GATE
COMPUTER SCIENCE & IT

SOLVED PAPERS
2001 - 2015

- ✓ Topic wise Solutions with memorizing tips
- ✓ Descriptive Solutions with Alternate method
- ✓ Easy to understand
- ✓ Memorizing tips

GATE 2015Cut-off Marks				
BRANCH	**GENERAL**	**SC/ST/PD**	**OBC(Non-Creamy)**	**Total Appeared**
Computer Sc. & IT	25.00	16.67	22.50	115425

Contents

Chapter : 7 DBMS

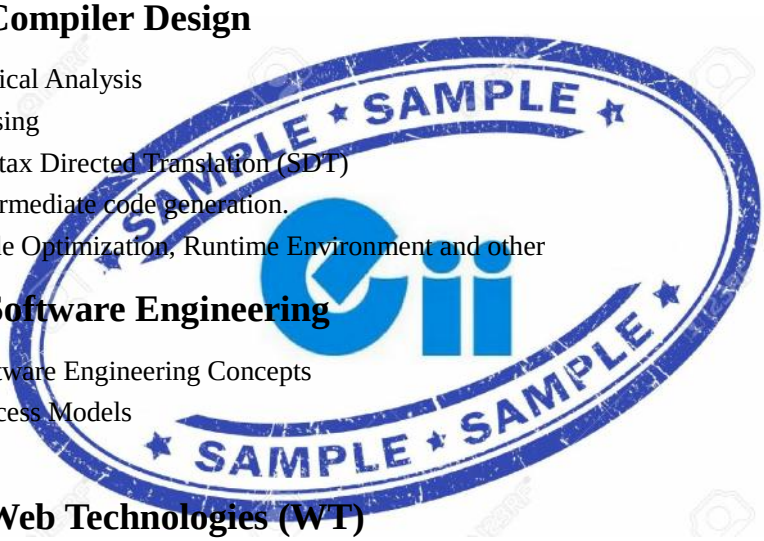
1. ER Model
2. Functional Dependencies and Normalization
3. SQL
4. Relational Algebra and Calculus
5. Transactions and Concurrency Control
6. File Structures and Indexing

Chapter : 8 Theory of computation

1. Regular Language and F.A
2. CFL and PDA
3. CSL, REC, RE and TM
4. Decidability and NP – Completeness

Chapter : 9 Compiler Design

1. Lexical Analysis
2. Parsing
3. Syntax Directed Translation (SDT)
4. Intermediate code generation.
5. Code Optimization, Runtime Environment and other



Chapter : 10 Software Engineering

1. Software Engineering Concepts
2. Process Models

Chapter : 11 Web Technologies (WT)

Chapter : 12 Discrete Mathematics

1. Mathematical Logic
2. Combinatorics
3. Graph Theory
4. Set Theory & Algebra
5. Linear Algebra

Chapter : 13 Computer Networks

1. Introduction, Physical Later, DLL
2. Network Layer (IP)
3. Transport Layer (TCP, UDP)
4. Upper Layers, Network Security

Chapter : 1 C-Programming

C Programming

1. The output of the following C program is _____ **GATE-2015 (SET-1)**

```
void f1(int a, int b)  {
 int c;
 c=a; a=b; b=c;
}
void f2 (int *a, int*b)  {
 int c;
 c=*a;  *a=*b;  *b=c;
}
int main () {
 int a=4, b=5, c=6;
 f1(a,b);
 f2(&b, &c);
 printf ("%d", c-a-b);
}
```


2. What is the output of the following C code? Assume that the address of x is 2000 (in decimal) and an integer requires four bytes of memory. **GATE-2015 (SET-1)**

```
int main () {
    unsigned int x[4][3] =
      {{1,2,3},{4,5,6},{7,8,9},{10,11,12}};
    printf("%u, %u, %u", x+3, *(x+3), *(x+2)+3);
```

(a) 2036, 2036, 2036 (b) 2012, 4, 2204 (c) 2036, 10, 10 (d) 2012, 4, 6

3. Consider the following function written in the C programming language. **GATE-2015 (SET-2)**

```
void foo (char *a) {
    if ( *a && *a != ' ') {
       foo(a+1);
       putchar(*a);
    }
     }
```

The output of the above function on input "ABCD EFGH " is

(a) ABCD EFGH (b) ABCD (c) HGFE DCBA (d) DCBA

4. Consider the following C function. **GATE-2015 (SET-2)**

```
int fun(int n){
  int x=1, k;
  if (n=1)return x;
  for (k=1; k<n; ++k)
     x = x + fun(k) * fun(n-k);
  return x;
}
```

The return value of fun (5) is _____

5. Consider the C program below. **GATE-2015 (SET-2)**

```
#include <stdio.h>
int *A, stkTop;
int stkFunc(int opcode, int val)
{
   static int size=0, stkTop=0;
   switch (opcode) {
      case -1: size = val; break;
      case 0: if (stkTop < size) A[stkTop++] = val; break;
      default: if (stkTop) return A[--stkTop];
   }
   return -1;
}
int main()
{
   int B[20]; A = B; stkTop = -1;
   stkFunc (-1, 10);
   stkFunc (0, 5);
   stkFunc (0, 10);
   printf ("%d\n", stkFunc(1, 0) + stkFunc(1, 0));
}
```

The value printed by the above program is _____.

6. Consider the following C program. **GATE-2015 (SET-3)**

```
# include <stdio.h>
int f1(void);
int f2 (void);
int f3(void);
```

```
int x = 10;

int main ( )
{
   int x= 1;
   x += f1( ) + f2( ) + f3( ) + f2( );
   printf (“%d”, x);
   return 0;
}

int f1( ) { int x = 25; x++; return x;}
int f2( ) {static int x = 50; x++; return x;)
int f3( ) {x *= 10; return  x};
```

The output of the program is _____

7. Consider the following C program. **GATE-2015 (SET-3)**

```
# include<stdio.h>
int main ( )
{
   static int a[ ] = {10, 20, 30, 40, 50};
   static int *p[ ] = {a, a+3, a+4, a+1, a+2};
   int **ptr = p;
   ptr++;
   printf(“%d%d”, ptr-p, **ptr);
}
```

The output of the program is _____.

……(Sample file here) …… Question Continue ……..

Answer Key & Descriptive Solutions

Ans : 1 Numerical answer : **–5**

EXP:

```
int main (){
   int a = 4, b=5, c=6;
   f1(a,b);
   f2(&b,&c);
   printf (“%d,”c-a-b);
```

```
void f1 (inta, intb){
   intc;
   c=a; a=b; b=c;
}

void f2 (int*a,int*b)
{
   intc;
   c=*a; *a=*b; *b=c;
}
```

a	b	c
4	~~5~~ 6	~~6~~ 5
1000	2000	3000

When the function f1(a,b) is called, values of a, b are passed

So,

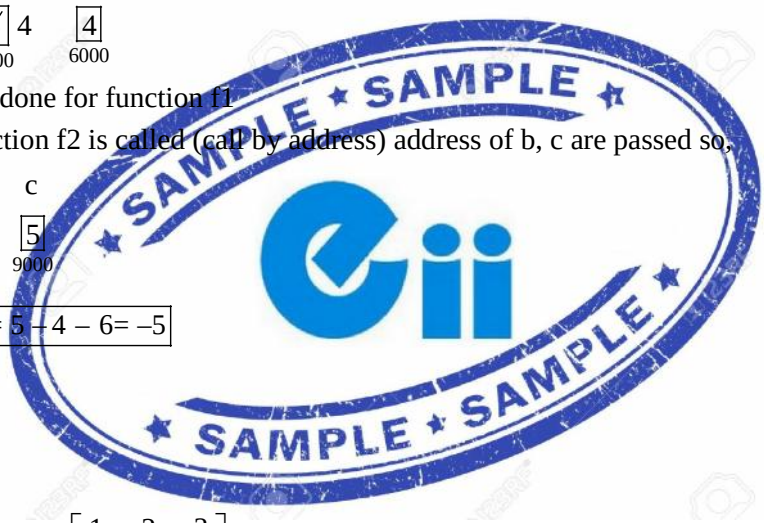
a	b	c
5 ~~4~~	~~5~~ 4	4
4000	5000	6000

call by value is done for function f1

now, when function f2 is called (call by address) address of b, c are passed so,

a	b	c
2000	3000	5
7000	8000	9000

Now (c–a–b) = 5 – 4 – 6= –5



Ans : 2 a

EXP: int X [4] [3] = $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \end{bmatrix}$ is a 2D array

