KEC'S

A COMPLETE TU SOLUTION & PRACTICE SETS

B.Sc. CSIT (V Semester)

All Subjects

Features

- New Syllabus
- TU Questions-Answers
- Model Questions Sets

FOR 2079

KEC'S

A Complete TU Solution

&

Practice SETS

B.Sc.CSIT (V Semester)

All Subjects

Bima Loka Nath Lamsal

FOR 2079

Features

- New Syllabus
- TU Questions-Answers
- Model Questions Sets

KEC

Publication and Distribution Pvt. Ltd.

Kathmandu, Nepal

Phone: 01-4168301

A Complete TU Solution & Practice Sets

B.Sc.CSIT (V Semester)

Edition: 2079

Publisher: KEC Publication and Distribution Pvt. Ltd.
Phone: 01-4168301

Distributor: KEC Publication and Distribution Pvt. Ltd.
Phone: 01-4168301, 01-4241777

email: kecpublication14@gmail.com
Price: Rs. 295/-

CONTENTS

Design and Analysis of Algorithms

Course Title: Design and Analysis of Algorithms **Full Marks:** 60+ 20+20
Course No: CSC 314 **Pass Marks:** 24+8+8
Nature of the Course: Theory + Lab **Credit Hrs:** 3

Course Description: This course introduces basic elements of the design and analysis of computer algorithms. Topics include asymptotic notations and analysis, divide and conquer strategy, greedy methods, dynamic programming, basic graph algorithms, NP-completeness, and approximation algorithms. For each topic, beside in-depth coverage, one or more representative problems and their algorithms shall be discussed.

Course Objectives

- Analyze the asymptotic performance of algorithms.
- Demonstrate a familiarity with major algorithm design techniques
- Apply important algorithmic design paradigms and methods of analysis.
- Solve simple to moderately difficult algorithmic problems arising in applications.
- Able to demonstrate the hardness of simple NP-complete problems

Course Contents

Unit 1: Foundation of Algorithm Analysis **(4)**

1.1. Algorithm and its properties, RAM model, Time and Space Complexity, detailed analysis of algorithms (Like factorial algorithm), Concept of Aggregate Analysis
1.2. Asymptotic Notations: Big-O, Big-Ω and Big-Θ Notations their Geometrical Interpretation and Examples.
1.3. Recurrences: Recursive Algorithms and Recurrence Relations, Solving Recurrences (Recursion Tree Method, Substitution Method, Application of Masters Theorem)

Unit 2: Iterative Algorithms **(4)**

2.1. Basic Algorithms: Algorithm for GCD, Fibonacci Number and analysis of their time and space complexity
2.2. Searching Algorithms: Sequential Search and its analysis
2.3. Sorting Algorithms: Bubble, Selection, and Insertion Sort and their Analysis

Unit 3: Divide and Conquer Algorithms **(8)**

3.1. Searching Algorithms: Binary Search, Min-Max Finding and their Analysis
3.2. Sorting Algorithms: Merge Sort and Analysis, Quick Sort and Analysis (Best Case, Worst Case and Average Case), Heap Sort (Heapify, Build Heap and Heap Sort Algorithms and their Analysis), Randomized Quick sort and its Analysis
3.3. Order Statistics: Selection in Expected Linear Time, Selection in Worst Case Linear Time and their Analysis.

Unit 4: Greedy Algorithms **(6)**

4.1. Optimization Problems and Optimal Solution, Introduction of Greedy Algorithms, Elements of Greedy Strategy.
4.2. Greedy Algorithms: Fractional Knapsack, Job sequencing with Deadlines, Kruskal's Algorithm, Prims Algorithm, Dijkstra's Algorithm and their Analysis
4.3. Huffman Coding: Purpose of Huffman Coding, Prefix Codes, Huffman Coding Algorithm and its Analysis

Unit 5: Dynamic Programming (8)

5.1. Greedy Algorithms vs Dynamic Programming, Recursion vs Dynamic Programming, Elements of DP Strategy

5.2. DP Algorithms: Matrix Chain Multiplication, String Editing, Zero-One Knapsack Problem, Floyd Warshwall Algorithm, Travelling Salesman Problem and their Analysis.

5.3. Memoization Strategy, Dynamic Programming vs Memoization

Unit 6: Backtracking (5)

6.1. Concept of Backtracking, Recursion vs Backtracking

6.2. Backtracking Algorithms: Subset-sum Problem, Zero-one Knapsack Problem, N-queen Problem and their Analysis.

Unit 7: Number Theoretic Algorithms (5)

7.1. Number Theoretic Notations, Euclid's and Extended Euclid's Algorithms and their Analysis.

7.2. Solving Modular Linear Equations, Chinese Remainder Theorem, Primility Testing: Miller-Rabin Randomized Primility Test and their Analysis

Unit 8: NP Completeness (5)

8.1. Tractable and Intractable Problems, Concept of Polynomial Time and Super Polynomial Time Complexity

8.2. Complexity Classes: P, NP, NP-Hard and NP-Complete

8.3. NP Complete Problems, NP Completeness and Reducibility, Cooks Theorem, Proofs of NP Completeness (CNF-SAT, Vertex Cover and Subset Sum)

8.4. Approximation Algorithms: Concept, Vertex Cover Problem, Subset Sum Problem

Laboratory Work

This course can be learnt in effective way only if we give focus is given in practical aspects of algorithms and techniques discussed in class. Therefore student should be able to implement the algorithms and analyze their behavior. Students should:

- Implement comparison sorting algorithms and perform their empirical analysis.
- Implement divide-and-conquer sorting algorithms and perform their empirical analysis.
- Implement algorithms for order statistics and perform their empirical analysis.
- Implement algorithms by using Greedy, DP and backtracking paradigm
- Implement NP-complete problems and realize their hardness.

