GATE

Computer Science & IT

(Previous Years Solved Papers 1987-1994)

Contents

MADE EASY PUBLICATIONS

Discrete & Engineering Mathematics

UNIT

I

CONTENTS

Discrete and Engineering Mathematics

1. Mathematical Logic

1.1 Indicate which of the following well-formed formula are valid:
(a) $((P \Rightarrow Q) \wedge (Q \Rightarrow R)) \Rightarrow (P \Rightarrow R)$
(b) $(P \Rightarrow Q) \Rightarrow (\neg P \Rightarrow \neg Q)$
(c) $(P \wedge (\neg P V \neg Q)) \Rightarrow Q$
(d) $((P \Rightarrow R) \vee (Q \Rightarrow R)) \Rightarrow ((P \vee Q)) \Rightarrow R)$

[1990 : 2 Marks]

1.2 Which of the following predicate calculus statements is/are valid
(a) $(\forall x)\ P(x) \vee (\forall x)\ Q(x) \rightarrow (\forall x)\ \{P(x) \vee Q(x)\}$
(b) $(\exists x)\ P(x) \wedge (\exists x)\ Q(x) \rightarrow (\exists x)\ \{P(x) \wedge Q(x)\}$
(c) $(\exists x)\ \{P(x) \vee Q(x)\} \rightarrow (\forall x)\ P(x) \vee (\forall x)\ Q(x)$
(d) $(\exists x)\ \{P(x) \vee Q(x)\} \rightarrow \exists x\ P(x) \vee \forall x\ Q(x)$

[1992 : 1 Mark]

1.3 Which of the following is/are tautology:
(a) $(a \vee b) \rightarrow (b \wedge c)$ (b) $(a \wedge b) \rightarrow (b \vee c)$
(c) $(a \vee b) \rightarrow (b \rightarrow c)$ (d) $(a \rightarrow b) \rightarrow (b \rightarrow c)$

[1992 : 1 Mark]

1.4 The proposition $p \wedge (\sim p \vee q)$ is
(a) A tautology (b) $\Leftrightarrow (p \wedge q)$
(c) $\Leftrightarrow (p \vee q)$ (d) A contradiction

[1993 : 1 Mark]

1.5 Let p and q be propositions. Using only the truth table decide whether $p \Leftrightarrow q$ does not imply $p \rightarrow \neg q$ is true or false.

[1994 : 2 Marks]

2. Set Theory and Algebra

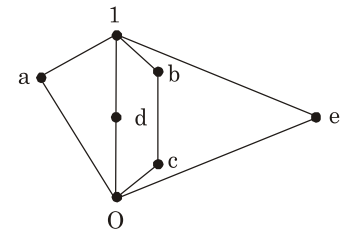
2.1 State whether the following statement are TRUE or FALSE:
The union of two equivalence relations is also an equivalence relation.

[1987 : 1 Mark]

2.2 (a) How many binary relations are there on a set A with n elements?
(b) How many one–to–one functions are there from a set A with n elements onto itself.

[1987 : 1 Mark]

2.3 The complement(s) of the element 'a' in the lattice shown in figure is (are) ......



[1988 : 2 Marks]

2.4 The transitive closure of the relation {(1, 2) (2, 3) (3, 4) (5, 4)} on the set A = {1, 2, 3, 4, 5} is ____

[1989 : 2 Marks]

2.5 Let S be an infinite set and $S_1, S_2, \ldots S_n$ be sets such that $S_1 \cup S_2 \cup \ldots\ldots \cup S_n = S$. Then
(a) At least one of the sets S_i is a finite set
(b) Not more than one of the sets S_i can be finite
(c) At least one of sets S_i is infinite
(d) Not more than one the sets S_i is an a infinite set.

[1993 : 1 Mark]

2.6 Let A be a finite set of size n. The number of elements in the power set of A × A is
(a) 2^{2^n} (b) 2^{n^2}
(c) 2n (d) 2^n

[1993 : 1 Mark]

2.7 Some group (G, o) is known to be abelian. Then, which one of the following is true for G?
(a) $g = g^{-1}$ for every $g \in G$.
(b) $g = g^2$ for every $g \in G$
(c) $(goh)^2 = g^2 \text{ o } h^2$ for every $g, h \in G$.
(d) G is of finite order

[1994 : 2 Marks]

3. Combinatorics

3.1 (a) Solve the recurrence equations:
$T(n) = T(n - 1) + n$
$T(1) = 1$
(b) What is the generating function G(z) for the sequence of Fibonacci numbers?

[1987 : 2 Marks]

3.2 Solve the recurrence equations:

$$T(n) = T\left(\frac{n}{2}\right) + 1$$

$$T(1) = 1$$

[1988 : 2 Marks]

3.3 How many substrings can be formed from a character string of length n?

[1989 : 2 Marks]

3.4 The number of binary strings of n zeros and k ones that no two ones are adjacent is

(a) ${}^{n+1}C_k$ (b) ${}^{n}C_k$

(c) ${}^{n}C_{k+1}$ (d) None of these

[1990 : 1 Mark]

3.5 The number of substrings (of all lengths inclusive) that can be formed from a character string of length n is

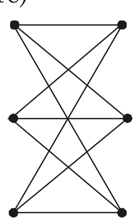
(a) n (b) n^2

(c) $\frac{n(n-1)}{2}$ (d) $\frac{n(n+1)}{2}$

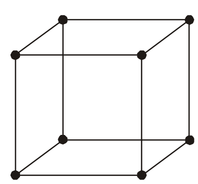
[1994 : 2 Marks]

4. Graph Theory

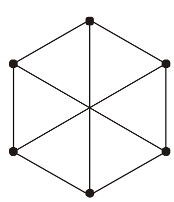
4.1 Which of the following graphs is/are planar? (see Figure)



G1



G2



G3

[1989 : 2 Marks]

(a) G1 only (b) G1 and G2

(c) G2 only (d) G2 and G3

4.2 A graph is planar if and only if,

(a) If does not contain subgraphs homeomorphic to K_5 and $K_{3,3}$.

(b) It does not contain subgraphs isomorphic to K_5 or $K_{3,3}$.

(c) It does not contain subgraphs isomorphic to K_5 and $K_{3,3}$.

(d) It does not contain subgraph homeomorphic to K_5 or $K_{3,3}$.

[1990 : 2 Marks]

4.3 The maximum number of possible edges in an undirected graph with 'a' vertices and 'k' components is _______.

[1991 : 2 Marks]

4.4 A non-planar graph with minimum number of vertices has

(a) 9 edges, 6 vertices

(b) 6 edges, 4 vertices

(c) 10 edges, 5 vertices

(d) 9 edges, 5 vertices

[1992 : 1 Mark]

4.5 Maximum number of edges in a planar graph with n vertices is _______.

[1992 : 2 Marks]

4.6 The number of distinct simple graphs with upto three nodes is

(a) 15 (b) 10

(c) 7 (d) 9

[1994 : 1 Mark]

5. Probability

5.1 Let A and B be any two arbitrary events, then, which one of the following is true?

(a) $P(A \cap B) = P(A)\,P(B)$

(b) $P(A \cup B) = P(A) + P(B)$

(c) $P(A \mid B) = P(A \cap B)/P(B)$

(d) $P(A \cup B) < P(A) + P(B)$

[1994 : 1 Mark]

6. Linear Algebra

6.1 If a, b and c are constants, which of the following is a linear inequality?

(a) $ax + bcy = 0$

(b) $ax^2 + cy = 21$

(c) $abx + a^2y \geq 15$

(d) $xy + ax \geq 20$

[1987 : 2 Marks]

6.2 A square matrix is singular whenever:
(a) The rows are linearly independent
(b) The columns are linearly independent
(c) The rows are linearly dependent
(d) None of the above

[1987 : 2 Marks]

6.3 If A and B are real symmetric matrices of size $n \times n$. Then, which one of the following is true?
(a) $AA^T = I$
(b) $A = A^{-1}$
(c) $AB = BA$
(d) $(AB)^T = BA$

[1994 : 2 Marks]

■■■■

Answers Discrete and Engineering Mathematics

1.1	(a)	**1.2**	(a)	**1.3**	(b)	**1.4**	(b)	**1.5**	Sol.	**2.1**	Sol.	**2.2**	Sol.	**2.3**	Sol.	**2.4**	Sol.		
2.5	(c)	**2.6**	(b)	**2.7**	(c)	**3.1**	Sol.	**3.2**	Sol.	**3.3**	Sol.	**3.4**	(a)	**3.5**	(d)	**4.1**	(c)		
4.2	(d)	**4.3**	Sol.	**4.4**	(c)	**4.5**	Sol.	**4.6**	(c)	**5.1**	(c)	**6.1**	(c)	**6.2**	(c)	**6.3**	(d)		

Explanations Discrete and Engineering Mathematics

1. Mathematical Logic

1.1 (a)

Option (a) is well known valid formula (tautology) because it is a rule of inference called hypothetical syllogism.
By applying boolean algebra and simplifying, we can show that (b), (c) and (d) are invalid.
For example,
Choice (b) $\equiv (P \Rightarrow Q) \Rightarrow (\neg P \Rightarrow \neg Q)$
$\equiv (P \Rightarrow Q) \Rightarrow (P' \Rightarrow Q')$
$\equiv (P' + Q) \Rightarrow (P + Q')$
$\equiv (P' + Q)' + (P + Q')$
$\equiv PQ' + P + Q'$
$\equiv P + Q'$
$\neq 1$
So, invalid.

1.2 (a)

According to distributive properties

$$\forall x(P(x) \wedge Q(x)) \leftrightarrow \forall xP(x) \wedge \forall xQ(x)$$
$$(\forall x\, P(x) \vee \forall xQ(x)) \rightarrow \forall x(P(x) \vee Q(x))$$
$$\exists x(P(x) \vee Q(x)) \leftrightarrow \exists xP(x) \vee \exists xQ(x)$$
$$\exists x(P(x) \wedge Q(x)) \rightarrow \exists xP(x) \wedge \exists xQ(x)$$

So option (a) is valid.

1.3 (b)

(a) $(a \vee b) \rightarrow (b \wedge c)$
$\equiv (a + b)' + bc$
$\equiv a'\,b' + bc$

Therefore, $((a \vee b) \rightarrow (b \wedge c))$ is contingency and not tautology.

(b) $(a \wedge b) \rightarrow (b \vee c)$
$\equiv ab \rightarrow b + c$
$\equiv (ab)' + b + c$
$\equiv a' + b' + b + c$
$\equiv a' + 1 + c$
$\equiv 1$

So $((a \wedge b) \rightarrow (b \vee c))$ is tautology.