%μ, X +3 indicates → skipping 3 row's from the base address but the 4th row is not yet selected as there is no * operator involved

so, 9 elements are skipped and every element is of 4 bytes

$2000 + 9 \times 4 = 2036$

%μ, *(X +3) indicates skipping three rows and the 4th row is selected which also points to the some address 2036

%μ, *(X + 2)+3 indicates skipping two rows, third row is selected and then skipping 3 columns but the elements are not yet selected as further no * operator is involved.

i.e., $2000 + (6\times4) + (3\times4) = 2036$ Hence, option (a) is correct.

Ans : 3 d

EXP: The program prints all characters before ' ' or '/0' (whichever comes first) in reverse order.

Hence, the correct option is (d) DCBA

Ans : 4 Numerical answer : **51**

EXP:

fun(5) = 1 + fun(1) × fun(4) + fun(2) × fun(3) + fun(3) × fun(2) + fun(4) × fun(1)

= 1 + 2 × [fun(1) × fun(4) + fun(2) × fun(3)] …(1)

Putting the value of fun(1) = 1

The equation (1) becomes

1 + 2 × [fun(4) + fun(2) × fun(3)]

Now calculating the value of fun(2), fun(3) and f(4) respectively,

fun(2) = 1 + fun(1) × fun(1) = 1 + 1 × 1 = 2

fun(3) = 1 + 2 × fun(1) × fun(2) = 1 + 2 × 1 × 2 = 5

fun(4) = 1 + 2 × fun(2) × fun(3) + fun(2) × fun(2) = 1 + 2 × 1 × 5 + 2 × 2 = 15

Substituting these values in equation (1)

fun(5) = 1 + 2 × [15 + 2 × 5] = 51

Ans : 5 Numerical answer : **15**

EXP: The code in main program, basically initializes a stack of size 10, and then 5 is pushed on to the stack and then 10 is pushed.

Finally, the printf statement prints sum of two pop operations which is: 10 + 5 = 15

```
Stk fun c (-1, 10) → Initialize size as 10
Stk fun c (0, 5) →  pushes 5
Stk fun c( 0, 10) → pushes 10
Printf("%d\h", stkfunc(1,0) + stkfun c(1,0); →
```

Prints the sum of two pop operations which is 15.

Ans : 6 Numerical answer : **230**

EXP:

```
x + = f1( ) + f2 + f3 ( ) + f2 ( );
x = x + f1 ( ) + f2( ) + f3 ( ) + f2 ( );
f1( ) returns 26
f2 ( ) returns 51
f3 ( ) returns 100
second call to f2 ( ) returns 52
(x is static  in f2( ))
x = 1+26+51+100+52 = 230
```

Ans : 7 Numerical answer : **140**

EXP:

```
int main ( )
{
static int a [ ] = { 10, 20, 30, 40, 50}
static int * P[ ] = {a, a+3, a+4, a+1, a+2}
int **p = p;
ptr ++;
print + ("%d, %d ", ptr-p, ** ptr);
```

	1000	1002	1004	1006	1008
$\rightarrow a$	10	20	30	40	50

	2000	2004	2008	2012	2016
$\rightarrow p$	1000	1006	1008	1002	1004

2000

ptr

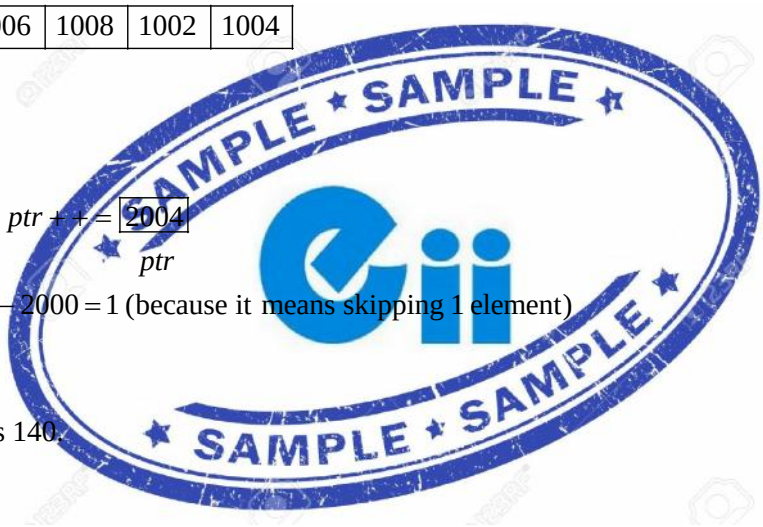
$ptr++ = 2004$

ptr

$ptr - p = 2004 - 2000 = 1$ (because it means skipping 1 element)

$** ptr = 40$

So, the output is 140.



Chapter : 2 Data Structures

Arrays

1. Consider the following two C code segments. Y and X are one and two dimensional arrays of size n and n × n respectively, where $2 \le n \le 10$ .Assume that in both code segments, elements of Y are initialized to 0 and each element `X[i][j]` of array X is initialized to `i+j`. Further assume that when stored in main memory all elements of X are in same main memory page frame.

Code segment 1: **GATE-2015 (SET-3)**

```
// initialize elements of y to 0
// initialize elements of X[i][j] of x to i+j
    For(i = 0; i < n; i++)
        Y[i] += X[0][i];
```

Code segment 2:

```
// initialize elements of y to 0
// initialize elements of X[i][j] of x to i+j
    For(i = 0; i < n; i++)
        Y[i] += X[i][0];
```

Which of the following statements is/are correct?

S1: Final contents of array y will be same in both code segments

S2: Elements of array X accessed inside the for loop shown in code segment 1 are contiguous in main memory

S3: Elements of array X accessed inside the for loop shown in code segment 2 are contiguous in main memory.

(a) Only S2 is correct

(b) Only S3 is correct.

(c) Only S1 and S2 are correct.

(d) Only S1 and S3 are correct.

2. A program P reads in 500 integers in the range [0, 100] representing the scores of 500 students. It then prints the frequency of each score above 50. What would be the best way for P to store the frequencies? **GATE-2005 (SET-3)**

(a) An array of 50 numbers (b) An array of 100 numbers

(c) An array of 500 numbers (d) A dynamically allocated array of 550 numbers

3. A single array A[1..MAXSIZE] is used to implement two stacks. The two stacks grow from opposite ends of the array. Variables top1 and top2 (top1 < top2) point to the location of the topmost element in each of the stacks. If the space is to be used efficiently, the condition for "stack full" is **GATE-2004**

(a) (top 1 = MAXSIZE/2) AND (top 2 = MAXSIZE/2 + 1)

(b) top 1 + top 2 = MAXSIZE

(c) (top 1 = MAXSIZE/2) or (top 2 = MAXSIZE)

(d) top 1 = top 2 – 1

4. Two matrices M_1 and M_2 are to be stored in arrays A and B respectively. Each array can be stored either in row-major or column-major order in contiguous memory locations. The time complexity of an algorithm to compute $M_1 \times M_2$ will be **GATE-2004**

(a) Best if A is in row-major, and B is in column-major order
(b) Best if both are in row-major order
(c) Best if both are in column-major order
(d) Independent of the storage scheme

……(Sample file here) …… Question Continue ……..