Recommended Books

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to algorithms", Third Edition.. The MIT Press, 2009.
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekiaran, "Computer Algorithms", Second Edition, Silicon Press, 2007.
3. Kleinberg, Jon, and Eva Tardos, "Algorithm Design", Addison-Wesley, First Edition, 2005

TRIBHUVAN UNIVERSITY

Institution of Science and Technology

Design and Analysis of Algorithms

TU QUESTIONS-ANSWERS 2076

Course Title: Design and Analysis of Algorithms — Full Marks: 60
Course No: CSC 314 — Pass Marks: 24
Nature of the Course: Theory + Lab — Time: 3 hrs.
Semester: V

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. What do you mean by complexity of an algorithm? Explain about the asymptotoc notations used to describe the time/space complexity of any algorithm with their gemoetrical interpretation and example. **(1+6)**

Ans: As we know that a by Designing an Algorithm, a problem can be solved. To perform some tasks based on what the problem is asking. There could be many solutions for a single problem, and each solution results in a unique or similar algorithm. So it becomes very difficult to decide which algorithm to choose from the set of algorithms so that we can get a solution in few steps or finite time. Here, the complexities of an algorithm come into the picture.

The complexity of an algorithm defines the performance of the algorithm in terms of the input size. We consider the complexities of every algorithm and compare them while choosing the most efficient algorithm to solve our problem.

There are 2 types of complexity to consider for an algorithm

- Time Complexity
- Space Complexity

Time complexity is the time taken by the algorithm to execute each set of instructions. It is always better to select the most efficient algorithm when a simple problem can solve with different methods.

Space complexity is usually referred to as the amount of memory consumed by the algorithm. It is composed of two different spaces; Auxiliary space and Input space.

Asymptotoc notations used to describe the time/space complexity of any algorithm

Whenever we want to perform analysis of an algorithm, we need to calculate the complexity of that algorithm. But when we calculate complexity of an algorithm it does not provide exact amount of resource required. So instead of taking exact amount of resource we represent that complexity in a general form which produces the basic nature of that algorithm. We use that general form for analysis process.

Complexity analysis of an algorithm is very hard if we try to analyze exact. We know that the complexity (worst, best, or average) of an algorithm is the mathematical function of the size of the input. So if we analyze the algorithm in terms of bound (upper and lower) then it would be easier. For this purpose, we need the concept of asymptotic notations. Asymptotic Notation is a way of comparing function that ignores constant factors and small input sizes. Three notations are used to calculate the running time complexity of an algorithm.

Big Oh (O) notation

When we have only asymptotic upper bound then we use O notation. If f and g are any two functions from set of integers to set of integers then function f(x) is said to be big oh of g(x) i.e. f(x)=O(g(x))) if and only if there exists two positive constants c and x_0 such that for all x >= x_0, f(x) <= c*g(x)

The above relation says that g(x) is an upper bound of f(x)

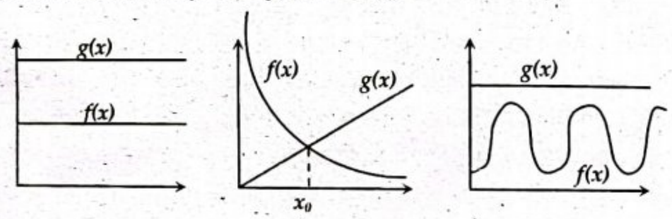


Fig: Geometric intrepretation of Big-Oh notation

Example: Find big oh of given function $f(n) = 3n^2 + 4n + 7$

Solution: we have $f(n) = 3n^2 + 4n + 7 <= 3n^2 + 4n^2 + 7n^2 <= 14n^2$

$f(n) <= 14n^2$

where, c=14 and $g(n) = n^2$, thus $f(n) = O(g(n)) = O(n^2)$

Big Omega (Ω) notation

Big omega notation gives asymptotic lower bound. If f and g are any two functions from set of integers to set of integers, then function f(x) is said to be big omega of g(x) i.e. f(x)=Ω(g(x)) if and only if there exists two positive constants c and x_0 such that

For all x >= x_0, f(x)>=c*g(x)

The above relation says that g(x) is a lower bound of f(x).

