is not regular.

Every finite subset of non regular set is regular

True, Every Finite Language is regular.

The union of two non regular set is not regular

False. Let us consider Union of a non-regular Language L and its complement L^c, result is Σ^* (regular).

Infinite union of finite set is regular

False.

$\{ab\} \cup \{a^2 b^2\} \cup \{a^3 b^3\} \cup ....= \{a^n b^n \mid n>1\}$, which is context free but not regular.

16. Ans: (c)

Sol: $S \rightarrow aSa|bSb|a|b|\varepsilon$ accepts all palindromes over {a,b}

$S \rightarrow aSa|bSb|a|b$ accepts all odd length palindromes over {a,b}

$S \rightarrow aSa|bSb|\varepsilon$ accepts all even length palindromes over {a,b}

17. Ans: (a)

Sol: We have 32 bits in IPV4 addresses, so 2^{32} addresses are possible.

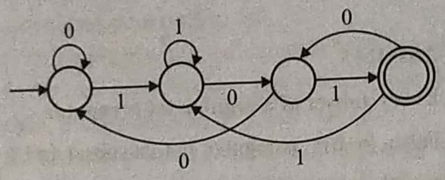
We can consider the set of 2^{32} IP addresses as a finite language.

We know that all the finite languages are regular.

Regular Languages or higher can validate an IPv4 address in dotted decimal format.

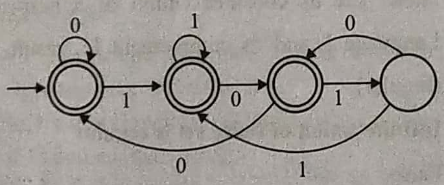
18. Ans: (b)

Sol: DFA for accepting strings ending with "101"



Now, complement of the above DFA, resultant DFA accepts binary strings not ending in "101"

(Change Final states to Non-Final and Non-final states to Final)



2. Context Free Languages

01. Consider the following grammar.

$S \rightarrow AB$

$A \rightarrow a$

~~$A \rightarrow BaB$~~

$B \rightarrow bbA$

Which of the following statements is FALSE?

(a) The length of every string produced by this grammar is even.

(b) No string produced by this grammar has three consecutive a's

(c) The length of substring produced by B always odd.

(d) No string produced by this grammar has four consecutive b's

[ISRO - 2014]

02. Consider the following statements about the context-free grammar:

G = {S→SS, S→ab, S→ba, S→∈}

I. G is ambiguous

II. G produces all strings with equal number of a's and b's

III. G can be accepted by a deterministic PDA

Which combinations below expresses all the true statements about G?

(a) I only (b) I and III only

(c) II and III only (d) I, II and III

[ISRO - 2016]

03. What is the highest type number that can be assigned to this following grammar :

S→ Aa, A→ Ba, B → abc

(a) Type 0 (b) Type 1

(c) Type 2 (d) Type 3

[ISRO - 2016]

04. Consider the grammar with productions

$S \rightarrow aSb \mid SS \mid \varepsilon$

This grammar is

(a) not context-free, not linear

(b) not context-free, linear

(c) context-free, not linear

(d) context free, linear

[ISRO - 2017(Dec)]

05. Identify the lan... grammar

$S \rightarrow AB$

$A \rightarrow aAb \mid \varepsilon$

$B \rightarrow bB \mid b$

(a) $\{a^m b^n \mid n \geq m$...

(c) $\{a^m b^n \mid n >$...

06. CFG (Context...

(a) Union

(c) Kleene sta...

07. A CFG (Co... in Chomsky... productions... G be a CFG... of length x, ...

(a) 2x – 1

(c) 2x + 1

08. Context free...

(a) union, in...

(b) union, k...

(c) intersect...

(d) complem...