Stacks and Queues

5. The result evaluating the postfix expression 10 5 + 60 6 / * 8 – is
(a) 284 (b) 213 (c) 142 (d) 71 **GATE-2015 SET-3**

6. Suppose a stack implementation supports an instruction REVERSE, which reverses the order of elements on the stack, in addition to the PUSH and POP instructions. Which one of the following statements is TRUE *with respect to this modified stack?* **GATE-2014**

(a) A queue cannot be implemented using this stack.

(b) A queue can be implemented where ENQUEUE takes a single instruction and DEQUEUE takes a sequence of two instructions.

(c) A queue can be implemented where ENQUEUE takes a sequence of three instructions and DEQUEUE takes a single instruction.

(d) A queue can be implemented where both ENQUEUE and DEQUEUE take a single instruction each.

7. Consider the following operation along with Enqueue and Dequeue operations on queues, where k is a global parameter **GATE-2013**

```
MultiDequeue (Q) {
      m = k
      while (Q is not empty) and (m > 0) {
         Dequeue (Q)
         m = m – 1
      }
}
```

What is the worst case time complexity of a sequence of n queue operations on an initially empty queue?
(a) $\Theta(n)$ (b) $\Theta(n + k)$ (c) $\Theta(nk)$ (d) $\Theta(n^2)$

Answer Key & Descriptive Solutions

Ans : 1 c

EXP: Basically, what this program is describing is that →

In C, 2D arrays are stored in row major order.

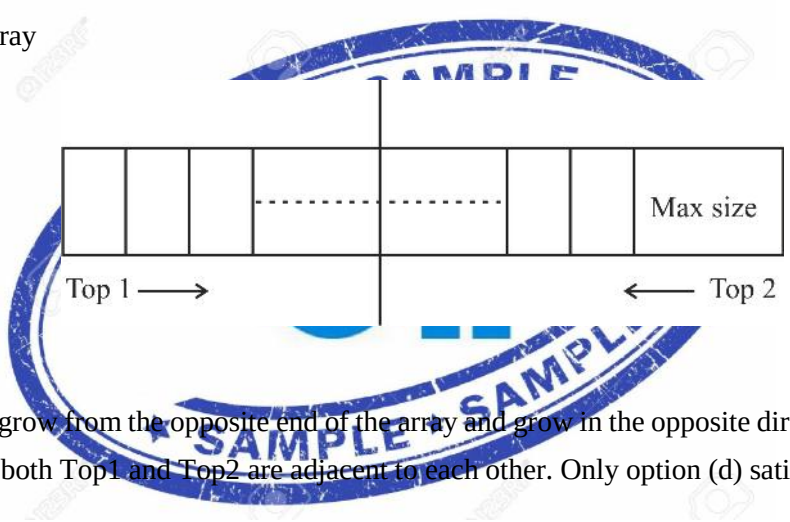
Therefore, S_2 is correct, but S_3 is not correct.

Ans : 2 a

EXP: An array of size 50 looks the best option to store number of students for each score. We need to store frequencies of scores above 50. We can ignore scores below 50 and to index the scores above 50.

Ans : 3 d

EXP: A single array



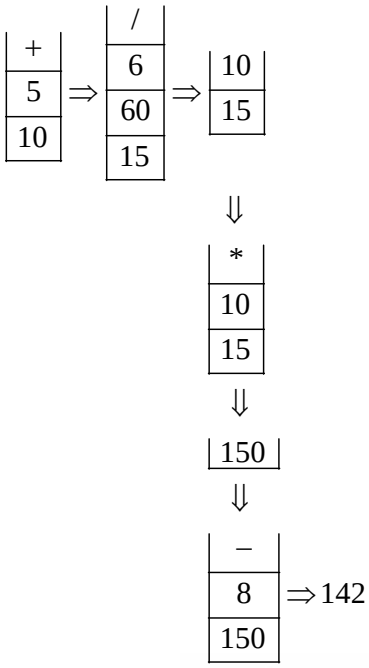
As both stack grow from the opposite end of the array and grow in the opposite direction so stack will be full only if both Top1 and Top2 are adjacent to each other. Only option (d) satisfies the condition.

Ans : 4 d

EXP: Since matrices are stored in array, complexity is independent of how they are stored. If the starting address of array is known, with indexes of elements their address can be calculated.

Ans : 5 c

EXP: 10 5 + 60 6 / * 8 –



Ans : 6 c

EXP: We can implement queue with the help of two stacks but if we have one stack with REVERSE operation then we can implement the queue operation.

For de-queue operation

(*i*) Pop the top of the stack

For Enqueue operation

(*i*) REVERSE (*ii*) PUSH (*iii*) REVERSE

So de-queue can be implemented by 1-stack operation and enqueue can be implemented by 3-stack operation option (*c*) is correct.

Ans : 7 c

EXP: In the stack and queue every operation takes 0(1) time. Here there are total *n* operations.

Let *a* be the en-queue operations

b be the de-queue operations

c be the multi-queue operation

$$a + b + c = n$$

All en-queue, de-queue, and multi-queue operations takes constant time so total time complexity is 0(*n*).

Chapter : 3 Algorithms (DAA)

Algorithm Analysis and Asymptotic Notations

1. Consider the following C function. **GATE-2015 SET-1**

```
int funl (int n) {
 int i,j,k,p,q=0;
 for(i=1; i<n; ++1)   {
    p=0;
    for (j=n; j>1; j=j/2)
       ++p;
    for (k=1; k<p; k=k*2)
       ++q;
 }
 return q;
}
```

Which one of the following most closely approximates the return value of the function `funl`?

(a) n^3 (b) $n (\log n)^2$ (c) $n \log n$ (d) $n \log (\log n)$

2. Consider a complete binary tree where the left and the right subtrees of the root are max-heaps. The lower bound for the number of operations to convert the tree to a heap is **GATE-2015 SET-1**

(a) $\Omega(\log n)$ (b) $\Omega(n)$ (c) $\Omega(n \log n)$ (d) $\Omega(n^2)$

3. Consider the equality $\sum_{i=0}^{n} i^3 = X$ and the following choices for X **GATE-2015 SET-3**

I. $\bar{}(n^4)$

II. $\bar{}(n^5)$

III. $o(n^5)$

IV. $\Omega(n^3)$

The equality above remains correct if X is replaced by

(a) Only I

(b) Only II

(c) I or III or IV but not II

(d) II or III or IV but not I

4. Consider the following recursive C function. **GATE-2015 SET-3**

```
Void get (int  n)
 {
   if  (n<1)  return;
   get (n–1) ;
   get (n–3) ;
   printf (“%d” ,  n) ;
 }
```

If `get (6)` function is being called in `main ( )` then how many times will the `get ( )` function be invoked before returning to the `main ( )` ?

(a) 15 (b) 25 (c) 35 (d) 45

5. Let f(n) =n and g(n) = $n^{(1+\sin n)}$, where n is a positive integer. Which of the following statements is/are correct? **GATE-2015 SET-3**

I. $f(n) = o(g(n))$

II. $f(n) = \Omega(g(n))$

(a) Only I

(b) Only II

(c) Only I and II

(d) Neither I nor II

Answer Key & Descriptive Solutions

Ans : 1 d

EXP:

```
    int fun 1(intn)}
int, i, j, k, p, q = 0;
for (i=1; i<n; i++)⇒ This loop runs θ(n) time
   }
    p =0
        for (j=n; j>1; j=j/2)⇒This loop runs logn time and P =
                                        logn
        for (k=1;k<p; k=k×2)⇒This loop also runs θ(Logn) time and
             k=log logn [p=logn], 2^k = logn, [k=loglogn]

   hence, overall time complexity is θ(nloglogn)
```

Ans : 2 a

EXP: Call Max-heapify procedure which recurses at most through the height of the heap i.e., Ω (logn)

Ans : 3 c

EXP:

$$\sum_{i=0}^{\infty} i^3 = X$$

X = sum of natural no. cubes.