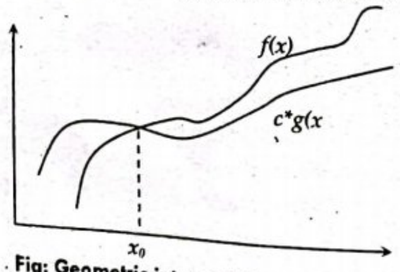


Fig: Geometric intrepretation of Big Omega notation

Example: Find big omega of $f(n) = 3n^2 + 4n + 7$

Solution: Since we have $f(n) = 3n^2 + 4n + 7 >= 3n^2$

⇨ $f(n) >= 3n^2$

where, c=3 and $g(n) = n^2$, thus $f(n) = \Omega(g(n)) = \Omega(n^2)$

Big Theta (Θ) notation

When we need asymptotically tight bound then we use this notation. If f and g are any two functions from set of integers to set of integers then function f(x) is said to be big theta of g(x) i.e. $f(x) = \Theta(g(x))$ if and only if there exists three positive constants c_1, c_2 and x_0 such that for all

x >= x_0, c_1*g(x) <= f(x) <= c_2*g(x)

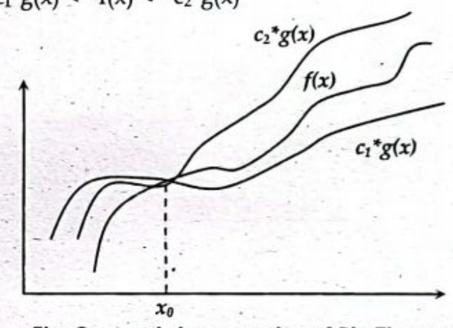


Fig: Geometric intrepretation of Big Theta notation

Example: If $f(n) = 3n^2 + 4n + 7$ $g(n) = n^2$, then prove that $f(n) = \Theta(g(n))$.

Proof: let us choose c_1, c_2 and n_0 values as 14, 1 and 1 respectively then we can have,

$f(n) <= c_1*g(n)$, n>=n_0 as $3n^2 + 4n + 7 <= 14*n^2$, and

$f(n) >= c_2*g(n)$, n>=n_0 as $3n^2 + 4n + 7 >= 1*n^2$

For all n >= 1(in both cases).

So $c_2*g(n) <= f(n) <= c_1*g(n)$ is trivial.

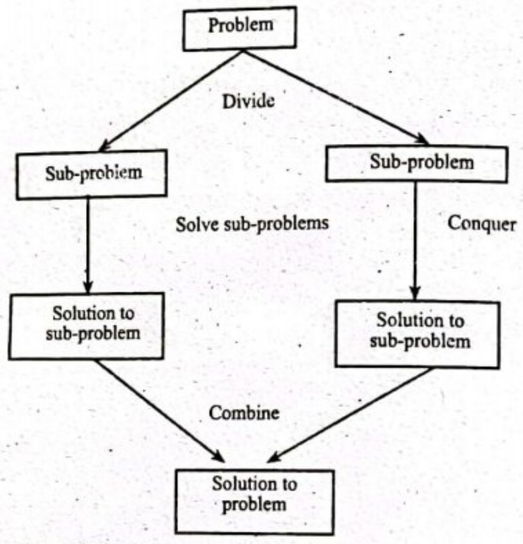
Hence $f(n) = \Theta(g(n))$.

2. Explain about the divide and conquer paradigm form algorithm design with suitable example. Write the Quick sort algorithm using randomized approch and explain its time complexity. (4+6)

Ans: Divide and Conquer is an algorithmic pattern. In algorithmic methods, the design is to take a dispute on a huge input, break the input into minor pieces, decide the problem on each of the small pieces, and then merge the piecewise solutions into a global solution. This mechanism of solving the problem is called the Divide & Conquer Strategy. Divide and Conquer algorithm consists of a dispute using the following three steps.

- **Divide** the original problem into a set of sub-problems.
- **Conquer:** Solve every sub-problem individually, recursively.

- **Combine:** Put together the solutions of the sub-problems to get the solution to the whole problem.



Randomized Quick Sort

The algorithm is called randomized if its behavior depends on input as well as random value generated by random number generator. The beauty of the randomized algorithm is that no particular input can produce worst-case behavior of an algorithm. IDEA: Partition around a random element. Running time is independent of the input order. No assumptions need to be made about the input distribution. No specific input elicits the worst-case behavior. The worst case is determined only by the output of a random-number generator. Randomization cannot eliminate the worst-case but it can make it less likely!

Algorithm:

```
RandQuickSort(A,l,r)
{
    if(l<r)
    {
    m = RandPartition (A, l, r);
    RandQuickSort (A, l, m-1);
    RandQuickSort (A, m+1, r);
    }
}
RandPartition (A, l, r)
{
    k = random (l, r); //generates random number between i and j including both.
    swap(A[l],A[k]);
    return Partition(A, l, r);
}
Partition (A, l, r)
{
```



```
    x =l; y =r; p = A[l];
    while(x<y)
    {
    do {
    x++;
      }while(A[x] <= p);
    do {
    y--;
      } while(A[y] >=p);
    if(x<y)
    swap(A[x],A[y]);
    }
    A[l] = A[y]; A[y] = p;
    return y;  //return position of pivot
}
```

Time Complexity

Worst Case

T(n) = worst-case running time

Let k be the partition element then there are two sub problems of size k and (n-k), Since there are n elements so we need at most O(n) time for dividing.