(c) $(a \vee b) \rightarrow (b \rightarrow c)$
$\equiv (a + b) \rightarrow (b' + c)$
$\equiv (a + b)' + b' + c$
$\equiv a'\,b' + b' + c$
$\equiv b' + c$

So $((a \vee b) \rightarrow (b \rightarrow c))$ is contingency but not tautology.

(d) $(a \rightarrow b) \rightarrow (b \rightarrow c)$
$\equiv (a' + b) \rightarrow (b' + c)$
$\equiv (a' + b)' + b' + c$
$\equiv ab' + b' + c$
$\equiv b' + c$

Therefore, $((a \rightarrow b) \rightarrow (b \rightarrow c))$ is contingency but not tautology.

1.4 (b)

The proposition $p \wedge (\sim p \vee q)$
$\equiv p(p' + q) \equiv pq$
$\equiv p \wedge q$

1.5 Sol.

TRUE

p	q	$p \leftrightarrow q$	$p \rightarrow \sim q$	$(p \leftrightarrow q) \rightarrow (p \rightarrow \sim q)$
T	T	T	F	F
T	F	F	T	T
F	T	F	T	T
F	F	T	T	T

From the truth table, $(p \leftrightarrow q) \rightarrow (p \rightarrow \sim q)$ is not tautology, hence it is true that $p \leftrightarrow q$ doesn't imply $p \rightarrow \neg q$.

2. Set Theory and Algebra

2.1 Sol.

A relation is said to be equivalence relation is
(i) Reflexive
(ii) Symmetric, and
(iii) Transitive

Union of two reflexive relations and two symmetric relations are reflexive and symmetric respectively. However, union of two transitive relations need not to be transitive. Therefore, union of two equivalence relations need not be an equivalence relation.

***Example*:**

Let R_1 and R_2 on set A = {1, 2, 3}

R_1 = {(1, 1), (2, 2,), (3, 3) (1, 2), (2, 1)} is an equivalence relation

R_2 = {(1, 1),(2, 2), (3, 3), (2, 3), (3, 2)} is an equivalence relation

$R_1 \cup R_2$ = {(1, 1), (2, 2), (3, 3,), (1, 2), (2, 1), (2, 3), (3, 2)} is not an equivalence relation,
because (1, 2) & (2, 3) needs (1, 3) element to be in transitive relation.

2.2 Sol.

(a) Set A contain n elements. Every subset of A × A is a binary relation on the set A.

∴ Number of binary relations on a set A with n elements = 2^{n^2}.

(b) Number of one-to-one functions from a set A with m-elements to a set B with n-elements is nP_m.

So the number of one-to-one functions from a set A with n-elements to itself is $nP_n = n!$.

2.3 Sol.

The complement of an element x is x′ iff LUB of x and x′ is 1 (greatest element) and GLB of x and x′ is 0 (least element).

∴ The complement of the element 'a' in the lattice = {b, c, d, e}.

2.4 Sol.

The relation have (1, 2) (2, 3) then add (1, 3) to relation. Since relation have (2, 3) (3, 4) then add (2, 4) to relation.

So the resultant relation is

= {(1, 2), (2, 3), (3, 4), (5, 4), (1, 3), (2, 4)}

Now the resultant relation have (1, 2) (2, 4) then add (1, 4) to the relation

∴ {(1,2), (2,3), (3,4), (5,4), (1,3), (2,4), (1,4)}

So the transitive closure of the relation is

= {(1,2), (2,3), (3,4), (5,4), (1,3), (2,4), (1,4)}

2.5 (c)

$S = S_1 \cup S_2 \cup S_3 \cup \dots \cup S_n$.

For S to be infinite set, atleast one of sets S_i must be infinite, since if all S_i were finite, then S will also be finite.

2.6 (b)

Number of elements in A × A = n^2

∴ Number of elements in the power set of A × A

$= 2^{n^2}$.

2.7 (c)

$(g \circ h)^2 = (g \circ h) \circ (g \circ h)$

Since group is abelian so it is commutative as well as associative.

$= g \circ (hog) h$
$= g \circ (goh) \circ h$
$= (g \circ g) \circ (hh) = g^2 \circ h^2$

So $(g \circ h)^2 = g^2 \circ h^2$ for every $g, h \in G$

3. Combinatorics

3.1 Sol.

(a) $T(n) = T(n-1) + n$

$T_n - T_{n-1} = n$

For **Homogeneous solution:**

$T_n - T_{n-1} = 0$

$$t - 1 = 0$$
$$t = 1$$

Therefore, homogenous solution is

$$T_n = C(1)^n = C$$

For **Particular solution:**

Let particular solution be $(d_0 + d_1 n)n$

$\Rightarrow$ $(d_0 + d_1 n)n - (d_0 + d_1(n-1))(n-1) = n$

$\Rightarrow$ $d_0 n + d_1 n^2 - d_0 n + d_0 - d_1(n-1)^2 = n$

$\Rightarrow$ $d_0 n + d_1 n^2 - d_0 n + d_0 - d_1(n^2 - 2n + 1) = n$

$\Rightarrow$ $d_0 + 2d_1 n - d_1 = n$

$d_0 - d_1 = 0$ and $2d_1 = 1$

$\Rightarrow$ $d_0 = d_1$ and $d_1 = \dfrac{1}{2}$

$\therefore$ $d_0 = d_1 = \dfrac{1}{2}$

So particular solution is,

$$\left(\frac{1}{2} + \frac{1}{2}n\right) \times n = \frac{n(n+1)}{2} = \frac{n^2+n}{2}$$

So complete solution is,

$$T(n) = C + \frac{1}{2}n(n+1)$$

Given, $T(1) = 1$

$$1 = C + \frac{1}{2} \times 1 \times (1+1)$$
$$1 = C + 1$$
$$C = 0$$

Therefore complete solution of the recurrence relation is $T(n) = \dfrac{n(n+1)}{2}$.

(b) The Fibonacci numbers are defined as

$$a_0 = 1, a_1 = 1$$
$$a_r = a_{r-1} + a_{r-2} \quad r \geq 2$$

$$\Rightarrow \sum_{r=2}^{\infty} a_r x^r = \sum_{r=2}^{\infty} a_{r-1} x^r + \sum_{r=2}^{\infty} a_{r-2} x^r$$

$$\Rightarrow \sum_{r=2}^{\infty} a_r x^r = x\sum_{r=2}^{\infty} a_{r-1} x^{r-1} + x^2 \sum_{r=2}^{\infty} a_{r-2} x^{r-2}$$

$\Rightarrow$ $A(x) - a_0 - a_1 x = x(A(x) - a_0) + x^2 A(x)$

Since, $a_0 = 1$ and $a_1 = 1$

$\Rightarrow$ $A(x) - 1 - x = x(A(x) - 1) + x^2 A(x)$

$$\Rightarrow \quad A(x) = \frac{1}{1 - x - x^2}$$

3.2 Sol.

$$T(n) = T\left(\frac{n}{2}\right) + 1$$

$$\Rightarrow \quad T(n) - T\left(\frac{n}{2}\right) = 1$$

Let, $n = 2^k$.

$$\Rightarrow \quad T(2^k) - T\left(\frac{2^k}{2}\right) = 1$$

$\Rightarrow$ $T(2^k) - T(2^{k-1}) = 1$

Let, $T(2^k) = x_k$

$$x_k - x_{k-1} = 1$$

For Homogeneous solution:

$$x_k - x_{k-1} = 0$$
$$t - 1 = 0$$
$$t = 1$$

So homogeneous solution is: $x_k = C(1)^k = C$

For Particular solution:

Let particular solution be $d_1 k$

$$d_1 k - d_1(k-1) = 1$$

$\Rightarrow$ $d_1 = 1$

Particular solution is k

$\therefore$ Complete solution is

$$x_k = C + k$$
$$T(2^k) = C + k$$
$$T(n) = C + \log_2 n$$

Given, $T(1) = 1$

$\Rightarrow$ $C = 1$

$\therefore$ Complete solution is: $T(n) = \log_2 n + 1$

3.3 Sol.

Let the string be of length 4 : abcd

Number of substrings of length 0 = 1 (only ε)

Number of substrings of length 1 = 4

a, b, c, d

Number of substrings of length 2 = 3

ab, bc, cd

Number of substrings of length 3 = 2

abc, bcd

Number of substrings of length 4 = 1

abcd

$\therefore$ Total number of substrings

$= 1 + (4 + 3 + 2 + 1)$

$= 1 +$ (Sum of 4 natural number)

$$= 1 + \frac{4 \times (4+1)}{2}$$

$= 11$

Therefore, total number of substrings (maximum) that can be formed from a character string of length $1+\frac{n(n+1)}{2}$.

3.4 (a)

First arranging all n zeros in a row. There is only 1 way for arranging n zeros in a row. By arranging n zeros in a row, we get (n + 1) positions to place ones.
So number of ways arranging k ones in (n + 1) positions $= {}^{n+1}C_k$.
$\therefore$ Required number of binary strings of n zeroes and k ones that no two ones are adjacent $= 1 \times {}^{n+1}C_k = {}^{n+1}C_k$.

3.5 (d)

For a string of length n:
The number of substrings of length 1 = n
The number of substrings of length 2 = n – 1
The number of substrings of length 3 = n – 2
and so on...
The number of substrings of length n is 1
So total number of substrings

$= n + (n-1) + \ldots + 1$

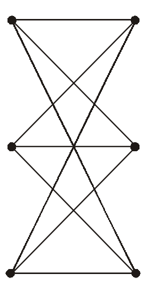
= Sum of n natural numbers

$= \frac{n(n+1)}{2}$

4. Graph Theory

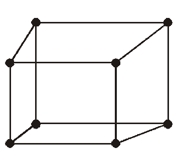
4.1 (c)

G_1:

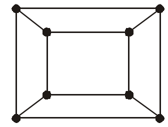


G_1 is $k_{3,3}$ which is a well-known non-planar graph.

G_2:

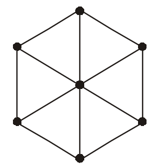


Graph G_2 is isomorphic to the following graph:



The above graph is planar. So G_2 is planar.

G_3:



G_3 is isomorphic to $K_{3,3}$ which is well known non-planar graph. Therefore, G_3 is a non-planar graph.

4.2 (d)

Kuratowski's theorem: A graph is planar if and only if, it does not contain subgraph homeomorphic to K_5 or $K_{3,3}$.

4.3 Sol.

The maximum number of possible edges in an undirected graph with "a" vertices and "k" components is $\frac{(a-k+1)(a-k)}{2}$.

4.4 (c)

K_5 is smallest non-planar graph in terms of number of vertices.
The number of vertices in K_5 is 5 and number of edges in K_5 is $\frac{5\times 4}{2} = 10$.

4.5 Sol.

Maximum number of edges in a connected, planar, simple graph with n vertices is 3n – 6.