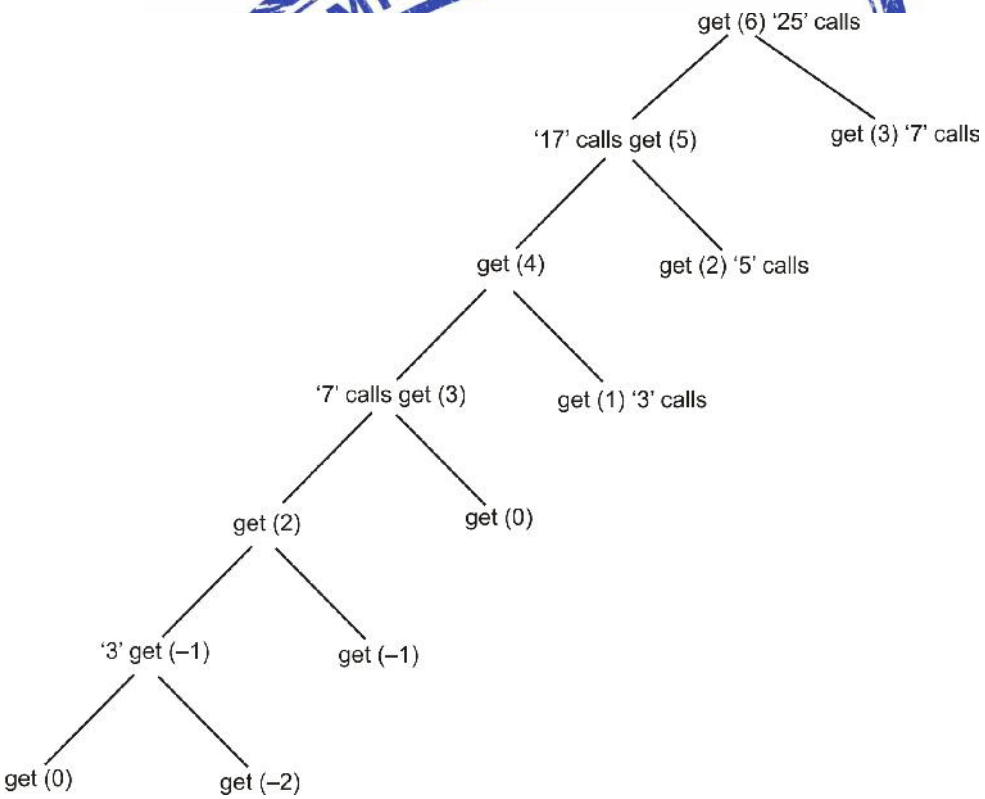
i.e. = $1^3 + 2^2 + 3^3 + 4^3 + ...$

$$= \left(\frac{n(n+1)}{2}\right)^2 = \frac{n^2(n+1)^2}{4} = O(n^4) \text{ or } O(n^5) \text{ or } \Omega(n^3)$$

i.e. option (c) is correct.

Ans : 4 b

EXP:



Total 25 calls

Ans : 5 d

EXP:

$f(n) = n$

$g(n) = n^{(1+\sin n)}$

Since, trigonometric function are non-comparable

$\therefore$ $f(n) = O(g(n))$ *and* $f(n) = \Omega(g(n))$ are both false.

The value of sine function varies from –1 to 1 for sin = 1 or other negative values. I becomes false. For sin = 1 or other positive values. II becomes false.

Chapter : 4 Digital Logic

1. Boolean Algebra and K-Maps

1. Consider the operations **GATE-2015 SET-1**

$f(X,Y,Z) = X'YZ + XY'+Y'Z'$ and $g(X,Y,Z) = X'YZ + X'YZ'+ XY$

Which one of the following is correct?

(a) Both {f} and {g} are functionally complete

(b) Only {f} is functionally complete

(c) Only {g} is functionally complete

(d) Neither {f} nor {g} is functionally complete

ANS: b

55. The number of min-terms after minimizing the following Boolean expression is _____.

$[D' + AB' + A'C + AC'D + A'C'D]'$ **GATE-2015 SET-2**

ANS: 1

40. Given the function F = P' + QR, where F is a function in three Boolean variables P, Q and R and P' = ! P, consider the following statements. **GATE-2015 SET-3**

(S1) F = $\sum(4, 5, 6)$

(S2) F = $\sum(0, 1, 2, 3, 7)$

(S3) F = $\prod(4, 5, 6)$

(S4) F = $\prod(0, 1, 2, 3, 7)$

Which of the following is true?

(a) (S1)- False, (S2)-True, (S3)-True, (S4)-False

(b) (S1)- True, (S2)-False, (S3)-False, (S4)-True

(c) (S1)- False, (S2)-False, (S3)-True, (S4)-True

(d) (S1)- True, (S2)-True, (S3)-False, (S4)-False

ANS: a

62. The total number of prime implicants of the function $f(w,x,y,z) = \sum(0,2,4,5,6,10)$ is _____.

GATE-2015 SET-3

ANS: 3

1. Consider the following Boolean expression for F: **GATE-2014: SET-1**

$F(P,Q,R,S) = PQ + \bar{P}QR + \bar{P}Q\bar{R}S$

The minimal sum of products form of F is

(a.) PQ + QR + QS (b.) P + Q + R + S (c.) $\bar{P}+\bar{Q}+\bar{R}+\bar{S}$ (d.) $\bar{P}R+\bar{P}\bar{R}S+P$

2. The dual of a Boolean function $F(x_1, x_2, \ldots, x_n, +, ., ')$ F^D, is the same expression as that of F with + and. Swapped. F is said to be self duel if $F = F^D$. The number of self dual functions with n Boolean variables is **GATE-2014: SET2**

(a.) 2^n (b.) 2^{n-1} (c.) 2^{2^n} (d.) $2^{2^{n-1}}$

3. Consider the following minterm expression for F: **GATE-2014: SET-3**

$$F(P,Q,R,S)=\sum 0,2,5,7,8,10,13,15$$

The minterms 2, 7, 8 and 13 are 'do not care' terms. The minimal sum – of – products form for F is

(a.) $Q\bar{S}+\bar{Q}S$ (b.) $\bar{Q}\bar{S}+QS$

(c.) $\bar{Q}\bar{R}\bar{S}+\bar{Q}R\bar{S}+Q\bar{R}S+QRS$ (d.) $PQ\bar{S}+PQS+P\bar{Q}S+P\bar{Q}\bar{S}$

4. A binary operation $\oplus$ on a set of integers is defined as $x \oplus y = x^2 + y^2$. Which one of the following statements is **TRUE** about $\oplus$? **GATE-2013**

(a) commutative but not associative (b) both commutative and associative

(c) associative but not commutative (d) neither commutative nor associative

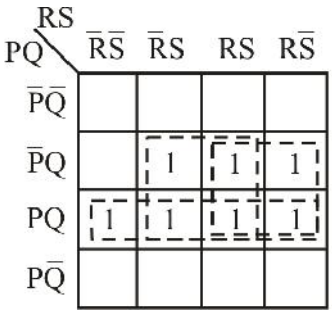
……(Sample file here) …… Question Continue ……..