Thus their recurrence relation can be defined as,

$$T(n) = \max_{1 \le k \le n-1} (T(k) + T(n-k)) + O(n) \quad \ldots\ldots(1)$$

Where, k is some partitioned point produced by random number generator

Now, by using substitution method to show that the running time of Quick sort is $O(n^2)$

Guess $T(n) = O(n^2)$

⇨

$$T(n) \le cn^2 \quad \ldots\ldots(2)$$

Now proof this by using mathematical induction

Basic step: for n=1,

$T(1) \le c.\ 1^2$

Or $1 \le c$ which is true for $c > 0$

Inductive step:

Let's assume that it is true for all $k < n$

I.e. $T(k) \le ck^2$ for any $k < n$

It is also true for k=n-k,

I.e. $T(n-k) \le c\,(n-k)^2$

Now equation 1 becomes,

$T(n) \le \max_{1 \le k \le n-1} (ck^2 + c(n-k)^2) + O(n)$

$= c \cdot \max_{1 \le k \le n-1} (k^2 + (n-k)^2) + O(n)$

The expression $k^2 + (n-k)^2$ achieves a maximum over the range $1 \le k \le n-1$ at one of the endpoints

$\max_{1 \le k \le n-1} (k^2 + (n-k)^2) = 1^2 + (n-1)^2 = n^2 - 2(n-1)$

$T(n) \le cn^2 - 2c(n-1) + O(n)$

$\le cn^2$

⇨ $T(n) = O(n^2)$

3. Explain in brief about the Backtracking approach for algorithm design. How it differs with recursion? Explain the N-Queen Problem and its algorithm using backtracking and analyze its time complexity. (2+2+6)

Ans: The Backtracking is an algorithmic-method to solve a problem with an additional way. The Backtracking is an algorithmic-technique to solve a problem by an incremental way. It uses recursive approach to solve the problems. We can say that the backtracking is used to find all possible combination to solve an optimization problem.

Have you ever seen poor blind people walking in roads? If they find any obstacles in their way, they would just move backward. Then they will proceed in other direction. How a blind person could move backward when he finds obstacles? Simple answer by intelligence! Similarly, if an algorithm backtracks with intelligence, it is called backtracking algorithm.

Backtracking is general algorithm for finding solution to some computational problem. We have set of several choices. If one choice from set of choices proves incorrect, computation backtracks or restarts at the point of choice and tries another choice. In backtracking you use recursion in order to explore all the possibilities until you get the best result for the problem.

N-queen Problem

This problem is to find an arrangement of N queens on a chess board, such that no queen can attack any other queens on the board. The chess queens can attack in any direction as horizontal, vertical, horizontal and diagonal way. A binary matrix is used to display the positions of N Queens, where no queens can attack other queens.

So initially we are having n × n un-attacked cells where we need to place n queens. Let's place the first queen at a cell (i, j), so now the number of un-attacked cells is reduced, and number of queens to be placed is n−1. Place the next queen at some un-attacked cell. This again reduces the number of un-attacked cells and number of queens to be placed becomes n−2. Continue doing this, as long as following conditions hold.

- The number of un-attacked cells is not 0.
- The number of queens to be placed is not 0.

If the number of queens to be placed becomes 0, then it's over, we found a solution. But if the number of un-attacked cells become 0, then we need to backtrack, i.e. remove the last placed queen from its current cell, and place it at some other cell. We do this recursively.

Algorithm

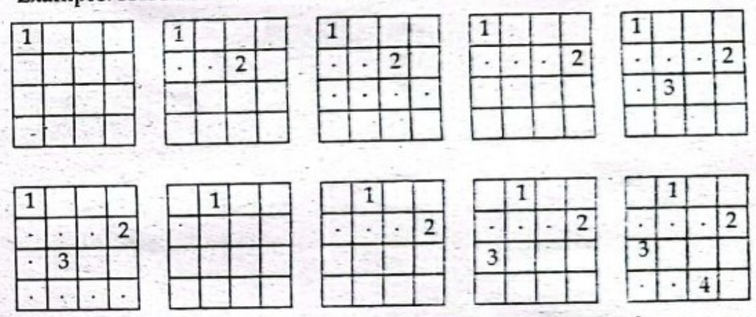
1. Start
2. Place the queens column wise, start from the left most column
3. If all queens are placed.
   a. Return true and print the solution matrix.
4. Else
   a. Try all the rows in the current column.
   b. Check if queen can be placed here safely if yes mark the current cell in solution matrix as 1 and try to solve the rest of the problem recursively.
   c. If placing the queen in above step leads to the solution return true.

   d. If placing the queen in above step does not lead to the solution, BACKTRACK, mark the current cell in solution matrix as 0 and return false.
5. If all the rows are tried and nothing worked, return false and print NO SOLUTION.
6. Stop

Analysis

Solution of N Queen problem using backtracking checks for all possible arrangements of N Queens on the chessboard. And then checks for the validity of the solution. Now number of possible arrangements of N Queens on N x N chessboard is N!. So average and worst case complexity of the solution is O(N!). The best case occurs if we find our solution before exploiting all possible arrangements. This depends on our implementation. And if we need all the possible solutions, the best, average and worst case complexity remains O(N!)