KEY
01. (d)
05. (d)

Scanned with CamScanner

05. Identify the language generated by the following grammar
$S \rightarrow AB$
$A \rightarrow aAb|\varepsilon$
$B \rightarrow bB|b$

A B
ε b b

(a) $\{a^m b^n | n \geq m, m > 0\}$ (b) $\{a^m b^n | n \geq m, m \geq 0\}$
(c) $\{a^m b^n | n > m, m > 0\}$ (d) $\{a^m b^m | n > m, m \geq 0\}$
[ISRO - 2017(Dec)]

06. CFG (Context Free Grammar) is not closed under
(a) Union (b) Complementation
(c) Kleene star (d) Product
[ISRO - 2018]

07. A CFG (Context Free Grammar) is said to be in Chomsky Normal Form (CNF), if all the productions are of the form $A \rightarrow BC$ or $A \rightarrow a$. Let G be a CFG in CNF. To derive a string of terminals of length x, the number of products to be used is
(a) $2x - 1$ (b) $2x$
(c) $2x + 1$ (d) 2^x
[ISRO - 2018]

08. Context free languages are closed under
(a) union, intersection
(b) union, kleene closure
(c) intersection, complement
(d) complement, kleene closure **[ISRO - 2020]**

KEY & Detailed Solutions			
01. (d)	02. (b)	03. (d)	04. (c)
05. (d)	06. (b)	07. (a)	08. (b)

01. Ans: (d)
Sol: $A \rightarrow a$
string of Length 1 always
$B \rightarrow bbA$
2 bs (even) +1A (odd) ====>odd
$A \rightarrow BaB$
1 a(odd) + 2 Bs(even)====>odd
$S \rightarrow AB$
1A (odd) + 1 B (odd)=====>even
The length of substring produced by (B) is always odd. Options (c) is True.
The length of every string produced by this grammar is always even. Hence Option (a) is True.
Strings can have a maximum of 2 consecutive a's. Hence Option (b) is True.
$S \rightarrow AB$
$S \rightarrow aB$
$S \rightarrow abbA$
$S \rightarrow abbBaB$
$S \rightarrow abbbbAabbA$
$S \rightarrow abbbbaabba$
Given Grammar can produce a string which has 4 consecutive b's. Hence, Option(d) is wrong.

02. Ans: (b)
Sol: I. G is ambiguous
True.
Example:
The string "ab" has multiple derivation trees like
Derivation1: $S \rightarrow SS \rightarrow abS \rightarrow ab$
Derivation2: $S \rightarrow ab$.
II. G produces all strings with equal number of a's and b's
False.
G does not produce all strings with equal no. of a`s and b's.
(Example: "aabb" cannot be generated).
III. G can be accepted by a deterministic PDA
True.
The given grammar G generates the language (ab+ba)*,
which is Regular and therefore also a DCFL. So, a D-PDA can be designed for G.

03. Ans: (d)

Sol: Given Grammar is Left Linear Grammar. So, Type-3 is satisfied. (highest)

The Language produced by the grammar has only one string

L={abcaa}

04. Ans: (c)

Sol: Context free grammars have rules of the form

$P \rightarrow s$

where P is a non-terminal and s is a string (can be empty) made up of terminals and non-terminals.

$S \rightarrow aSb|SS|\varepsilon$

No production is violating the rules of CFG, so the given grammar is context free.

$S \rightarrow SS$

This production is making the grammar both Left linear and right Linear simultaneously.

To be linear, The grammar must be either left linear or right linear but not both.

05. Ans: (d)

Sol: $B \rightarrow bB|b \Rightarrow \{b^p|p>=1\}$ or $\{b^p|p>0\}$

$A \rightarrow aAb|\varepsilon \Rightarrow \{a^m b^m|\ m>=0\}$

$S \rightarrow AB \Rightarrow \{a^m b^m b^p|\ m>=0, p>0\}$

$= \{a^m b^{m+p}|\ m>=0, p>0\}$

since p >0, m+p>m

$\{a^m b^{m+p}|\ m>=0, m+p>m\}$

If we replace m+p with n

$L = \{a^m b^n|\ n>m, m>=0\}$

06. Ans: (b)

Sol: Context free languages are not closed under Complement operation.

operation	Regular	DCFL	CFL	CSL	Recursive	RE
Union	yes	no	yes	yes	yes	yes
Intersection	yes	no	no	yes	yes	yes
Complement	yes	yes	no	yes	yes	no
Concatenation	yes	no	yes	yes	yes	yes
Kleene star	yes	no	yes	yes	yes	yes
Homomorphism	yes	no	yes	no	no	yes
ε-free Homomorphism	yes	no	yes	yes	yes	yes
Substitution (ε-free)	yes	no	yes	yes	no	yes
Inverse Homomorphism	yes	yes	yes	yes	yes	yes
Reverse	yes	no	yes	yes	yes	yes
Intersection with a regular language	yes	yes	yes	yes	yes	yes

07. Ans: (a)

Sol: A context free grammar (CFG) is said to be in Chomsky Normal Form (CNF) if all production rules are in the form:

$S \rightarrow AB$

$A \rightarrow a$

A non-terminal symbol generate a terminal Symbol

(or)

A non-terminal symbol generates two non terminals symbols adjacently.