4.6 (c)

Assume that the vertices are unlabelled. Number of distinct simple graphs with 1 node = 1

(i) 

Number of distinct simple graphs with 2 nodes = 2

(i) (No edges)

(ii) (1 edge)

Number of distinct simple graphs with 3 nodes = 4

(i) (No edges)

(ii) (1 edge)

(iii) (2 edges)

(iv)  (3 edges)

Therefore, total number of distinct simple graphs upto three nodes = 1 + 2 + 4 = 7.

5. Probability

5.1 (c)

(a) $P(A \cap B) = P(A)\,P(B)$ is false since this is true if and only if A and B are independent events.

(b) $P(A \cup B) = P(A) + P(B)$ is false since $P(A \cap B)$ is zero if and only if A and B are mutually exclusive.

(c) $P(A \mid B) = P(A \cap B)/P(B)$ is true.

(d) $P(A \cup B) < P(A) + P(B)$ is false.

Since $P(A \cup B) \le P(A) + P(B)$

6. Linear Algebra

6.1 (c)

(a) Equation $ax + bcy = 0$ is linear equality not linear inequality.

(b) $ax^2 + cy = 21$ is not linear inequality because here degree of x is 2 and also it is equality not inequality.

(c) $abx + a^2y \ge 15$ is linear inequality because both x and y is in its first degree.

(d) $xy + ax \ge 20$ is not linear inequality because xy term is of degree 2 not of degree 1.

6.2 (c)

If the rows (or columns) of a square matrix are linearly dependent, then the determinant of matrix becomes zero.

Therefore, whenever the rows are linearly dependent, the matrix is singular.

6.3 (d)

The matrix M is said to be symmetric iff $M^T = M$

$$(AB)^T = B^TA^T = BA$$

Since A and B are symmetric, $A^T = A$ and $B^T = B$.

Theory of Computation

UNIT

II

CONTENTS

Theory of Computation

1. Finite Automata : Regular Languages

1.1 How many substrings (of all lengths inclusive) can be formed from a character string of length n? Assume all characters to be distinct. Prove your answer.

[1989 : 2 Marks]

1.2 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

[1990 : 2 Marks]

1.3 Let $r = 1\,(1 + 0)^*$, $s = 11^*0$ and $t = 1^*\,0$ be three regular expressions. Which one of the following is true?
(a) $L(s) \subseteq L(r)$ and $L(s) \subseteq L(t)$
(b) $L(r) \subseteq L(s)$ and $L(s) \subseteq L(t)$
(c) $L(s) \subseteq L(t)$ and $L(s) \subseteq L(r)$
(d) $L(t) \subseteq L(s)$ and $L(s) \subseteq L(r)$

[1991 : 2 Marks]

1.4 Which of the following regular expression identifies are true?
(a) $r\,(^*) = r^*$
(b) $(r^*\,s^*)^* = (r + s)^*$
(c) $(r + s)^* = r^* + s^*$
(d) $r^*\,s^* = r^* + s^*$

[1992 : 2 Marks]

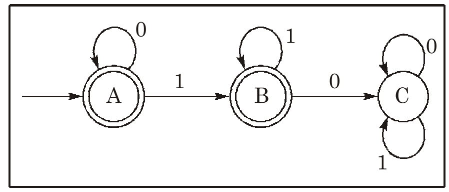
1.5 State True or False with one line explanation.
A FSM (Finite State Machine) can be designed to add two integers of any arbitrary length (arbitrary number of digits).

[1994 : 1 Mark]

1.6 The number of substrings (of all lengths inclusive) that can be formed from a character string of length n is
(a) n (b) n^2
(c) $\dfrac{n(n-1)}{2}$ (d) $\dfrac{n(n+1)}{2}+1$

[1994 : 2 Marks]

1.7 The regular expression for the language recognized by the finite state automation



[1994 : 2 Marks]

2. Push Down Automata: CFL & DCFL

2.1 A context-free grammar is ambiguous if
(a) the grammar contains useless non-terminals
(b) it produces more than one parse tree for some sentence
(c) some production has two non-terminals side by side on the right-hand side
(d) None of the above

[1987 : 2 Marks]

2.2 Fortran is
(a) regular language
(b) context free language
(c) context sensitive language
(d) None-of the above

[1987 : 2 Marks]

2.3 Context free languages and regular languages are both closed under the operation(s) of
(a) union
(b) intersection
(c) concatenation
(d) complementation

[1989 : 2 Marks]

2.4 Context-free languages are
(a) closed under union
(b) closed under complementation
(c) closed under intersection
(d) closed under Kleene closure

[1992 : 2 Marks]

2.5 If G is a context-free grammar and w is a string of length n in L(G), how long is a derivation of w in G, if G is Chomsky normal form?
(a) 2n (b) 2n + 1
(c) 2n – 1 (d) n
[1992 : 2 Marks]

2.6 Which of the following features cannot be captured by context-free grammar?
(a) Syntax of if-then-else statements
(b) Syntax of recursive procedures
(c) Whether a variable has been declared before its use
(d) Variable names of arbitrary length.
[1994 : 2 Marks]

3. Turing Machine: RE, REC and Undecidability

3.1 Which of the following problems are undecidable?
(a) Membership problem in context-free languages
(b) Whether a given context-free language is regular
(c) Whether a finite state automation halts on all inputs
(d) Membership problem for type 0 languages
[1989 : 2 Marks]

3.2 It is undecidable whether
(a) an arbitrary Turing machine halts within 10 steps
(b) a Turing machine prints a specific letter
(c) a Turing machine computes the product of two numbers
(d) None of the above
[1990 : 2 Marks]

3.3 Recursive languages are:
(a) a proper superset of context free languages
(b) always recognizable by pushdown automata
(c) also called type-0 languages
(d) recognizable by turing machines
[1990 : 2 Marks]

3.4 Which one of the following is the strongest correct statement about a finite language over some finite alphabet Σ?
(a) It could be undecidable
(b) It is turing-machine recognizable
(c) It is a regular language
(d) It is a context-sensitive language
[1991 : 2 Marks]

3.5 In which of the cases stated below is the following statement true?
"For every non-deterministic machine M_1 there exists an equivalent deterministic machine M_2 recognizing the same language".
(a) M_1 is non-deterministic finite automaton
(b) M_1 is a non-deterministic PDA
(c) M_1 is a non-deterministic Turing machine
(d) For no machine M_1 use the above statement true
[1992 : 2 Marks]

3.6 Which of the following conversions is not possible (algorithmically)?
(a) Regular grammar to context free grammar
(b) Non-deterministic FSA to deterministic FSA
(c) Non-deterministic PDA to deterministic PDA
(d) Non-deterministic Turing machine to deterministic Turing machine
[1994 : 2 Marks]

■■■■

Answers Theory of Computation

1.1	Sol.	**1.2**	(c)	**1.3**	(a, c)	**1.4**	(b)	**1.5**	Sol.	**1.6**	(d)	**1.7**	Sol.
2.1	Sol.	**2.2**	(c)	**2.3**	(a, c)	**2.4**	(a, d)	**2.5**	(c)	**2.6**	(c)	**3.1**	(b, d)
3.2	(b, c)	**3.3**	(a, d)	**3.4**	(c)	**3.5**	(a, c)	**3.6**	(c)				

Explanations Theory of Computation

1. Finite Automata : Regular Languages

1.1 Sol.

Number of substrings (of all lengths inclusive) that can be formed from a character strings of length n is $\frac{n(n+1)}{2}+1$.

Example: Let the length of string be 4, |w| = 4 and w = ABCD

No. of substrings of length 0 is 1(ε)
No. of substrings of length 1 is 4(A, B, C, D)
No. of substrings of length 2 is 3(AB, BC, CD)
No. of substrings of length 3 is 2(ABC, BCD)
No. of substrings of length 4 is 1 (ABCD)
So, no.of substrings of all length by a string of length 4 is

$$1 + (4 + 3 + 2 + 1) = 1+\frac{4\times(4+1)}{2}=11$$

1.2 (c)

Since regular languages are closed under union, intersection, complementation, concatenation as well as Kleene closure.
So (c) is correct and (a), (b) and (d) are false.

1.3 (a, c)

$$\mathbf{r} = 1(1+0)^*$$

So the language corresponds to **r** is all strings starting with 1.

$$\mathbf{s} = 1\,1^*\,0$$

The language corresponds to s is all strings starting with 1 followed by any number of 1 and end with 0. So **s** has more restrictions as compared to r.

So, $L(\mathbf{s}) \subseteq L(\mathbf{r})$

$$\mathbf{t} = 1^*\,0$$

The language corresponds to t is all strings ending with 0. Since **s** has to start with atleast one 1 but t can start with zero 1s. So s has more restriction compared to **t**.

So, $L(\mathbf{s}) \subseteq L(\mathbf{t})$

Note: Choices (a) and (c) are actually same.

1.4 (b)

According to identities of regular expression

$$(r^*\, s^*)^* = (r + s)^*$$

1.5 Sol.

FALSE, A FSM (Finite State Machine) can't be designed to add two integers of any arbitrary length because FSM have finite memory and can't store integers of any arbitrary length.

1.6 (d)

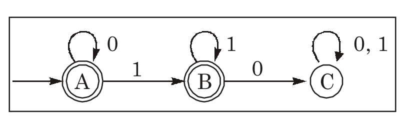
Number of substrings (of all lengths inclusive) that can be formed from a character strings of length n is $\frac{n(n+1)}{2}+1$.

Example: Let the length of string be 4, | w | = 4 and w = ABCD

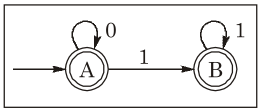
No. of substrings of length 0 is 1(ε)
No. of substrings of length 1 is 4(A, B, C, D)
No. of substrings of length 2 is 3(AB, BC, CD)
No. of substrings of length 3 is 2(ABC, BCD)
No. of substrings of length 4 is 1 (ABCD)
So, no.of substrings of all length by a string of length 4 is

$$1 + (4 + 3 + 2 + 1) = 1+\frac{4\times(4+1)}{2}=11$$

1.7 Sol.



Since C is dead state so just remove this



Regular expression = RE of A + RE of B

= $0^* + 0^*\ 11^*$

= $0^*\ (\varepsilon + 11^*) = 0^*\ 1^*$

2. Push Down Automata: CFL & DCFL

2.1 Sol.

A context-free grammar is ambiguous if

1. It produces more than one parse tree for some sentence.

or

2. It produces more than one leftmost derivation for some sentence.

or

3. It produces more than one rightmost derivation for some sentence.

2.2 (c)

The languages like C, Fortran and Pascal are context sensitive languages. Due to presence of the following two features which cannot be handled by a PDA:

1. Variable declared before use
2. Matching formal and actual parameters of functions.

2.3 (a, c)

Regular language is closed under the all the operations of union, inter-section, concatenation, complementation and Kleene closure. However context free languages are closed under the operations of union, concatenation and Kleene closure but not closed under inter-section and complementation.