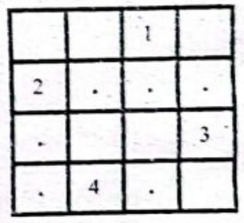
Example: Here's how it works for N=4.



Which is one of the solutions of 4-queen problem of sequence is (2, 4, 1, 3)

Also by mirroring of this solution we get another possible solution as below,

The sequence is (3, 1, 4, 2)



Section B

Attempt any eight questions. (8 × 5 = 40)

4. Write the algorithm for Selection Sort and explain its time and space complexity. (5)

Ans: In selection sort, the smallest value among the unsorted elements of the array is selected in every pass and inserted to its appropriate position into the array. First, find the smallest element of the array and place it on the first position. Then, find the second smallest element of the array and place it on the second position. The process continues until we get the sorted array. The array with n elements is sorted by using n-1 pass of selection sort algorithm.

1. In 1st pass, smallest element of the array is to be found along with its index pos. then, swap A[0] and A[pos]. Thus A[0] is sorted, we now have n -1 elements which are to be sorted.
2. In 2nd pas, position pos of the smallest element present in the sub-array A[n-1] is found. Then, swap, A[1] and A[pos]. Thus A[0] and A[1] are sorted, we now left with n-2 unsorted elements.
3. In n-1th pass, position pos of the smaller element between A[n-1] and A[n-2] is to be found. Then, swap, A[pos] and A[n-1].....and so on.

Tracing: Sort the following data items by using Selection sort

A[ ]={25, 57, 48, 37, 12, 92, 86, 33}

Solution:

Array position	0	1	2	3	4	5	6	7
Initial state	25	57	48	37	12	92	86	33
Pass 1	12	57	48	37	25	92	86	33
Pass 2	12	25	48	37	57	92	86	33
Pass 3	12	25	33	37	57	92	86	48
Pass 4	12	25	33	37	57	92	86	48
Pass 5	12	25	33	37	48	92	86	57
Pass 6	12	25	33	37	48	57	86	92
Pass 7	12	25	33	37	48	57	86	92
Pass 8	12	25	33	37	48	57	86	92

Algorithm

1. Start
2. Consider the first element to be sorted and the rest to be unsorted
3. Assume the first element to be the smallest element.
4. Check if the first element is smaller than each of the other elements:
   If yes, do nothing
   If no, choose the other smaller element as minimum and repeat step 3
5. After completion of one iteration through the list, swap the smallest element with the first element of the list.
6. Now consider the second element in the list to be the smallest and so on till all the elements in the list are covered.
7. Stop

Time Complexity

Inner loop executes for (n-1) times when i=0, (n-2) times when i=1 and so on:
Time complexity = (n-1) + (n-2) + (n-3) + ... +2 +1
= $O(n^2)$

There is no best-case linear time complexity for this algorithm, but number of swap operations is reduced greatly.

Space Complexity of Selection Sort

The space complexity of Selection Sort is O(1).
This is because we use only constant extra space such as:

- 2 variables to enable swapping of elements.
- One variable to keep track of smallest element in unsorted array.

Hence, in terms of Space Complexity, Selection Sort is optimal as the memory requirements remain same for every input.

5. **Solve the following recurrence relations using master method.** (2.5+2.5)

a. $T(n) = 7\,T(n/2) + n^2$

Ans: Here we have a=7, b=2 and $f(n) = n^2$

Now, $n^{\log_b a} = n^{\log_2 7} = n^{(\log 7/\log 2)} = n^{2.80}$

Also $f(n) = n^2$

Since $f(n) \le n^{\log_b a - \varepsilon}$ where choose $\varepsilon = 0.1$

Thus it satisfy the first case of Master's method

Thus it's complexity,

$T(n) = \Theta(n^{\log_b a}) = \Theta(n^{\log_2 7}) = \Theta(n^{2.80})$

Thus $T(n) = \Theta(n^{2.8})$

b. $T(n) = 4\,T(n/4) + kn$

Ans: Here we have a=4, b=4 and $f(n) = kn$

Now, $n^{\log_b a} = n^{\log_4 4} = n^{(\log 4/\log 4)} = n^1$

Also $f(n) = kn^1$

Since $f(n) = n^{\log_b a}$

Thus it satisfy the third case of Master's method

Thus it's complexity,

$T(n) = \Theta(f(n) \log n) = \Theta(kn \log n)$

Thus $T(n) = \Theta(n \log n)$

6. **Explain the greedy algorithm for fractional knapsack problem with its time complexity.** (5)

Ans: **Statement:** A thief has a bag or knapsack that can contain maximum weight W of his loot. There are n items and the weight of i^{th} item is wi and it worth vi. Any amount of item can be put into the bag i.e. xi fraction of item can be collected, where $0 \le xi \le 1$. Here the objective is to collect the items that maximize the total profit earned. Here we arrange the items by ratio vi/wi.