Answer Key & Descriptive Solutions

Ans : 1 a

EXP:

$$F(P,Q,R,S)=PQ+\bar{P}QR+\bar{P}Q\bar{R}S$$



$F = PQ + QR + QS$

Ans : 2 d

EXP: For 'n' variables.

Total number of Boolean functions = 2^{2^n}

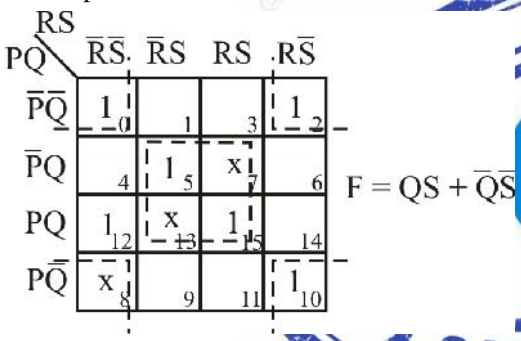
And number of self dual functions = $2^{2^{n-1}}$

Ans : 3 b

EXP: F (P, Q, R, S) = Σ (0, 2, 5, 7, 8, 10, 13, 15)

Where 2, 7, and 13 are don't care

K – map



$F = QS + \bar{Q}\bar{S}$

Ans : 4 a

EXP: Verifying commutative law:

$x \oplus y = x^2 + y^2 = y^2 + x^2 = y \oplus x \longrightarrow$ Commutative

Verifying associative law:

$$x \oplus (y \oplus z) = x \oplus (y^2 + z^2) = x^2 + (y^2 + z^2)^2 \quad ...(1)$$

$$(x \oplus y) \oplus z = (x^2 + y^2) \oplus z = (x^2 + y^2)^2 + z \quad ...(2)$$

From (1) and (2) it is clear that,

$x \oplus (y \oplus z) \neq (x \oplus y) \oplus z \longrightarrow$ Not Associative.

Chapter : 5 Computer Organization

Machine Instructions and Addressing Modes

1. For computers based on three-address instruction formats, each address field can be used to specify which of the following: **GATE-2015 SET-1**

(S1) A memory operand

(S2) A processor register

(S3) An implied accumulator register

(a) Either S1 or S2

(b) Either S2 or S3

(c) Only S2 and S3

(d) All of S1, S2 and S3

2. Consider a processor with byte-addressable memory. Assume that all registers, including Program Counter (PC) and Program Status Word (PSW), are of size 2 bytes. A stack in the main memory is implemented from memory location $(0100)_{16}$ and it grows upward. The stack pointer (SP) points to the top elements of the stack. The current value of SP is $(016E)_{16}$. The CALL instruction is of two words, the first word is the op-code and the second word is the starting address of the subroutine (one word = 2 bytes). The CALL instruction is implemented as follows: **GATE-2015 SET-2**

- Store the current value of PC in the stack
- Store the value of PSW register in the stack
- Load the string address of the subroutine in PC

The content of PC just before the fetch of a CALL instruction is $(5FA0)_{16}$. After execution of the CALL instruction, the value of the stack pointer is

(a) $(016A)_{16}$ (b) $(016C)_{16}$ (c) $(0170)_{16}$ (d) $(0172)_{16}$

3. A machine has a 32-bit architecture, with 1-word long instructions. It has 64 registers, each of which is 32 bit long. It needs to support 45 instructions, which have an immediate operand in addition to two register operands. Assuming that the immediate operand is an unsigned integer, the maximum value of the immediate operand is _____. **GATE : 2014**

4. Consider a hypothetical processor with an instruction of type LW R1, 20 (R2), which during execution

reads a 32-bit word from memory and stores it in a 32-bit register R1. The effective address of the memory location is obtained by the addition of constant 20 and the contents of register R2. Which of the following best reflects the addressing mode implemented by this instruction for the operand in memory?

GATE : 2011

(a.) Immediate Addressing

(b.) Register Addressing

(c.) Register indirect scaled addressing

(d.) Base indexed Addressing

5. A computer handles several interrupt sources of which ,the following are relevant for this question.

GATE : 2011

Interrupt from CPU temperature sensor (raises interrupt if CPU temperature is too high).

Interrupt from mouse (raises interrupt if the mouse is moved or a button is pressed).

Interrupt from keyboard (raises interrupt when a key is pressed or released).

Interrupt from hard disk (raises interrupt when a disk read is completed).

Which one of these will be handled at **HIGHEST** priority?

(a.) Interrupt from Hard Disk (b.) Interrupt from mouse

(c.) Interrupt from Keyboard (d.) Interrupt from CPU temperature sensor

……(Sample file here) …… Question Continue ……..

Answer Key & Descriptive Solutions

Ans : 1 a

EXP: ……(Sample file here) ……

Ans : 1 d

EXP: ……(Sample file here) ……

3. ANS: 16383

number of Registers: 64

number of instructions supported by machine: 45

Instruction size: 32 bit

32Bit

OPCODE	Register Operand 1	Register Operand 2	Immediate Operand
$\lceil Log_2 45 \rceil$	$Log_2 64$	$Log_2 64$	x
6	6	6	x

x = 32 – 18 = 14bits

Immediate operand is the operand which is available in the instruction itself.

number of bits for immediate operand = 14 bits

Maximum value of immediate operand = $2^{14} - 1 = 16383$

4. ANS: d

$\boxed{LW \quad R_1, \ 20(R_2)}$

Immediate Addressing:

In this mode, operand is specified in the instruction itself.

Register Addressing:

In this mode, operands are in registers that reside within CPU.

In the above instruction:

EA of memory: 20 + [R_2]

$R_1 \leftarrow M\left[20 + [R_2]\right]$

In indexed AM, content of index register is added to address part of the instruction, to obtain the effective address.

5. ANS: d

Interrupt from CPU temperature sensor will be handled at highest priority.

Chapter : 6 Operating System

CPU Scheduling

1. Consider a uniprocessor system executing three tasks T_1, T_2 and T_3, each of which is composed of an infinite sequence of jobs (or instances) which arrive periodically at intervals of 3, 7 and 20 milliseconds, respectively. The priority of each task is the inverse of its period, and the available tasks are scheduled in order of priority, with the highest priority task scheduled first. Each instance of T_1, T_2 and T_3, requires an execution time of 1, 2 and 4 milliseconds, respectively. Given that all tasks initially arrive at the beginning of the 1st millisecond and task preemptions are allowed, the first instance of T_3 completes its execution at the end of _____ milliseconds. **GATE-2015 SET-1**

2. For the processes listed in the following table, which of the following scheduling schemes will give the lowest average turnaround time? **GATE-2015 SET-3**

Process	Arrival time	Processing time
A	0	3
B	1	6
C	4	4
D	6	2

(a) First Come First Serve
(b) Non-preemptive shortest job First
(c) Shortest Remaining Time
(d) Round Robin with Quantum value two

3. Consider the following set of processes that need to be scheduled on a single CPU. All the times are given in milliseconds.

Process Name	Arrival Time	Execution Time
A	0	6
B	3	2
C	5	4
D	7	6
E	10	3

Using the shortest remaining time first scheduling algorithm, the average process turnaround time (in *m*sec) is _____. **GATE-2014**

2. Three processes A, B and C each execute a loop of 100 iterations. In each iteration of the loop, a process performs a single computation that requires t_c CPU milliseconds and then initiates a single I/O operation that lasts for t_{io} milliseconds. It is assumed that the computer where the processes execute has sufficient number of I/O devices and the OS of the computer assigns different I/O devices to each process. Also, the scheduling overhead of the OS is negligible. The processes have the following characteristics: **GATE-2014**

Process id	t_c	t_{io}
A	100ms	500ms
B	350ms	500ms

C 200ms 500ms

The processes A, B, and C are started at times 0, 5 and 10 milliseconds respectively, in a pure time sharing system (round robin scheduling) that uses a time slice of 50 milliseconds. The time in milliseconds at which process C would *complete* its first I/O operation is _____.