2.4 (a, d)

Context free languages are closed under union, concatenation, and Kleene closure. However, context free languages are not closed under complementation and intersection.

2.5 (c)

A context-free grammar G is in Chomsky normal form if all productions are in one of two simple form, either:

1. $A \to BC$ where A, B and C are variables, or
2. $A \to a$ where A is a variable and a is a terminal

So for any string of length n first production of type $A \to BC$ is used $n - 1$ times to produce sentential form of length n containing only variables and then each variable is replaced by a terminal using productions of type $A \to a$, n times. So the length of derivation of string w of length n in CNF is

$$(n - 1) + n = 2n - 1$$

Example:

$S \to XY$

$Y \to XZ$

$X \to a$

$Z \to b$

To derive the string "aab" by using above CNF

$S \to XY$

$\Rightarrow aY$

$\Rightarrow aXZ$

$\Rightarrow aaZ$

$\Rightarrow aab$

It requires 5 steps

$= 2 \times 3 - 1 = 2 * |aab| - 1$

$= 2n - 1$

2.6 (c)

(a) The language corresponding to syntax of if-then-else statements is context-free language. So syntax of if-then-else statements can be captured by context-free grammar.

(b) The language corresponding to syntax of recursive procedures is context-free language. So syntax of recursive procedures can be captured by context-free-grammar.

(c) The language corresponding to variable has been declared before its use is

$L = \{ww \mid w \text{ is variable}\}$

L is context sensitive language. So the feature that a variable has been declared before its use can't be captured by context free grammar.

(d) The feature that variable names of arbitrary length can be captured by context free grammar.

3. Turing Machine: RE, REC and Undecidability

3.1 (b, d)

(a) **Decidable**: Membership means "given any context free grammar (CFG) G and a word $w \in \Sigma^*$, does $w \in L(G)$ i.e., can G generate given string w?" Membership problem in CFL's is decidable.

(b) **Undecidable**: Regularity of context free language is undecidable because there exist no algorithm to check whether a CFL is regular.

(c) **Decidable:** Finite state automaton is a decider. Decider is a machine that always halts. So finite automaton halts on all inputs is a trivially decidable problem, since the answer is always yes.

(d) **Undecidable:** Type-0 language is recursive enumerable language (RE) and membership problem for RE language is undecidable.
So membership problem for type0 languages is undecidable problem.

3.2 (b, c)

(a) It is decidable whether an arbitrary turing machine halts within 10 steps.
This can be decided by running the TM program on UTM for finite number (10) of steps.

(b) It is undecidable whether a turing machine prints a specific letter because for printing specific letter turing machine has to enter a specific state on an input. State entry problem can be reduced to this problem. State entry problem is undecidable so this problem also become undecidable.

(c) It is undecidable whether a given turing machine computes the product of two numbers. To decide this we have to first of all decide whether the given TM program halts. But the halting problem is undecidable.
So, this problem is also undecidable.

3.3 (a, d)

(a) **True.** Every CFL is recursive language whereas there exist some recursive languages which are not CFL's. So recursive languages are a proper superset of context free languages.
CFL $\subset$ REC

(b) **False.** Since there exist some recursive languages which are not context free. So recursive languages are not always recognizable by pushdown automata.

(c) **False.** Recursive enumerable languages are called as type 0 languages.

(d) **True.** Recursive languages are recognizable as well as decidable by turing machine.

3.4 (c)

Every finite language over some finite alphabet Σ is regular because for every finite language DFA or NFA can be created by brute force.

3.5 (a, c)

(a) There exist an algorithm to convert every NFA to DFA. So recognition power of NFA and DFA, is same.

(b) Since there exist no algorithm to convert NPDA to DPDA and recognition power of both are not same.

(c) The recognition power of NTM (non-deterministic turing machine) and DTM (deterministic turing machine) is same.

(d) Since for turing machines and finite automation the above statement is true so (d) is not true.

3.6 (c)

(a) Every regular grammar is also context free grammar. So this is trivially decidable.

(b) There exist algorithm to convert non-deterministic FSA to deterministic FSA since their recognition power is same.

(c) Since the recognition power of non-deterministic PDA and deterministic PDA is not same i.e., there exists some languages like $L = \{ww^R \mid w \in (a, b)^*\}$ which are accepted by non-deterministic PDA but not by deterministic PDA. So, there exist no algorithm for converting non-deterministic PDA to deterministic PDA.

(d) Since the recognition power of non-deterministic turing machine and deterministic turing machine are same. So there exist an algorithm to convert every non-deterministic turing machine to deterministic turing machine.

■■■■

Digital Logic

UNIT

III

CONTENTS

Digital Logic

1. Logic Functions and Minimization

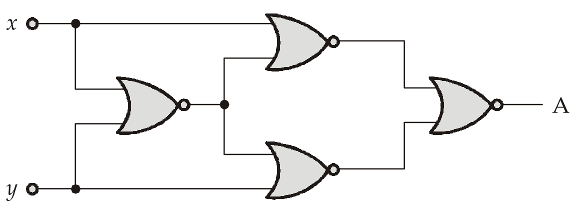
1.1 Find the minimum product of sum of the following expression $f = ABC + \overline{A}\overline{B}\overline{C}$

[1990 : 5 Marks]

1.2 Find the minimum sum of products form of the logic function
$f(A, B, C, D) = \Sigma d(3, 11, 12, 14) + \Sigma m(0, 2, 8, 10, 15)$
Where m and d denote the minterm and don't care respectively

[1991 : 5 Marks]

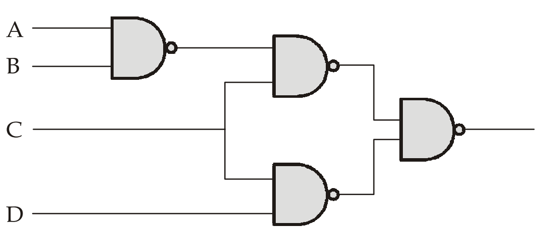
1.3 Identify the logic function performed by the circuit shown in figure



(a) Exclusive OR (b) Exclusive NOR
(c) NAND (d) NOR

[1993 : 1 Marks]

1.4 The logic expression for the output of the circuit shown in below figure is



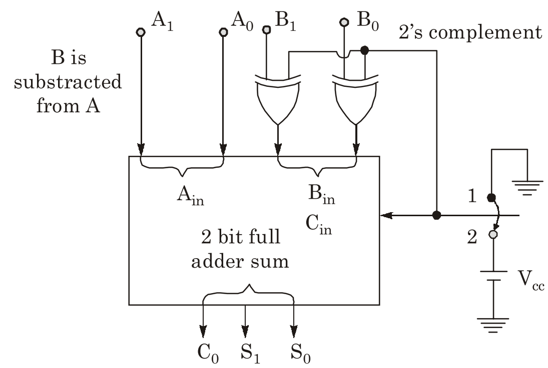
(a) $\overline{A}C + \overline{B}C + CD$
(b) $A\overline{C} + B\overline{C} + \overline{C}D$
(c) $ABC + \overline{C}\overline{D}$
(d) $\overline{A}\overline{B} + \overline{B}\overline{C} + \overline{C}\overline{D}$

[1994 : 2 Marks]

2. Combinational Circuits

2.1 Fill in the blanks:

In the two bit full-adder/subtractor unit shown in figure, when the switch is in position 2. ...... using .... arithmetic



[1990 : 2 Marks]

2.2 Show with the help of a block diagram how the Boolean function: $f = AB + BC + CA$ can be realized using only a 4 : 1 multiplexer.

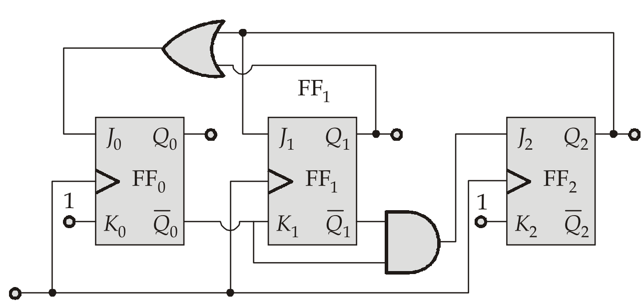
[1990 : 5 Marks]

3. Sequential Circuits

3.1 RAM is a combinational Circuit and PLA is a sequential circuit? (T/F)

[1990 : 2 Marks]

3.2 For the synchronous counter shown in figure write the truth table of Q_0, Q_1 and Q_2 after each pulse starting from $Q_0 = Q_1 = Q_2 = 0$ and determine the counting sequence also the modulus of the counter

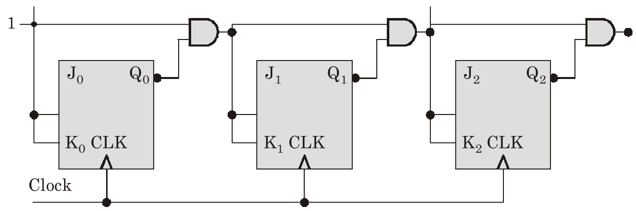


What is the modulus of the counter with initial states $Q_2 Q_1 Q_0 = 000$.

(a) 3 (b) 4
(c) 5 (d) 6

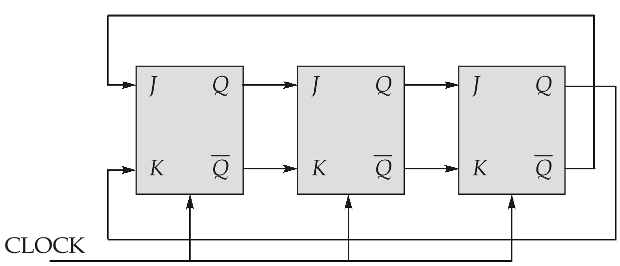
[1990 : 5 Marks]

3.3 Find the maximum clock frequency at which the counter in figure, can be operated. Assume that the propagation delay through each flip-flop and AND gate is 10 ns. Also assume that the setup time for the JK inputs of the flip-flops is negligible.