Algorithm

```
GreedyFracKnapsack (W, n)
{
for(i=1; i<=n; i++)
x[i] = 0.0;
tempW = W;
for(i=1; i<=n; i++)
{
if(w[i] > tempW) then
break;
x[i] = 1.0;
tempW -= w[i];
}
if(i<=n)
x[i] = tempW/w[i];
}
```

Analysis

We can see that the above algorithm just contain a single loop i.e. no nested loops the running time for above algorithm is O(n). However our requirement is that v[1 ... n] and w[1 ... n] are sorted, so we can use sorting method to sort it in O(n log n) time such that the complexity of the algorithm above including sorting becomes O(n log n).

Example: Consider five items along with their respective weights and values,

I = {I1, I2, I3, I4, I5}

w = {5, 10, 20, 30, 40}

v = {30, 20, 100, 90, 160}

The knapsack has capacity W=60, then find optimal profit earned by using fractional knapsack.

Solution: Initially

Items	wi	vi
I1	5	30
I2	10	20
I3	20	100
I4	30	90
I5	40	160

Step 2: calculate vi/wi as,

Items	Wi	vi	Pi=vi/wi
I1	5	30	6.0
I2	10	20	2.0
I3	20	100	5.0
I4	30	90	3.0
I5	40	160	4.0

Step 3: Arranging the items with decreasing order of Pi as,

Items	Wi	vi	Pi=vi/wi
I1	5	30	6.0
I3	20	100	5.0
I5	40	160	4.0
I4	30	90	3.0
I2	10	20	2.0

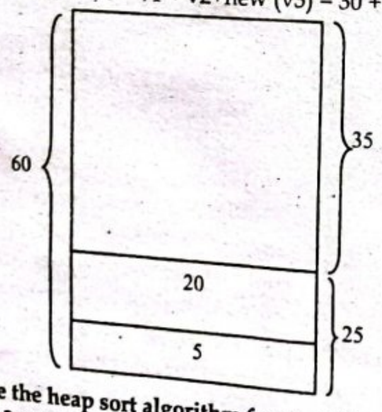
Now filling the knapsack according to decreasing value of Pi

Since, 40 w = 160 vi

1 w = 160/40 = 4vi

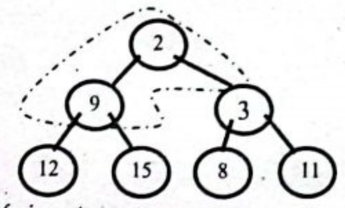
35 w = 35 * 4 = 140 vi

Thus, maximum value = v1 + v2+new (v3) = 30 + 100 + 140 = 270

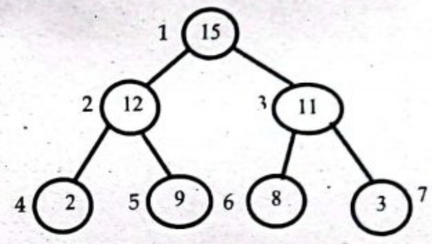


7. Trace the heap sort algorithm for the following data: {2, 9, 3, 12, 15, 8, 11} (5)

Ans: At first construct a binary tree of given array.

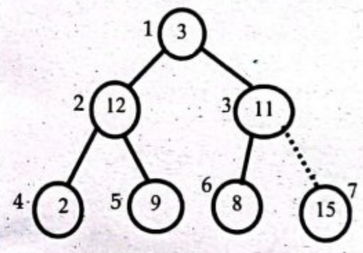


Now construct a heap of given tree as,

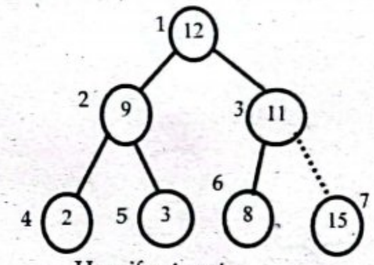


Heap sort

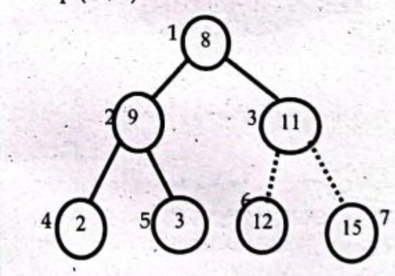
Swap (15, 3)



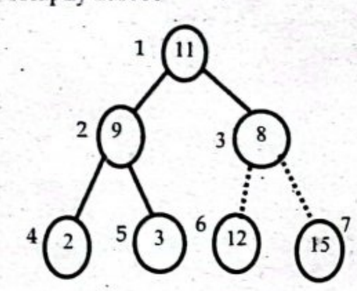
Heapify at root



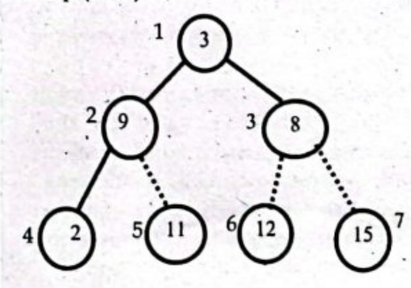
Swap (12, 8)