3. ***An operating system uses*** **shortest remaining time first** ***scheduling algorithm for pre-emptive scheduling of processes. Consider the following set of processes with their arrival times and CPU burst times (in milliseconds):*** **GATE-2014**

Process	Arrival Time	Burst Time
P1	***0***	***12***
P2	***2***	***4***
P3	***3***	***6***
P4	***8***	***5***

The average waiting time (in milliseconds) of the processes is _____.

……(Sample file here) …… Question Continue ……..

Answer Key & Descriptive Solutions

Operating System Solutions

Ans : 1 Numerical answer : 12

EXP:

Periods of T_1, T_2 and T_3 are 3ms, 7ms and 20ms

Since Priority is inverse of period, T_1 is the highest priority, then T_2 and finally T_3

Every instance of T_1 requires 1ms, that of T_2 requires 2ms and that of T_3 requires 4ms.

Initially all T_1, T_2 and T_3 are ready to get processor, T_1 is preferred.

Second instance of T_1, T_2 and T_3 shall arrive at 3, 7 and 20 units of time

Third instance of T_1, T_2 and T_3 shall arrive at 6, 14 and 49 respectively.

Time-Interval	Process
0 – 1	T_1
1 – 2	T_2
2 – 3	T_2
3 – 4	T_1 (second instance of T_1 arrives)
4 – 5	T_3
5 – 6	T_3
6 – 7	T_1 (Third instance of T_1 arrives)

7 – 8	T_2(Second instance of T_2 arrives)
8 – 9	T_2
9 – 10	T_1(Fourth instance of T_1 arrives)
10 – 11	T_3
11 – 12	T_3(First instance of T_3 arrives)

Ans : 2 c

EXP: Turnaround time is the total time token between the submission of a program for execution and the return of the complete output.

Turnaround time = Completion time – Arrival time

FCFS = First come first serve (A, B, C, D)

SJF = Non-pre emptive shortest job first (A, B, C, D)

SRT = Shortest remaining time first (A(3), B(1), C(4), D(2), B(5))

RR = Round Robin with time quantum 2

(A(2), B(2), A(1), C(2), B(2), D(2), C(2), B(2))

Ans : 3 Numerical answer : 7.2 m sec

EXP:

Process name	Arrival name	Execution time	Completion time	Turnaround time
A	0	6	8	8
B	3	2	5	2
C	5	4	12	7
D	7	6	21	14
E	10	3	15	5
				36

CPU Scheduling Algo:

Shortest remaining time First (SRTF)

A	B	A	A	C	C	E	D

0 3 5 7 8 10 12 15 21

Average T.A.T = $\frac{36}{5} = 7.2 m\text{sec}$

Ans : 4 Numerical answer : 1000

EXP:

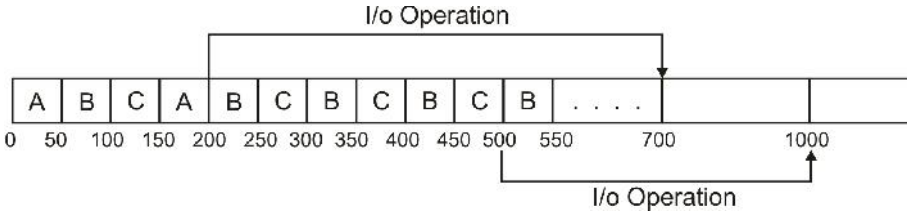
Process ID	Arrival Time	tc	t_{IO}
A	0ms	100ms	500ms

B	5ms	350ms	500ms
C	10ms	200ms	500ms

CPU scheduling Algo: Round Robin (RR)

Time Quantum: 50ms

Gantt Chart:



Ready Queue:

A, B, C, A, B, C, B, C, B, C, B.....

At t = 200, process A completes its computation and initiates its first I/O operation which completes at t = 700ms.

At t = 500, process C initiates its first I/O operation which completes at t = 1000ms. Note that I/O operations of different processes can be overlapped.

Whenever a process initiates I/O operation it changes its state from running state to wait state.

Ans : 5 Numerical answer : 5.5

EXP:

CPU Scheduling Algorithm: SRTF

Process	Arrival time	Burst time	Completion time	Completion time – arrival time	Waiting time
P_1	0	12	27	27	15
P_2	2	4	6	4	0
P_3	3	6	12	9	3
P_4	8	5	17	9	4
					22

P_1	P_2	P_3	P_3	P_4	P_1

0 2 6 8 12 17 27

Average Waiting Time: 22/4 = 5.5ms

Chapter : 7 DBMS

ER Model

1. Consider an Entity-Relationship (ER) model in which entity sets E_1 and E_2 are connected by an m:n relationship R_{12}. E_1 and E_3 are connected by a 1:n (1 on the side of E_1 and n on the side of E_3) relationship R_{13}.

E_1 has two single-valued attributes a_{11} and a_{12} of which a_{11} is the key attribute. E_2 has two single-valued attributes a_{21} and a_{22} of which a_{21} is the key attribute. E_3 has two single-valued attributes a_{31} and a_{32} of which a_{31} is the key attribute. The relationships do not have any attributes.

If a relational model is derived from the above ER model, then the minimum number of relations that would be generated if all the relations are in 3NF is _____. **GATE-2015 SET-I**

2. Given the basic ER and relational models, which of the following is **INCORRECT**?

GATE-2012

(a) An attribute of an entity can have more than one value

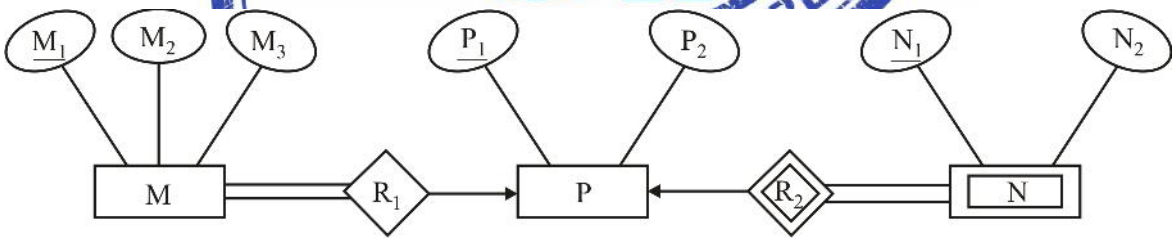
(b) An attribute of an entity can be composite

(c) In a row of a relational table, an attribute can have more than one value

(d) In a row of a relational table, an attribute can have exactly one value or a NULL value

Statement for Linked Answer Questions 3 and 4: **GATE-2008**

Consider the following ER diagram



3. The minimum number of tables needed to represent M, N, P, R1, R2, is

(a.) 2 (b.) 3 (c.) 4 (d.) 5

4. Which of the following is a correct attribute set for one of the tables for the correct answer to the above questions?

(a.) $\{M1, M2, M3, P1\}$ (b.) $\{M1, P1, N1, N2\}$

(c.) $\{M1, P1, N1\}$ (d.) {M1, P1}

5. Let E_1 and E_2 be two entities in an E/R diagram with simple single-valued attributes. R_1 and R_2 are two relationships between $E_1$and E_2, where R_1 is one-to-many and R_2 is many-to-many. R_1 and R_2 do not have any attributes of their own. What is the minimum number of tables required to represent this situation in the relational model? **GATE-2005**

(a.) 2 (b.) 3 (c.) 4 (d.) 5

……(Sample file here) …… Question Continue ……..