[1991 : 2 Marks]

3.4 For the initial state of 000, the function performed by the arrangement of the J-K flip-flops in figure is:



(a) Shift register
(b) Mod-3 counter
(c) Mod-6 counter
(d) Mod-2 counter

[1993 : 1 Marks]

3.5 The number of flip-flops required to construct a binary modulo N counter is ________.

[1994 : 1 Marks]

4. Number Systems

4.1 Consider the number given by the decimal expression.

$$16^3 \times 9 + 16^2 \times 7 + 16 \times 5 + 3$$

The number of 1's in the unsigned binary representation of the number is ______

[1991 : 2 Mark]

4.2 Using the 8087 arithmetic coprocessor with the 80186 CPU requires that the 8087 CPU is operated for _______

[1991 : 2 Mark]

4.3 When two 4-bit binary number $A = a_3\, a_2\, a_1\, a_0$, $B = b_3\, b_2\, b_1\, b_0$ are multiplied, the digit c_1 of the product C is given by

[1991 : 2 Mark]

4.4 A 32-bit floating-point number is represented by a 7-bit signed exponent, and a 24-bit fractional mantissa. The base of the scale factor is 16,

(a) the range of the exponent is _____
(b) the range of the exponent is _____ if the scale factor is represented in excess-64 format.

[1990 : 2 Marks]

4.5 Consider n-bit (including sign bit) 2's complement representation of integer numbers. The range of integer values N, that can be represented is ________ $\leq$ N $\leq$ ________.

[1994 : 1 Marks]

■■■■

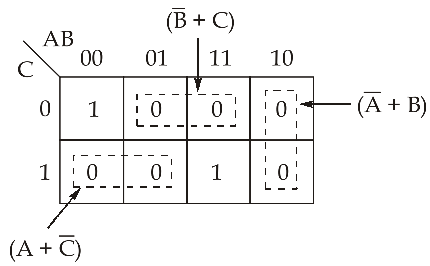
Answers Digital Logic

1.1	Sol.	**1.2**	Sol.	**1.3**	(b)	**1.4**	(a)	**2.1**	Sol.	**2.2**	Sol.	**3.1**	Sol.	**3.2**	(c)	**3.3**	Sol.
3.4	(c)	**3.5**	Sol.	**4.1**	Sol.	**4.2**	Sol.	**4.3**	Sol.	**4.4**	Sol.	**4.5**	Sol.				

Explanations Digital Logic

1. Logic Functions and Minimization

1.1 Sol.

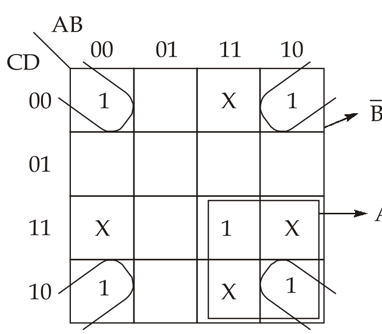


POS from is

$$f = (\overline{B}+C)(\overline{A}+B)(A+\overline{C})$$

1.2 Sol.

$$f = (A, B, C, D)$$
$$= \Sigma m(0, 2, 8, 10, 15) + \Sigma d(3, 11, 12, 14)$$



$f(A, B, C, D) = \overline{B}\overline{D} + AC$

1.3 (b)

$$\overline{\left[x+\overline{(x+y)}\right]+\left[y+\overline{(x+y)}\right]}$$

$$\overline{\left[x+\overline{(x+y)}\right]}\cdot\overline{\left[y+\overline{(x+y)}\right]}$$

$$= (x+\overline{x}.\overline{y})(y+\overline{y}.\overline{x})$$

$$= (x+\overline{y})(y+\overline{x}) = xy+\overline{x}\,\overline{y} = x \odot y$$

1.4 (a)

$$\overline{\overline{\overline{AB}.C.\overline{CD}}} = \overline{AB}.C+CD$$
$$= (\overline{A}+\overline{B})C + CD$$
$$= \overline{A}C+\overline{B}C+CD$$

2. Combinational Circuits

2.1 Sol.

When the switch is in position 2, 2's complement of B is added to A.
This means B in subtracted from A.
Therefore, when the switch is in position 2

B is subtracted from A using

2's complement arithmetic.

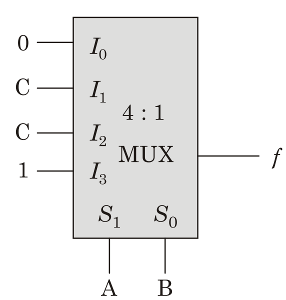
2.2 Sol.

$$f = \underset{(6,7)}{\underset{\downarrow}{AB}} + \underset{(3,7)}{\underset{\downarrow}{BC}} + \underset{(5,7)}{\underset{\downarrow}{CA}}$$

$$= \Sigma m(3,\ 5,\ 6,\ 7)$$

$S_1 \leftarrow A$, $S_0 \leftarrow B$

		I_0	I_1	I_2	I_3
		00	01	10	11
$\overline{C}$	0	0	2	4	6
C	1	1	3	5	7
		0	C	C	1



3. Sequential Circuits

3.1 Sol.

False: A RAM is sequential circuit and PLA is combinational circuit.

3.2 (c)

		FF2		FF1		FF0	
CLK	$Q_2\, Q_1\, Q_0$	$J_2 = \bar{Q}_1\bar{Q}_0$	$K_2 = 1$	$J_1 = Q_2$	$K_1 = \bar{Q}_0$	$J_0 = Q_2 + Q_1$	$K_0 = 1$
0	0 0 0	1	1	0	1	0	1
1	1 0 0	1	1	1	1	1	1
2	0 1 1	0	1	0	0	1	1
3	0 1 0	0	1	0	1	1	1
4	0 0 1	0	1	0	0	0	1
5	0 0 0						

Counter is back to initial state after 5 clock pulse, so the mod value is 5.

3.3 Sol.

All the flip flops are operated by same clock, together all takes one propagation delay.

All the AND gates consumes one propagation delay individually.

Total propagation delay

$$= T_{CLK} \geq T_{\text{flip-flops}} + T_{\text{AND gates}}$$

$$= 10 \text{ ns} + (10 + 10 + 10) \text{ ns} = 40 \text{ ns}$$

Maximum clock frequency

$$= \frac{1}{T_{CLK}} = \frac{1}{40\,\text{ns}} = \frac{10^9}{40}$$

$$= 25 \text{ MHz}$$

Maximum clock frequency at which the counter can operate is 25 MHz.

3.4 (c)

Given circuit is a 3 bit Johnson counter. Hence mod value is 6.

3.5 Sol.

$\log_2^N$

4. Number Systems

4.1 Sol.

The number in decimal expression is given as

$= 16^3 \times 9 + 16^2 \times 7 + 16 \times 5 + 3$

$= 16^3 \times 9 + 16^2 \times 7 + 16^1 \times 5 + 16^0 \times 3$

The above number can be represented in hexadecimal as $= (9753)_H$

The binary equivalent of above hexadecimal number is

$$(9753)_H = (1001\ 0111\ 0101\ 0011)_B$$

The number of 1's in the unsigned binary representation of the number is 9.

4.2 Sol.

Floating point arithmetic.

4.3 Sol.

$$C = A \times B$$

$$C_7C_6C_5C_4C_3C_2C_1C_0 = (a_3a_2a_1a_0) \times (b_3b_2b_1b_0)$$

$$\begin{array}{r} a_3\,a_2\,a_1\,a_0 \\ \times\ b_3\,b_2\,b_1\,b_0 \end{array}$$

			a_3b_0	a_2b_0	a_1b_0	a_0b_0
		a_3b_1	a_2b_1	a_1b_1	a_0b_1	
	a_3b_2	a_2b_2	a_1b_2	a_0b_2		
a_3b_3	a_2b_3	a_1b_3	a_0b_3			
a_3b_3	$(a_3b_2 + a_2b_3)$	$(a_3b_1 + a_2b_2 + a_1b_3)$	$(a_3b_0 + a_2b_1 + a_1b_2 + a_0b_3)$	$(a_2b_0 + a_1b_1 + a_0b_2)$	$(a_1b_0 + a_0b_1)$	(a_0b_0)
C_6	C_5	C_4	C_3	C_2	C_1	C_0

The digit C_1 of the product C is given by

$$C_1 = a_1b_0 + a_0b_1$$

4.4 Sol.

The given floating point number format is

1	7	24
S	EXP	Fraction (Mantissa)
31	30 24	23 0

$$\text{Exponent} = \text{bias} + 64$$

Above floating point architecture is IBM floating point architecture.

The number is represented as the following formula:

$$(-1)^S \times 0\,.\,\text{Mantissa} \times 10^{\,EXP-64}$$

(a) The range of the exponent is the range of the 7-bit signed numbers, which is –64 to +63.

(b) The range of the exponent if the scale factor is represented in excess-64 format is 0 to +127.

4.5 Sol.

$(-2^{n-1}, +2^{n-1} - 1)$

■■■■

Computer Organization and Architecture

UNIT

IV

CONTENTS

Computer Organization & Architecture

4. Memory Organization

4.1 A block-set associative cache memory consists of 128 blocks divided into four block sets. The main memory consists of 16,384 blocks and each block contains 256 eight bit words.

(i) How many bits are required for addressing the main memory?

(ii) How many bits are needed to represent the TAG, SET and WORD fields?

[1990 : 2 Marks]

4.2 In a two-level virtual memory, the memory access time for main memory, $t_{A_1} = 10^{-8}$ sec, and the memory access time for the secondary memory, tag = 10^{-3} sec. What must be the hit ratio, H such that the access efficiency is within 80 percent of its maximum value.

[1990 : 2 Marks]

4.3 State whether the following statements are TRUE or FALSE with reason. Transferring data in blocks from the main memory to the cache memory enables an interleaved main memory unit to operate unit at its maximum speed.

[1990 : 2 Marks]

4.4 The total size of address space in a virtual memory system is limited by

(a) the length of MAR

(b) the available secondary storage

(c) the available main memory

(d) All of the above

[1991 : 2 Marks]

5. IO Organization

5.1 On receiving an interrupt from an I/O device the CPU:

(a) halts for predetermined time

(b) hands over control of address bus and data bus to the interrupting device.

(c) branches off to the interrupt service routine immediately.

(d) branches off to the interrupt service routine after completion of the current instruction.

[1987 : 2 Marks]

5.2 The data transfer between memory and I/O devices using programmed I/O is faster than interrupt driven I/O.

[1990 : 2 Marks]

5.3 The flags are affected when conditional CALL or JUMP instructions are executed.

[1990 : 2 Marks]

5.4 Transferring data in blocks from the main memory to the cache memory enables an interleaved main memory unit to operate unit at its maximum speed.

[1990 : 2 Marks]

5.5 A certain moving arm disk storage, with one head, has the following specifications.

Number of tracks/recording surface = 200

Disk rotation speed = 2400 rpm

Track storage capacity = 62,500 bits

The average latency of this device is P msec and the data transfer rate is Q bits/sec.

Write the value of P and Q.