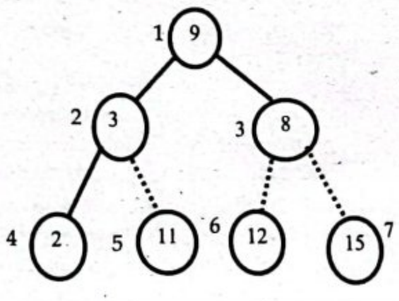
Heapify at root

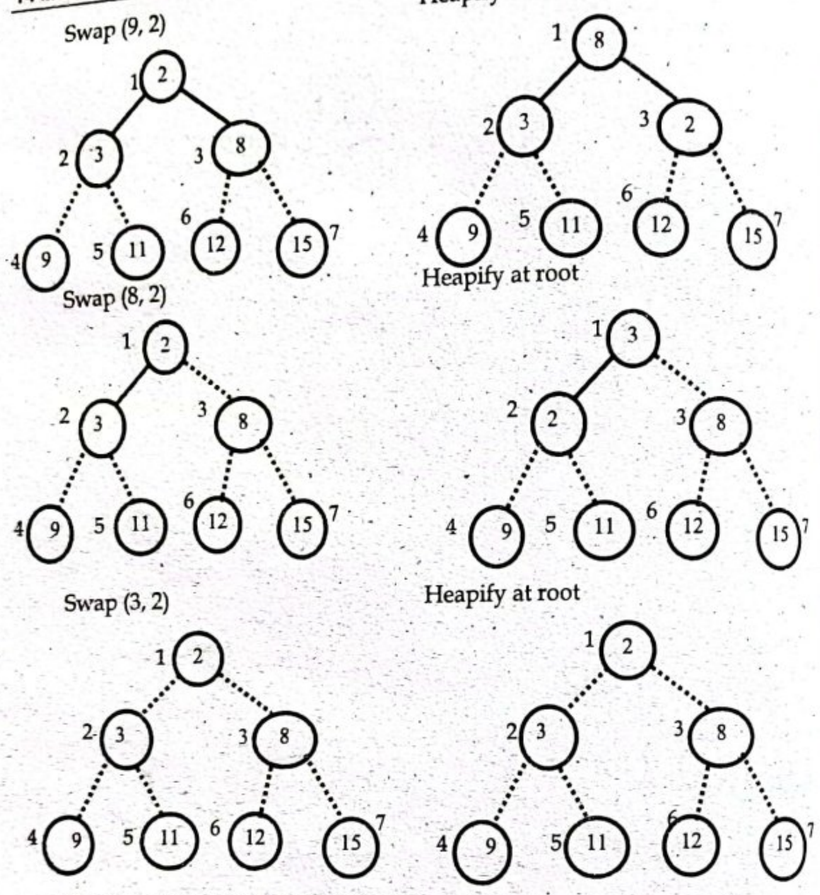


Swap (11, 3)



Heapify at root





The sorted list = {2, 3, 8, 9, 11, 12, 15}

8. What do you mean by Dynamic Programming strategy? Explain the element of DP. (2+3)

Ans: Dynamic Programming is the most powerful design technique for solving optimization problems. Divide & Conquer algorithm partition the problem into disjoint sub-problems solves the sub-problems recursively and then combine their solution to solve the original problems. Dynamic Programming is used when the sub-problems are not independent, e.g. when they share the same sub-problems. In this case, divide and conquer may do more work than necessary, because it solves the same sub problem multiple times.

Dynamic Programming solves each sub-problem just once and stores the result in a table so that it can be repeatedly retrieved if needed again. Dynamic Programming is a Bottom-up approach- we solve all possible small problems and then combine to obtain solutions for bigger problems. Dynamic Programming is a paradigm of algorithm design in which an optimization problem is solved by a combination of achieving sub-problem solutions and appearing to the "principle of optimality".

Characteristics of Dynamic Programming

Dynamic Programming works when a problem has the following features:-

- **Optimal Substructure:** If an optimal solution contains optimal sub solutions then a problem exhibits optimal substructure.
- **Overlapping sub-problems:** When a recursive algorithm would visit the same sub-problems repeatedly, then a problem has overlapping sub-problems.

If a problem has optimal substructure, then we can recursively define an optimal solution. If a problem has overlapping sub-problems, then we can improve on a recursive implementation by computing each sub-problem only once

Elements of DP Strategy

There are basically three elements that characterize a dynamic programming algorithm:

- **Substructure:** Decompose the given problem into smaller sub-problems. Express the solution of the original problem in terms of the solution for smaller problems.
- **Table Structure:** After solving the sub-problems, store the results to the sub problems in a table. This is done because sub-problem solutions are reused many times, and we do not want to repeatedly solve the same problem over and over again.
- **Bottom-up Computation:** Using table, combine the solution of smaller sub-problems to solve larger sub-problems and eventually arrives at a solution to complete problem.