Answer Key & Descriptive Solutions

ER Model

Ans : 1 Numerical answer : 4

EXP: ……(Sample file here)

Ans : 2 c

EXP:

a. Multi valued attributes are allowed in ER model and is represented by double ellipses.

b. ER model allows composite attributes *e.g.* Name is a composite attribute with component attributes first – name, middle – name, last – name.

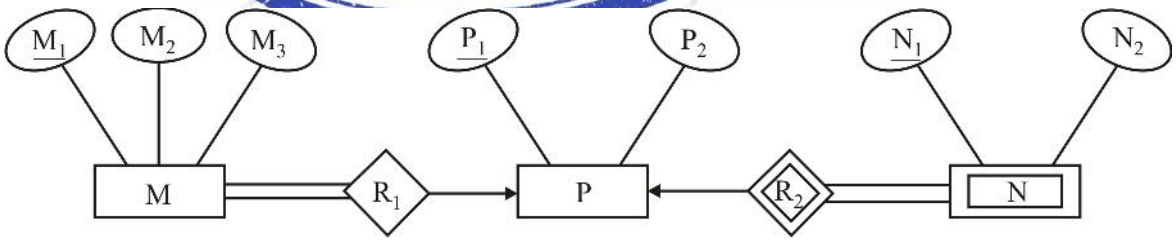
c. Multi valued attributes are not allowed in relational model.

d. Is true

C is incorrect statement.

Ans : 3 b

EXP:



Double rectangle represents weak entity set, double lines indicate total participation of an entity in a relationship set.

Strong entities are represented by separate tables, thus M, P are represented by separate tables.

Relation 1: R_1 (M_1, M_2, M_3, P_1)

Relation2: R_2 (P_1, P_2), N is a weak entity set.

Relation3: R_3(N_1, N_2, P_1)

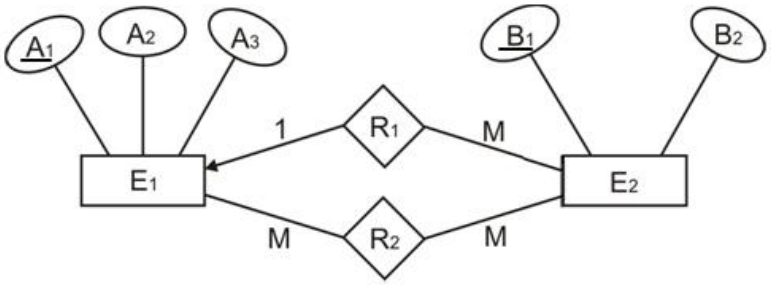
Thus 3 tables are required.

Ans : 4 a

EXP: ……(Sample file here)

Ans : 5 b

EXP:



E_1, E_2 are strong entities, thus must be represented as separate tables.

Relation1: R_1 (A_1, A_2, A_3)

Relation 2: R_2 (B_1, B_2, A_1)

M to M relationship R_2 must require a new table including P. keys of E_1, E_2

Relation3: R_3 (A_1, B_1)

Note:

In M to M relation: No merging is possible

In 1 to M relation: Merging is possible and 1 to M relation is transferred to M side table by taking primary key of 'one' side as foreign key.

In M to 1 relation: Merging is possible and M to 1 relation is transferred to M side table by taking primary key of 'one' side as foreign key.

In the above q also one to many relationship R_1 is transferred to 'many' side

Total 3 tables are required.

Table 1: $R_1(A_1, A_2, A_3)$

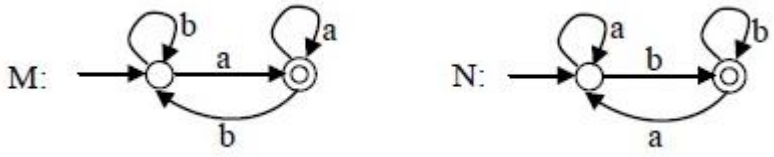
Table 2: $R_2(A_1, B_1, B_2)$

Table 3: R_3 (A_1, B_1)

Chapter : 8 Theory of computation

Regular Languages and Finite Automata

1.



Consider the DFAs M and N given above. The number of states in a minimal DFA that accepts the language L(M) $\cap$ L(N) is _____. **GATE-2015 SET-1**

ANS: 1

EXP: The language accepted by machine M is "strings ending with a"

The language accepted by machine N is "string ending with b"

The intersection between the two is ϕ as no string can end with both 'a' and 'b'.

so, for language = ϕ, the machine requires only one state.

Chapter : 9 Compiler Design

Lexical Analysis

1. The lexical analysis for a modern computer language such as java needs the power of which one of following machine model in a necessary and sufficient sense? **GATE: 2011**

(a.) Finite state automata (b.) Deterministic pushdown automata

(c.) Non-Deterministic pushdown automata (d.) Turing machine

2. In a complier, keywords of a language are recognized during **GATE: 2011**

(a.) Parsing of the program (b.) the code generation

(c.) The lexical analysis of the program (d.) dataflow analysis

3. Which data structure in a compiler is used for managing information about variables and their attributes? **GATE: 2010**

(a.) Abstract syntax tree (b.) Symbol table

(c.) semantic stack (d.) parse table

……(Sample file here) …… Question Continue ……..

Answer Key & Descriptive Solutions

1	2	3	4	5	6	7	8	9	10
a	**c**	**b**	**d**	**b**	**c**	**d**	**a**	**c**	**b**
11	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
c	**d**	**b**	**a**	**c**	**c**	**c**	**b**	**d**	**a**
21	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**
d	**a**	**a, b**	**d**	**b**	**b**	**d**	**b**	**d**	**a**
31	**32**	**33**	**34**	**35**	**36**	**37**	**38**	**39**	**40**
d	**c**	**b**	**c**	**c**	**b**	**b**	**a**	**c**	**a**
41	**42**	**43**	**44**	**45**	**46**	**47**	**48**	**49**	**50**
6	**d**	**a**	**d**	**b**	**b**	**c**	**b**	**b**	**b**
51	**52**	**53**	**54**	**55**					
a	**d**	**c**	**c**	**b**					

Compiler Design Solutions

1. (a)

Lexical analysis phase of the compiler is used to recognize tokens. For recognizing tokens, a finite automaton is used.

2. (c)

In compiler, keywords of a language are recognized during lexical analysis phase.

Purpose of lexical analysis is recognizing keywords, identifiers of a language.

3. (b)

Compiler uses symbol table for managing information about variables and their attributes

……(Sample file here) …… Question Continue ……..

Chapter : 12 Discrete Mathematics

1. Mathematical Logic

1. Which one of the following is NOT equivalent to p $\leftrightarrow$ q? **GATE-2015 SET-1**

(a) $(\neg p \vee q) \wedge (p \vee \neg q)$ (b) $(\neg p \vee q) \wedge (q \rightarrow p)$

(c) $(\neg p \wedge q) \vee (p \wedge \neg q)$ (d) $(\neg p \wedge \neg q) \vee (p \wedge q)$

2. Consider the following two statements. **GATE-2015 SET-2**

S1: If a candidate is known to be corrupt, then he will not be elected.

S2: If a candidate is kind, he will be elected.

Which one of the following statements follows from S1 and S2 as per sound inference rules of logic?

(a) If a person is known to be corrupt, he is kind.