[1991 : 2 Marks]

■■■■

Answers Computer Organization & Architecture

4.1 Sol. **4.2** Sol. **4.3** Sol. **4.4** (b) **5.1** (d) **5.2** Sol. **5.3** Sol. **5.4** Sol. **5.5** Sol.

Explanations Computer Organization & Architecture

4. Memory Organization

4.1 Sol.

(i) Main memory size

= 16384 blocks

$= 16384 \times 256$ words

$= 2^{14} \times 2^{8}$ words $= 2^{22}$ words

Number of bits required for addressing the main memory is 22 bits

22 bits

Physical Address

(ii) Set associative cache

Block size = 256 word = 2^8 words

$\therefore$ No. of bits in WORD OFFSET is 8 bits

No. of blocks in set-associative cache = 128

Number of blocks in one set = 4

$$\therefore \text{ No. of sets in cache} = \frac{128}{4} = 32 = 2^5$$

Number of bits in SET OFFSET is 5

22 bits

TAG	SET OFFSET	WORD OFFSET
9 bits	5 bits	8 bits

Number of bits in TAG = 22 – (8 + 5) = 9 bits

4.2 Sol.

Main memory access time,

$$t_{a_1} = 10^{-8} \text{ sec}$$

Secondary memory access time,

$$t_{a_2} = 10^{-3} \text{ sec}$$

Access efficiency,

$$\eta = 80\% = 0.8$$

Average access time,

$$t_{avg} = \eta \times t_{a_2} = 0.8 \times 10^{-3} \text{ sec}$$

Hit ratio, H

$$t_{avg} = H \times t_{a_1} + (1 - H) t_{a_2}$$

$$0.8 \times 10^{-3} = H \times 10^{-8} + (1 - H) \times 10^{-3}$$

$$0.8 = H \times 10^{-5} + (1 - H)$$

$$H(1 - 10^{-5}) = 0.2$$

$$H = \frac{0.2}{(1 - 10^{-5})} = 0.200002 \approx 20\%$$

The hit ratio H must be 20%.

4.3 Sol.

Statement is true, main memory transfers the data in the form of block to cache and cache transfers the data in the form of words, so it enables the interleaved main memory unit to operate unit at maximum speed.

4.4 (b)

Since virtual memory is implemented on secondary storage.

5. IO Organization

5.1 (d)

After getting the interrupt CPU first executes current instruction then services the interrupt.

5.2 Sol.

False: The data transfer between memory and I/O devices using interrupt driven I/O is faster than programmed I/O because programmed I/O technique require constant monitoring on peripheral by CPU, once data transfer is initiated, CPU have to wait for next transfer whereas in interrupt driven I/O once data transfer initiated, CPU executes next program without wasting time and the interface keep monitoring the device.

5.3 Sol.

False: Conditional CALL or JUMP instructions only test flags, they do not affect any of the flag values.

5.4 Sol.

True: If the main memory of a computer is structured as a collection of physically separate modules, each with its own address buffer register (ABR) and data buffer register (DBR), memory access operation may proceed in more than one module at the same time. Thus, the aggregate rate of transmission of words to and from the main memory system can be increased.

5.5 Sol.

Average latency,

$$P = \frac{1}{2} \times \text{Rotation time}$$

$$= \frac{1}{2} \times \frac{60}{2400} \text{sec} = 12.5 \text{ msec}$$

In $\frac{60}{2400}$ sec $\rightarrow 200 \times 62500$ bits

1 sec $\rightarrow \dfrac{200 \times 62500 \times 2400}{60}$ bits/sec

$\rightarrow$ 59.60 Mbps

Data transfer rate, Q = 59.60 Mbps

■■■■

Programming and Data Structures

UNIT

V

CONTENTS

Programming & Data Structures

1. Programming

1.1 Study the following program written in a block-structured language:

```
var x, y : integer;
procedure P(n:integer);
begin
    x : = (n + 2) / (n – 3);
end;
procedure Q
var x, y : integer;
begin
    x : = 3;
    y : = 4;
    P(y);
    Write (x)                    ...(1)
end;
begin
    x : = 7;
    y : = 8;
    Q;
    Write (x)                    ...(2)
end
```

What will be printed by write statements marked (1) and (2) in the program if variables are statically scoped?

(a) 3, 6
(b) 6, 7
(c) 3, 7
(d) None of these

[1987 : 2 Marks]

1.2 Match the pairs in the following

List-I

A. Pointer data type
B. Activation record
C. Repeat-until
D. Coercion

List-II

P. Type conversion
Q. Dynamic data structure
R. Recursion
S. Nondeterministic loop

[1990 : 2 Marks]

1.3 An unrestricted use of the "goto" statement is harmful because

(a) it makes it more difficult to verify programs
(b) it increases the running time of the programs
(c) it increases the memory required for the programs
(d) it results in the complier generating longer machine code

[1994 : 1 Mark]

2. Arrays

2.1 In a compact single dimensional array representation for lower triangular matrices (i.e. all the elements above the diagonal are zero) of size $n \times n$, non-zero elements (i.e. elements of the lower triangle) of each row are stored one after another, starting from the first row, the index of the $(i, j)^{th}$ element of the lower triangular matrix in this new representation is

(a) $i + j$
(b) $i + j - 1$
(c) $(j-1) + \frac{i(i-1)}{2}$
(d) $i + \frac{j(j-1)}{2}$

[1994 : 2 Marks]

3. Stacks and Queues

3.1 The following sequence of operations is performed on a stack:

PUSH (10), PUSH (20), POP, PUSH (10), PUSH (20), POP, POP, POP, PUSH (20), POP

The sequence of values popped out is:

(a) 20, 10, 20, 10, 20
(b) 20, 20, 10, 10, 20
(c) 10, 20, 20, 10, 20
(d) 20, 20, 10, 20, 10

[1991 : 2 Marks]

3.2 Which of the following permutations can be obtained in the output (in the same order) using a stack assuming that the input is the sequence 1, 2, 3, 4, 5 in that order?

(a) 3, 4, 5, 1, 2
(b) 3, 4, 5, 2, 1
(c) 1, 5, 2, 3, 4
(d) 5, 4, 3, 1, 2

[1994 : 2 Marks]

4. Linked List

4.1 In a circular linked list organization, insertion of a record involves modification of
(a) One pointer
(b) Two pointers
(c) Multiple pointers
(d) No pointer

[1987 : 2 Marks]

4.2 Linked lists are not suitable data structures of which one of the following problems?
(a) Insertion sort
(b) Binary search
(c) Radix sort
(d) Polynomial manipulation

[1994 : 2 Marks]

5. Trees

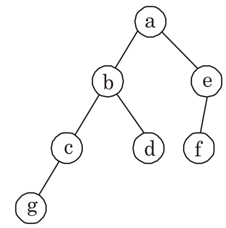
5.1 It is possible to construct a binary tree uniquely whose pre-order and post-order traversals are given. (True/False)

[1987 : 1 Mark]

5.2 If the number of leaves in a tree is not a power of 2, then the tree is not a binary tree.(True/False)

[1987 : 1 Mark]

5.3 If the binary tree in figure is traversed in inorder, then the order in which the nodes will be visited is ________.



[1991 : 2 Marks]

7. Hashing

7.1 A Hash table with ten buckets with one slot per bucket is shown in the following figure with the symbols S1 to S7 entered into it using some hashing function with linear probing. The worst case number of comparisons required when the symbol being searched is not in the table is

0	S_7
1	S_1
2	
3	S_4
4	S_2
5	
6	S_5
7	
8	S_6
9	S_3

[1989 : 2 Marks]

■■■■

Answers Programming & Data Structures

1.1	(a)	**1.2**	Sol.	**1.3**	(a)	**2.1**	(c)	**3.1**	(b)	**3.2**	(b)	**4.1**	(b)	**4.2**	(b)	**5.1**	Sol.
5.2	Sol.	**5.3**	Sol.	**7.1**	Sol.												

Explanations Programming & Data Structures

1. Programming

1.1 (a)

Following is the execution sequence of program.

main program

int int

x y

6 $\boxed{7}$ $\boxed{8}$

Procedure Q

x y

$\boxed{3}$ $\boxed{4}$

Procedure P

n

$\boxed{4}$

$x = \frac{4+2}{4-3} = \frac{6}{1} = 6$ (update the global x)

1 Write (x) – outputs 3.

2 write (x) – outputs 6

So the correct answer is option (a)

1.2 Sol.

- In programming language, coercion is a way of implicitly or explicity, changing an entity of one datatype into another.
- Repeat-until is a nondeterministic loop.
- The portion of the stack used for an invocation of a function is called the function's stack frame or activation record. So activation record is used for recursion.
- Dynamic data structure is used for pointer data type.
  a-q, b-r, c-s, d-p

1.3 (a)

A goto statement causes a program to unconditionally transfer control to the statement associated with the label specified on the goto statement. A goto statement is always undesirable because it makes program more difficult to verify.

2. Arrays

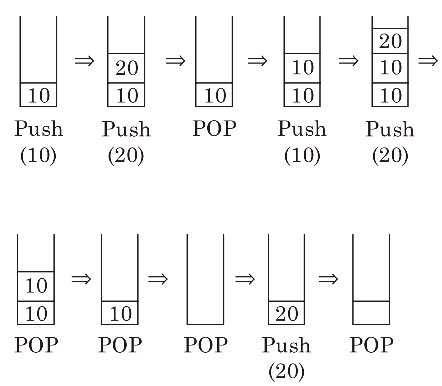
2.1 (c)

LOC (i, j) formula of lower triangular matrix used

$= (j-1) + \frac{i(i-1)}{2}$.

3. Stacks and Queues

3.1 (b)



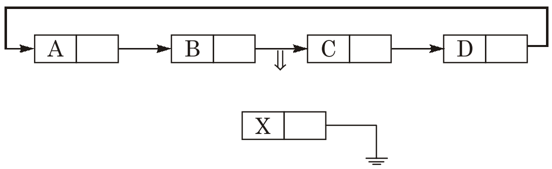
∴ The sequence of values popped out is 20, 20, 10, 10, 20.

3.2 (b)

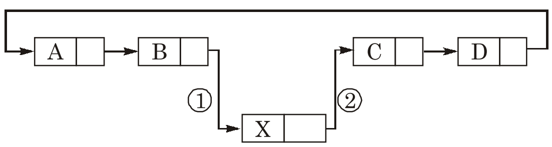
PUSH one by one element into Stack and POP accordingly desired output. Only second option can satisfy.

4. Linked List

4.1 (b)



In above linked list a new record X will be inserted in between B & C



Two pointers are modified to insert X record

4.2 (b)

Using linked list binary search will take O(n) time. So binary search is inefficient with linked list.

5. Trees

5.1 Sol.

False: For unique tree 'inorder' must be present along with pre-order or post-order.

5.2 Sol.

False: Tree may or may not be binary tree with given condition.

5.3 Sol.

g c b d a f e

7. Hashing

7.1 Sol.

Maximum no. of comparison can be 5.
Starting with cell no. 8 to 2.