For generating string w of length 'x' requires '2x−1' production or steps in CNF because in CNF at every step only 1 terminal can replace a variable whereas, In GNF (greibach normal form), every production has the form $A \rightarrow a A_1 A_2 ... A_k$ or $S \rightarrow \epsilon$

So in each derivation one terminal symbol gets used. So in n steps the whole input gets parsed.

08. Ans: (b)

Sol: Context-free languages are not closed under intersection and complementation, so none of these two operations should be included in the answer.

Opeation	Regular	DCFL	CFL	CSL	Recursive	RE
Union	yes	no	yes	yes	yes	yes
Intersection	yes	no	no	yes	yes	yes
Complement	yes	yes	no	yes	yes	no
Concatanation	yes	no	yes	yes	yes	yes
Kleene star	yes	no	yes	yes	yes	yes
Homomorphism	yes	no	yes	no	no	yes
ϵ-free Homomorphism	yes	no	yes	yes	yes	yes
Substitution (ϵ-free)	yes	no	yes	yes	no	yes
Inverse Homo morphism	yes	yes	yes	yes	yes	yes
Reverse	yes	no	yes	yes	yes	yes
Intersection with a regular language	yes	yes	yes	yes	yes	yes

3. Recursive Enumerable Languages

01. A problem whose language is recursive is called?
(a) Unified problem (b) Boolean function
(c) Recursive problem (d) Decidable
[ISRO - 2011]

02. Which of the following is FALSE with respect to possible outcomes of executing a Turning Machine over a given input?
(a) It may halt and accept the input
(b) It may halt by changing the input
(c) It may halt and reject the input
(d) It may never halt **[ISRO - 2014]**

03. If L and $\overline{L}$ are recursively enumerable, then L is
(a) Regular (b) Context-free
(c) Context-sensitive (d) Recursive
[ISRO - 2016]

04. $S \rightarrow aSa \mid bSb \mid a \mid b$
The language generated by the above grammar over the alphabets {a, b} is the set of
(a) All palindromes
(b) All odd length palindromes
(c) Strings that begin and end with same symbol
(d) All even length palindromes
[ISRO - 2016]

05. Given the following statements
S1: Every context-sensitive language L is recursive
S2 : There exists a recursive language that is not context-sensitive

Which statements are true?
(a) Only S1 is correct
(b) Only S2 is correct
(c) Both S1 and S2 are not correct
(d) Both S1 and S2 are correct
[ISRO - 2017(May)]

06. If L and P are two recursively enumerable languages, then they are not closed under
(a) Kleene Star L* of L (b) Intersection $L \cap P$
(c) Union $L \cup P$ (d) Set difference
[ISRO - 2017(May)]

07. Let L_1 be regular language, L_2 be a deterministic context free language and L_3 a recursively enumerable language, but not recursive. Which one of the following statements is false?
(a) $L_3 \cap L_1$ is recursive
(b) $L_1 \cap L_2 \cap L_3$ is recursively enumerable
(c) $L_1 \cup L_2$ is context free
(d) $L_1 \cap L_2$ is context free
[ISRO - 2017(Dec)]

08. Let $L = \{a^p \mid p \text{ is a prime}\}$. Then which of the following is true
(a) It is not accepted by a Turning Machine
(b) It is regular but not context free
(c) It is context free but not regular
(d) It is neither regular nor context free, but accepted by a Turning Machine
[ISRO - 2017(Dec)]

KEY & Detailed Solutions			
01. (d)	02. (b)	03. (d)	04. (b)
05. (d)	06. (d)	07. (a)	08. (d)

01. Ans: (d)

Sol: Recursive languages are Turing Decidable, whereas Recursively Enumerable languages are Turing Recognizable.

A problem whose language is recursion is called Decidable.

A problem whose language is recursively enumerable is called Recognizable.