(b) If a person is not known to be corrupt, he is not kind

(c) If a person is kind, he is not known to be output

(d) If a person is not kind, he is not known to be corrupt

3. Which one of the following well formed formulae is a tautology? **GATE-2015 SET-2**

(a) $\forall x \exists y R(x, y) \leftrightarrow \exists y \forall x R(x, y)$

(b) $(\forall x[\exists y R(x, y) \rightarrow S(x, y)]) \rightarrow \forall x \exists y S(x, y)$

(c) $[\forall x \exists y (P(x, y) \rightarrow R(x, y)] \leftrightarrow [\forall x \exists y (\neg P(x, y) \vee R(x, y)]$

(d) $\forall x \forall y P(x, y) \rightarrow \forall x \forall y (P(y, x))$

4. In a room there are two types of people, namely Type 1 and Type 2. Type 1 people always till the truth and Type 2 people always lie. You give a fair coin to a person in that room, without knowing which type he is from and tell him to toss it and hide the result from you till you ask for it. Upon asking, the person replies the following **GATE-2015 SET-3**

"The result of the toss is head if and only if I am telling the truth."

Which of the following options is correct?

(a) The result is head

(b) The result is tail

(c) If the person is of Type 2, then the result is tail

(d) If the person is of Type 1, then the result is tail.

……(Sample file here) …… Question Continue ……..

Answer Key & Descriptive Solutions

ANS: 1 c

EXP: P↔stands for Ex-NOR

i.e., (P'q' + pq)

now checking out the options. The correct option is (c)

(p' + q) (p+q')

= p'p + q'q' + pq + qq'

= o + p'q' + pq + o

= p'q' + pq

ANS: 2 c

EXP:

S_1 : If a candidate is known to be corrupt, then he will not be elected.

S_2 : If a candidate is kind, he will be elected.

If p→q, then ~q→~p

So from S_1 , elected → not corrupt; and S_2 is, kind →elected

Therefore, kind →not corrupt

ANS: 3 a

EXP: **……(Sample file here) ……**

ANS: 4 a

EXP: "The result of the Toss is head if any only if I am telling the truth"

If the person is of type 1 who always tell truth, then result must be head.

If the person is of type 2 who always tell lie, then result must be head.

Negation of a sentence of the form "X is true if and only if Y is true" is "Either X is true and Y is false, or X is false and Y is true".

Which means "Either toss is head and I am not telling truth, or toss is tail and I am telling truth"

Since the person always lie, it is "Either toss is head and I am not telling truth".

Chapter : 13 Computer Networks

1. Introduction, Physical, Layer, DLL

1. Suppose that the stop-and-wait protocol is used on a link with a bit rate of 64 kilobits per second and 20 milliseconds propagation delay. Assume that the transmission time for the acknowledgment and the processing time at nodes are negligible. Then the minimum frame size in bytes to achieve a link utilization of at least 50% is _____. **GATE-2015 SET-1**

ANS: 1 Numerical Answer : 320

EXP: Transmission speed = 64kbps

Propagation delay = 20 millisec.

Since stop and wait protocol used, a packet is sent only when the previous packet is acknowledged

Let the size of packet be 'x'

transmission time $= \frac{x}{64} ms$

Since utilization is atleast 50% minimum possible total time for one packet is twice of transmission delay.

Which means $\frac{x}{64} \times 2 = \frac{x}{32}$

$\frac{x}{32} > \frac{x}{64} + 40$

$\frac{x}{64} > 40$

$x > 2560$ bits $= 320$ bytes

2. Consider a LAN with four nodes S_1, S_2, S_3 and S_4. Time is divided into fixed-size slots, and a node can begin its transmission only at the beginning of a slot. A collision is said to have occurred if more than one node transmit in the same slot. The probabilities of generation of a frame in a time slot by S_1, S_2, S_3 and S_4 are 0.1, 0.2, 0.3 and 0.4, respectively. The probability of sending a frame in the first slot without any collision by any of these four stations is _____. **GATE-2015 SET-1**

ANS: 2 Numerical Answer : 0.40 to 0.46

EXP: .....(Sample file here) .....

3. A link has a transmission speed of 10^6 bits/sec. it uses data packets of size 1000 bytes each. Assume that the acknowledgment has negligible transmission delay, and that its propagation delay is the same as the data propagation delay. Also assume that the processing delays at nodes are negligible. The efficiency of the stop-and-wait protocol in this setup is exactly 25%. The value of the one-way propagation delay (in milliseconds) is _____ **GATE-2015 SET-2**

ANS: 3 Numerical Answer : 12

EXP: In stop and wait protocol next packet is sent only when acknowledgment of previous packet is received.

This causes poor link utilization.

Transmission speed = 10^6

Time to send a packet = $\frac{(1000\times 8)\text{ bits}}{10^6}$ = 8 milliseconds

Since, link utilization or efficiency is 25%, total time taken for 1 packet is $\frac{(8\times 100)}{25}$ = 32 milliseconds.

Total time is twice the one way propagation delay plus transmission delay. Propagation delay has to be considered for packet and acknowledgment both.

Transmission delay is considered only for the packet as the question says that transmission delay for acknowledgement is negligible.

Let propagation delay be x

2x+8 = 32

x = 12

4. Consider a CSMA/CD network that transmits data at a rate of 100 Mbps (10^8 bits per second) over a 1 km (Kilometer) cable with no repeaters. If the minimum frame size required for this network is 1250 bytes, what is the signal speed (km/sec) in the cable ? **GATE-2015 SET-3**

(a) 8000 (b) 10000 (c) 16000 (d) 20000

ANS: 4 d

EXP: Data should be transmitted at the rate of 100 Mbps

Transmission time = 2 * propagation time

$$\Rightarrow \quad \frac{(1250*8)}{100\times 100^6} = \frac{2*length}{signal\ speed}$$

$$\Rightarrow \quad signal\ speed = \frac{2\times 10\times 100\times 10^6}{1250\times 8}$$

$$= 2\times 10\times 10^6\, m/\sec$$

$$= 2\times 10\times 10^3\, km/\sec$$

$$= 20000 km/\sec$$

5. Consider a network connecting two systems located 8000 kilometers apart. The bandwidth of the network is 500×10^6 bit per second. The propagation speed of the media is 4×10^6 meters per second. It is needed to design a Go-Back-N sliding window protocol for this network. The average packet size is 10^7 bits. The network is to be used to its full capacity. Assume that processing delays at nodes are negligible. Then, the minimum size in bits of the sequence number field has to be _____.

GATE-2015 SET-3

ANS: 3 Numerical Answer : 8

EXP:

Propagation time $= \frac{8000\times 1000}{4\times 10^6} = 2\text{sec}$

Total round trip propagation time = 4 sec.

Transmission time for one packet $= \frac{Packet\ size}{bandwidth}$

$$= \frac{10^{7}}{500 \times 10^{6}} = 0.02 \text{sec.}$$

Total no. of packets that can be transferred before an acknowledgement comes back = $\frac{400}{0.02} = 200$

Maximum possible window size is 200

In GoBack-N protocol, max. sequence no. should be one more than window size.

So total 201 sequence numbers are needed 201 different sequence no. can be represented using 8 bits.

6. Two hosts are connected via a packet switch with 10^7 bits per second links. Each link has a propagation delay of 20 microseconds. The switch begins forwarded a packet 35 microseconds after it receives the same. If 10000 bits of data are to be transmitted between the two hosts using a packet size of 5000 bits, the time elapsed between the transmission of the first bit of data and the reception of the last bit of the data in microseconds is _____. **GATE-2015 SET-3**

ANS: 3 Numerical Answer : 1575

EXP: Sender host transmits first packet to switch, the transmission time is $\frac{500}{10^7}$ which is 500ms.

After 500ms, the second packet is transmitted.

The first packed reaches destination in 500 + 35 + 20 + 20 +500 = 1075 ms.

While the first packet is travelling to destination, the second packet starts its journey after 500ms. and rest of the time taken by second packet overlaps with first packet. So overall time is

1075 + 500 = 1575

Watch Our All India Topper Video