■■■■

Algorithms

UNIT VI

CONTENTS

Algorithms

1. Algorithm Analysis and Asymptotic Notations

1.1 Solve the recurrence equations
$T(n) = T(n - 1) + n$
$T(1) = 1$

[1987 : 2 Marks]

1.2 What is the generating function G(z) for the sequence of Fibonacci numbers?

[1987 : 2 Marks]

1.3 $\sum_{1 \le k \le n} O(n)$, where O(n) stands for order n is
(a) $O(n)$
(b) $O(n^2)$
(c) $O(m^3)$
(d) $O(3n^2)$

[1993 : 2 Marks]

1.4 Consider the following two functions:

$$g_1(n) = \begin{cases} n^3 \text{ for } 0 \le n \le 10{,}000 \\ n^2 \text{ for } n \ge 10{,}000 \end{cases}$$

$$g_2(n) = \begin{cases} n \text{ for } 0 \le n \le 100 \\ n^3 \text{ for } n > 100 \end{cases}$$

Which of the following is true?
(a) $g_1(n)$ is $O(g_2(n))$
(b) $g_1(n)$ is $O(n^3)$
(c) $g_2(n)$ is $O(g_1(n))$
(d) $g_2(n)$ is $O(n)$

[1994 : 2 Marks]

2. Divide and Conquer

2.1 Let P be a quicksort program to sort numbers in ascending order. Let t1 and t2 be the time taken by the program for the inputs [1 2 3 4] and [5 4 3 2 1], respectively. Which of the following holds?
(a) $t_1 = t_2$
(b) $t_1 > t_2$
(c) $t_1 < t_2$
(d) $t_1 = t_2 + 5 \log 5$

[1987 : 2 Marks]

2.2 Find a solution to the following recurrence equation

$$T(n) = T\left(\frac{n}{2}\right) + n$$

$T(1) = 1$

[1987 : 2 Marks]

2.3 Assume that the last element of the set is used as partition element in Quicksort. If n distinct elements from the set [1 ... n] are to be sorted, give an input for which Quicksort takes maximum time.

[1992 : 2 Marks]

2.4 Following algorithm (s) can be used to sort n integers in the range [1 ... n^3] in O(n) time?
(a) Heap sort
(b) Quick sort
(c) Merge sort
(d) Radix sort

[1992 : 2 Marks]

2.5 The recurrence relation that arises in relation with the complexity of binary search is
(a) $T(n) = T(n/2) + k$, k is a constant
(b) $T(n) = 2T(n/2) + k$, k is a constant
(c) $T(n) = T(n/2) + \log n$
(d) $T(n) = T(n/2) + n$

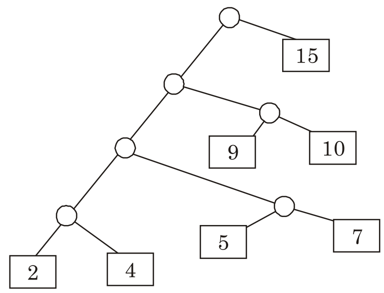
[1994 : 2 Marks]

2.6 Which one of the following statements is false?
(a) Optimal binary search tree construction can be performed efficiently using dynamic programming
(b) Breadth-first search cannot be used to find connected components of a graph
(c) Given the prefix and postfix walks over a binary tree, the binary tree cannot be uniquely constructed
(d) Depth-first search can be used to find connected components of a graph

[1994 : 2 Marks]

3. Greedy Method

3.1 The weighted external path length of the binary tree in figure is



[1991 : 2 Marks]

3.2 Kruskal's algorithm for finding a minimum spanning tree of a weighted graph G with n vertices and m edges has the time complexity of
(a) $O(n^2)$
(b) O(mn)
(c) O(m + n)
(d) O(mlogn)
(e) $O(m^2)$

[1991 : 2 Marks]

6. Miscellaneous Topics

6.1 Match the pairs in the following:

List-I
A. Straseen's matrix multiplication algorithm
B. Kruskal's minimum spanning tree algorithm
C. Biconnected components algorithm
D. Floyd's shortest path algorithm

List-II
P. Greedy method
Q. Dynamic programming
R. Divide and Conquer
S. Depth first search

[1990 : 2 Marks]

6.2 The minimum number of comparisons required to sort 5 elements is

[1991 : 2 Marks]

■■■■

Answers Algorithms

1.1	Sol.	**1.2**	Sol.	**1.3**	(b)	**1.4**	(a)	**2.1**	(c)	**2.2**	Sol.	**2.3**	Sol.	**2.4**	(d)	**2.5**	(a)
2.6	(b)	**3.1**	Sol.	**3.2**	(d)	**6.1**	Sol.	**6.2**	Sol.								

Explanations Algorithms

1. Algorithm Analysis and Asymptotic Notations

1.1 Sol.

By using substitution method we get following series:
$n + (n-1) + (n-2) + (n-3) \dots 3 + 2 + 1$
Which is sum of 'n' natural numbers.

$$= \frac{n(n+1)}{2} \Rightarrow O(n^2)$$

1.2 Sol.

The generating function for the Fibonacci numbers G(z) is

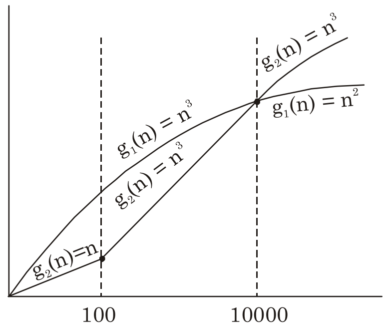
$$G(z) = \frac{z}{1 - z - z^2}$$

1.3 (b)

$$\sum_{1 \le k \le n} O(n) = O(1) + O(2) + O(3) + \dots + O(n)$$

$$= O\left(\frac{n(n+1)}{2}\right) = O(n^2)$$

1.4 (a)



Therefore;

$$n^2 \leq n^3 \text{ for } N \geq 10000$$

$$g_1(n) = O(g_2(n))$$

Option (a) is correct.

2. Divide and Conquer

2.1 (c)

t_1 is less than t_2

Sorting [1 2 3 4] takes less time compared to sorting [5 4 3 2 1] list. First list has 4 elements and second list has 5 elements. Both lists are worst cases of quick sort.

2.2 Sol.

By using substitution method we get following series:

$$\Rightarrow \quad T\left(\frac{n}{2^k}\right) + \frac{n}{2^{k-1}} + \frac{n}{2^{k-2}} \cdots \frac{n}{2^2} + \frac{n}{2^n} + n$$

$\Rightarrow$ So put $k = \log_2 n$ then we get

$$T\left(\frac{n}{2^{\log_2 n}}\right) + n\left(\frac{1}{2^0} + \frac{1}{2^1} + \frac{1}{2^2} \cdots \frac{1}{2^{\log_2 n-1}}\right) = O(n)$$

2.3 Sol.

Quicksort gives worst case when all elements are already sorted. So give any sorted order input either in ascending or descending order.

2.4 (d)

Radix sort is a non-comparative integer sorting algorithm that sorts data with integer keys by grouping keys which share same position and value. So it take O(n) time.

2.5 (a)

Binary search only half of the array.

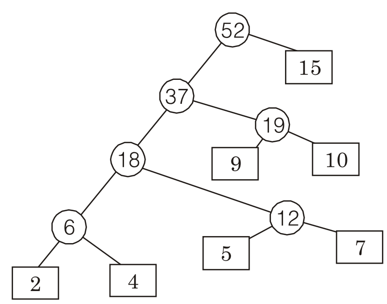
So, $T\left(\frac{n}{2}\right) + K$

2.6 (b)

Connected components of a graph can be computed in linear time by using either breadth-first search or depth-first search.

3. Greedy Method

3.1 Sol.



So the path length is 52.

3.2 (d)

Kruskal's algorithm time complexity

$$= O(e \log v) = O(m \log n)$$

6. Miscellaneous Topics

6.1 Sol.

A-R, B-P, C-S, D-Q

6.2 Sol.

Use either selection or insertion sort, minimum comparisons is 4.

■■■■

Compiler Design

UNIT

VII

CONTENTS

Compiler Design

1. Lexical Analysis

1.1 In a compiler the module that checks every character of the source text is called
(a) The code generator
(b) The code optimizer
(c) The lexical analyzer
(d) The syntax analyzer

[1988 : 1 Mark]

1.2 Match the followings:

Group-I
A. Lexical analysis
B. Code optmization
C. Code generation
D. Abelian groups

Group-II
P. DAG's
Q. Syntax trees
R. Push down automation
S. Finite automaton

[1990 : 1 Mark]

2. Parsing Techniques

2.1 Consider the SLR(1) and LALR (1) parsing tables for a context free grammar. Which of the following statements is/are true?
(a) The goto part of both tables may be different.
(b) The shift entries are identical in both the tables.
(c) The reduce entries in the tables may be different.
(d) The error entries in the tables may be different.

[1992 : 2 Marks]

3. Syntax Directed Translation

3.1 Generation of intermediate code based on an abstract machine model is useful in compilers because
(a) it makes implementation of lexical analysis and syntax analysis easier
(b) syntax-directed translations can be written for intermediate code generation
(c) it enhances the portability of the front end of the complier
(d) it is not possible to generate code for real machines directly from high level language programs

[1994 : 1 Mark]

■■■■

Answers Compiler Design

1.1 (c) **1.2** Sol. **2.1** (b, c, d) **3.1** (a)

Explanations Compiler Design

1. Lexical Analysis

1.1 (c)

Lexical analysis is a phase of compiler in which a sequence of characters get converted into a group of tokens. A program or function which performs lexical analysis is called a lexical analyzer. So lexical analyzer checks every character of the source text.

1.2 Sol.

(A)–(S), (B)–(P), (C)–(R), (D)–(Q)

2. Parsing Techniques

2.1 (b, c, d)

The difference in parsing tables for context free grammar of SLR & LALR parser is in reduce entires only. Due to difference in reduce entries there may be difference in error entries also.

3. Syntax Directed Translation

3.1 (a)

Generation of intermediate code based on an abstract machine model is useful in compilers because it makes implementation of lexical analysis and syntax analysis easier.

■■■■

Operating System

UNIT

VIII

CONTENTS

Operating System

1. Process Management-I
(Introduction, Process, Threads & CPU Scheduling)

1.1 State an undesirable characteristic of each of the following criteria for measuring performance of an operating system:
(a) Turn around time
(b) Waiting time

[1988 : 2 Marks]

1.2 The highest-response ratio next scheduling policy favours .... jobs, but it also limits the waiting time of .... jobs.