02. Ans: (b)

Sol: If the input string belongs to a recursive language TM may halt and accept the input or it may halt and reject the input.

If the input belongs to a recursively enumerable language, then either TM may halt and accept the input or it may never halt.

It cannot halt by changing the input, because TM just transits from one state to another state on a given input. It only does the transitions that it is supposed to do on the given input as per its definition.

03. Ans: (d)

Sol: L is recursively enumerable means a TM accepts all strings in L.

L^c is recursively enumerable means a TM accepts all strings in L^c.

So, we can always decide if a string is in L or not making L recursive.

04. Ans: (b)

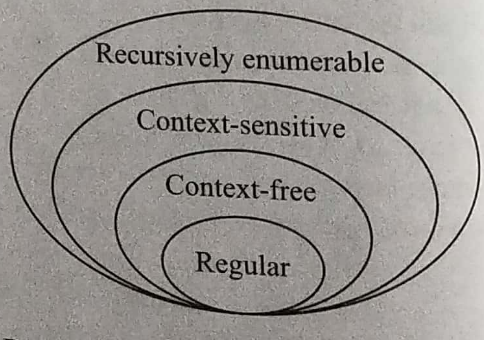
Sol: $S \to aSa \mid bSb \mid a \mid b \mid \varepsilon$ accepts all palindromes over $\{a, b\}$

$S \to aSa \mid bSb \mid a \mid b$ accepts all odd length palindromes over $\{a, b\}$

$S \to aSa \mid bSb \mid \varepsilon$ accepts all even length palindromes over $\{a, b\}$

05. Ans: (d)

Sol:



Both are correct.

Because the set of context sensitive languages is proper subset of recursive languages.

06. Ans: (d)

Sol: We know that, Set difference = A−B = A∩B^C.
Since, recursively enumerable languages are closed under intersection but not under complement, set difference of these two languages is not closed.

Operation	Regular	DCFL	CFL	CSL	Recursive	RE
Union	yes	no	yes	yes	yes	yes
Intersection	yes	no	no	yes	yes	yes
Complement	yes	yes	no	yes	yes	no
Concatena-tion	yes	no	yes	yes	yes	yes
Kleene star	yes	no	yes	yes	yes	yes
Homomor-phism	yes	no	yes	no	no	yes
∈-free Homo-morphism	yes	no	yes	yes	yes	yes
Substitution (∈-free)	yes	no	yes	yes	no	yes
Inverse Ho-momorphism	yes	yes	yes	yes	yes	yes
Reverse	yes	no	yes	yes	yes	yes
Intersection with a regular language	yes	yes	yes	yes	yes	yes

07. Ans: (a)

Sol: L3 ∩ L1 is recursive.(May or May not be)
L_1 is regular, Since every regular language is recursively enumerable, we can consider L_1 to be recursively enumerable. L_3 is recursively enumerable anyways.
$L_3 \cap L_1$ is recursively enumerable but may or may not be recursive.
$L_1 \cap L_2 \cap L_3$ is recursively enumerable
L_1 is regular, Since every regular language is recursively enumerable, we can consider L_1 to be recursively enumerable.
L_2 is DCFL, Since every DCFL is recursively enumerable, we can consider. L_2 to be recursively enumerable. L_3 is recursively enumerable anyways.
Since Recursively Enumerable Languages are closed under Intersection,
$L_1 \cap L_2 \cap L_3$ is recursively enumerable
$L_1 \cup L_2$ is context free
L_1 is regular, Since every regular language is CFL, we can consider L_1 to be CFL.
L_2 is DCFL,Since every DCFL is CFL, we can consider L_2 to be CFL.
Since CFLs are closed under Union, $L_1 \cup L_2$ is context free
$L_1 \cap L_2$ is context free
L_1 is regular,Since every regular language is DCFL, we can consider L_1 to be CFL.
L_2 is DCFL anyways.
Since DCFLs are closed under Intersection, $L_1 \cap L_2$ DCFL.
Since every DCFL is CFL, we can consider $L_1 \cap L_2$ to be CFL.

08. Ans: (d)

Sol: $L = \{a^p \mid p \text{ is a prime}\}$.
The above language is context sensitive but not context free.
Hence we can design a LBA, TM but PDA cannot be designed.