[1990 : 2 Marks]

1.3 Assume that the following jobs are to be executed on a single processor system

Job Id	CPU Brust time
p	4
q	1
r	8
s	1
t	2

The jobs are assumed to have arrived at time 0^+ and in the order p, q, r, s, t. Calculate the departure time (completion time) for job p if scheduling is round robin with time slice 1.
(a) 4 (b) 10
(c) 11 (d) 12

[1993 : 2 Marks]

2. Process Management-II
(IPC, Synchronization and Concurrency)

2.1 A critical region is
(a) One which is enclosed by a pair of P and V operations on semaphores.
(b) A program segment that has not been proved bug-free
(c) A program segment that often causes unexpected system crashes
(d) A program segment where shared resources are accessed.

[1987 : 2 Marks]

2.2 Semaphore operations are atomic because they are implemented within the OS......

[1990 : 2 Marks]

2.3 At a particular time of computation the value of a counting semaphore is 7. Then 20 P operations and 15 V operations were completed on this semaphore. The resulting value of the semaphore is:
(a) 42 (b) 2
(c) 7 (d) 12

[1992 : 2 Marks]

3. Deadlock

3.1 A computer system has 6 tape drives, with n process completing for them. Each process may need 3 tape drives. The maximum value of n for which the system is guaranteed to be deadlock free is
(a) 2 (b) 3
(c) 4 (d) 1

[1992 : 2 Marks]

3.2 Consider a system having m resources of the same type. These resources are shared by 3 processes A, B and C, which have peak demands of 3, 4 and 6 respectively. For what value of m deadlock will not occur?
(a) 7 (b) 9
(c) 10 (d) 13

[1993 : 2 Marks]

4. Memory Management and Virtual Memory

4.1 Under paged memory management scheme simple lock and key memory protection arrangement may still be required if the _________ processors do not have address mapping hardware.

[1990 : 2 Marks]

4.2 Match the pairs in the following questions.

(a) Critical region (p) Hoare's monitor
(b) Wait/Signal (q) Mutual exclusion
(c) Working set (r) Principle of locality
(d) Deadlock (s) Circular wait

[1990 : 2 Marks]

4.3 State whether the following statements are TRUE or FALSE with reason. The Link-load-and-go loading scheme required less storage space than the Link-and-go loading scheme.

[1990 : 2 Marks]

4.4 A "link editor" is a program that:

(a) Matches the parameters of the macro definition with locations of the parameters of the macro call.
(b) Matches external names of one program with their location in other programs.
(c) Matches the parameters of subroutine definition with the location of parameters of subroutine call.
(d) Acts as link between text editor and the user.
(e) Acts as a link between compiler and user program.

[1991 : 2 Marks]

4.5 Indicate all the false statements from the statements given below:

(a) The amount of virtual memory available is limited by the availability of secondary storage.
(b) Any implementation of a critical section requires the use of an indivisible machine-instruction, such as test-and-set.
(c) The LRU page replacement policy may cause hashing for some type of programs.
(d) The best fit techniques for memory allocation ensures the memory will never be fragmented.

[1991 : 2 Marks]

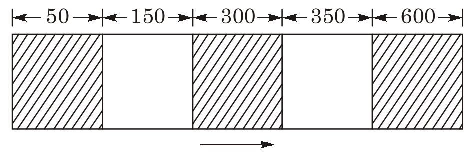
4.6 Which page replacement policy sometimes leads to more page faults when size of memory is increased?

(a) Optimal
(b) LRU
(c) FIFO
(d) None of these

[1992 : 1 Mark]

4.7 Consider the following heap (figure) in which blank regions are not in use and hatched regions are in use.

The sequence of requests for blocks of size 300, 25, 125, 50 can be satisfied if we use



(a) either first fit or best fit policy (any one)
(b) first fit but not best fit policy
(c) best fit but not first fit policy
(d) none of the above

[1994 : 2 Marks]

5. File System and Device Management

5.1 Disk requests come to disk driver for cylinders 10, 22, 20, 2, 40, 56 and 38 in that order at a time when the disk drive is reading from cylinder 20. The seek time is 6 msec per cylinder. Compute the total seek time if the disk arm scheduling algorithm is

(a) First come first served
(b) Closest cylinder next

[1989 : 2 Marks]

5.2 State whether the following statements are TRUE or FALSE with reason. The data transfer between memory and I/O devices using programmed I/O is faster than interrupt-driven I/O. **[1990 : 2 Marks]**

5.3 The root directory of a disk should be placed

(a) at a fixed address in main memory
(b) at a fixed location on the disk
(c) any where on the disk
(d) at a fixed location on the system disk

[1993 : 2 Marks]

6. Miscellaneous

6.1 Match the pairs in the following questions.

Pair-I

A. Virtual Memory
B. Shared memory
C. Look-ahead buffer
D. Look-aside buffer

Pair-II

P. Temporal locality
Q. Spatial locality
R. Address translation
S. Mutual exclusion

[1989 : 2 Marks]

6.2 Match the pairs in the following question by writing the corresponding letters only.

(a) Buddy system	(p) Run-time type specification
(b) Interpretation	(q) Segmentation
(c) Pointer type	(r) Memory allocation
(d) Virtual memory	(s) Garbage collection

[1991 : 2 Marks]

■■■■

Answers Operating System

1.1	Sol.	**1.2**	Sol.	**1.3**	(c)	**2.1**	(d)	**2.2**	Sol.	**2.3**	(b)	**3.1**	(a)	**3.2**	(d)	**4.1**	Sol.
4.2	Sol.	**4.3**	Sol.	**4.4**	(e)	**4.5**	(b, d)	**4.6**	(c)	**4.7**	(b)	**5.1**	Sol.	**5.2**	Sol.	**5.3**	(d)
6.1	Sol.	**6.2**	Sol.														

Explanations Operating System

1. Process Management-I (Introduction, Process, Threads & CPU Scheduling)

1.1 Sol.

Undesirable characteristics of

(a) Turn around time: Burst time of the process should not be too long.

(b) Waiting time: Arrival time of the processes should not be same.

1.2 Sol.

Shorter job, longer jobs.

1.3 (c)

Create the gantt chart with Round-Robin scheduling.

0	1	2	3	4	5	6	7	8	9	10	11
p	q	r	s	t	p	r	t	p	r	p	...

2. Process Management-II (IPC, Synchronization and Concurrency)

2.1 (d)

A critical region is a program segment where shared resources are accessed.

2.2 Sol.

Kernel.

2.3 (b)

Initially semaphore value is 7 then apply down and up operation. So first 20 down so –13, then 15 up so at last 2.

3. Deadlock

3.1 (a)

If there are 2 processes then each process will hold 3 tape drives as there are 6 tape drives for which the system is guaranteed to be deadlock free.

3.2 (d)

If anyone peak demand satisfied then we can't get deadlock.

4. Memory Management and Virtual Memory

4.1 Sol.

Input/output

4.2 Sol.

a-q, b-p, c-r, d-s

4.3 Sol.

False because in link load and go scheme some portion of memory is occupied by assembler which is simply wastage of memory. As this scheme is combination of assembler and loader activities so it uses large amount of memory.

4.4 (e)

Link editor is another name of 'linker'. Linker is a program which links the object code with it's library.

4.5 (b, d)

(b) Without indivisible machine instruction, critical section can be implemented like using monitors.

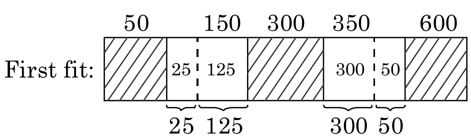
(d) Best fit also suffers from fragmentation.

4.6 (c)

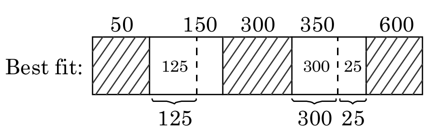
It's statement of Bleady's Anamoly. More frames in FIFO = More page faults.

4.7 (b)

Requests: 300, 25, 125, 50



⇒ First fit can satisfy all the requests.

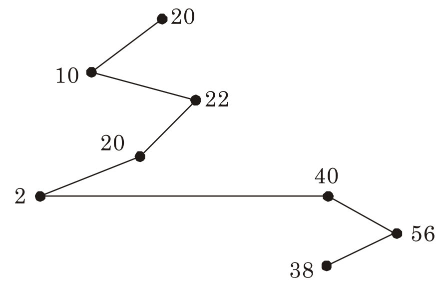


⇒ 50 can not be satisfied by best fit

5. File System and Device Management

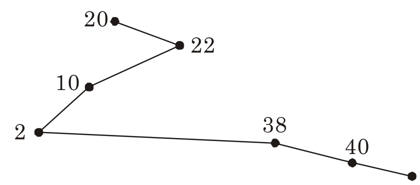
5.1 Sol.

(a) FCFS



$6 \times (10 + 12 + 2 + 18 + 38 + 16 + 18) = 684$

(b) Closest cylinder next



$6 \times (2 + 12 + 8 + 36 + 2 + 16) = 456$

5.2 Sol.

Statement is false, because in programmed I/O the processor has to wait a long time for the I/O module of concern to be reads for either reception or transmission of data; while in interrupt driven I/O CPU is in waiting state when interrupt comes. So takes less time to transfer the data and in interrupt driven I/O, the interface between memory and I/O works like a CPU.

5.3 (d)

The root directory of a disk is placed at a fixed location on the system disk.

6. Miscellaneous

6.1 Sol.

A-Q, B-S, C-P, D-R

6.2 Sol.

a-r, b-p, c-s, d-q

■■■■

Databases

UNIT

IX

CONTENTS

Databases

2. Database Design: Functional Dependencies and Normalization

2.1 State True or False with reason
There is always a decomposition into Boyce-Codd normal form (BCNF) that is lossless and dependency preserving.
[1994 : 1 Mark]

4. Relational Model: Relational Algebra and Tuple Calculus

4.1 An instance of a relational scheme R(A, B, C) has distinct values for attributes A. Can you conclude that A is a candidate key for R?
[1994 : 1 Mark]

4.2 Give a relational algebra expression using only the minimum number of operators from $\{\cup, -\}$ is equivalent to $R \cap S$.
[1994 : 1 Mark]

■■■■

Answers Databases

2.1 Sol. **4.1** Sol. **4.2** Sol.

Explanations Databases

2. Database Design: Functional Dependencies and Normalization

2.1 Sol.

False
Sometimes, there is no decomposition into BCNF that is dependency-preserving.
For example: Consider the relation schema ABD, if we have the FD's
$AB \rightarrow D$, $D \rightarrow B$
Then ABD is not in BCNF because D is not a key. If we try to decompose it, however, we can't preserve the dependency $AB \rightarrow D$.

4. Relational Model: Relational Algebra and Tuple Calculus

4.1 Sol.

No
For given instance "A" values unique doesn't mean candidate key. Because candidate key identification is based on constraints of the DB design and is not based on particular instance.

4.2 Sol.

$R \cap S = R - (R - S)$

■■■■