9. Explain the approximation algorithm for solving vertex cover with suitable example. (5)

Ans : An Approximate Algorithm is a way of approach NP-Completeness for the optimization problem. This technique does not guarantee the best solution. The goal of an approximation algorithm is to come as close as possible to the optimum value in a reasonable amount of time which is at the most polynomial time. Such algorithms are called approximation algorithm or heuristic algorithm.

- For the traveling salesperson problem, the optimization problem is to find the shortest cycle, and the approximation problem is to find a short cycle.
- For the vertex cover problem, the optimization problem is to find the vertex cover with fewest vertices, and the approximation problem is to find the vertex cover with few vertices.

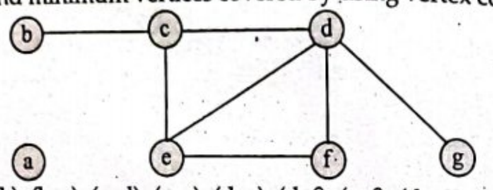
Vertex Cover Problem

An approximate algorithm is a way of dealing with NP-completeness for optimization problem. This technique does not guarantee the best solution. The goal of an approximation algorithm is to come as close as possible to the optimum value in a reasonable amount of time which is at most polynomial time.

A Vertex Cover of a graph G is a set of vertices such that each edge in G is incident to at least one of these vertices. The decision vertex-cover problem was proven NPC. Now, we want to solve the optimal version of the vertex cover problem, i.e., we want to find a minimum size vertex cover of a given graph. We call such vertex cover an optimal vertex cover C*.The idea is to

take an edge (u, v) one by one, put both vertices to C, and remove all the edges incident to u or v. We carry on until all edges have been removed. C is a VC.

Example: Find minimum vertices covered by using vertex cover problem.

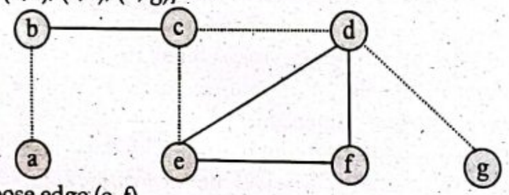


Solution: E = {(a, b), (b, c), (c, d), (c, e), (d, e), (d, f), (e, f), (d, g)}

Step 1: Lets choose edge C = (b, c)

Eliminate edges incident to vertex b and c

E = {(d, e), (d, f), (e, f), (d, g)}

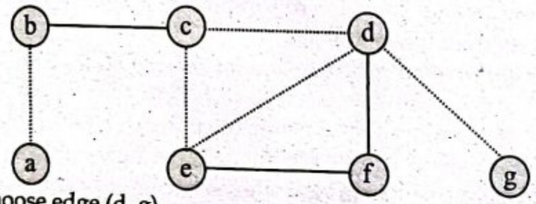


Step 2: Let's choose edge (e, f)

C = {b, c, e, f}

Eliminate edges incident to vertex e and f

E = {(d, g)}

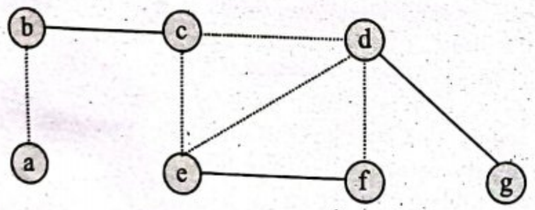


Step 3: Let's choose edge (d, g)

C = {b, c, e, f, d, g}

Eliminate edges incident to vertex d and g

E = {Φ}



Algorithm

Approx-Vertex-Cover (G = (V, E))

{

C = empty-set;

E'= E;

While E' is not empty do

{

Let (u, v) be any edge in E': (*)

Add u and v to C;

Remove from E' all edges incident to u or v;

}

Return C;

}

Analysis

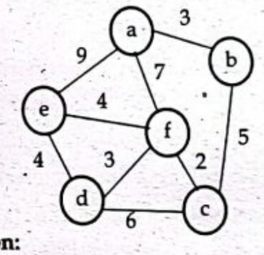
If E is represented using the adjacency lists the above algorithm takes O(V+E) since each edge is processed only once and every vertex is processed only once throughout the whole operation.

10. Explain the Prim's algorithm for MST problem and analyze its time complexity. (5)

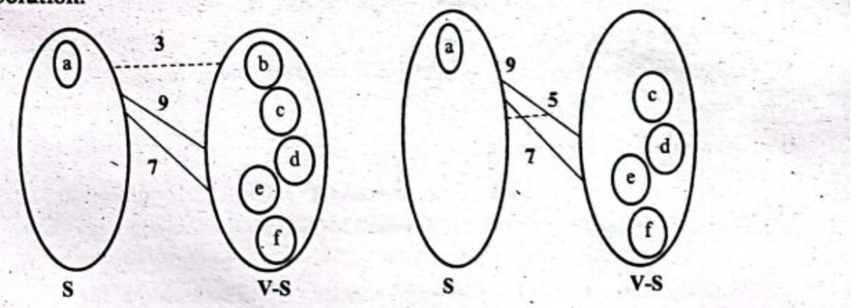
Ans: Prim's Algorithm

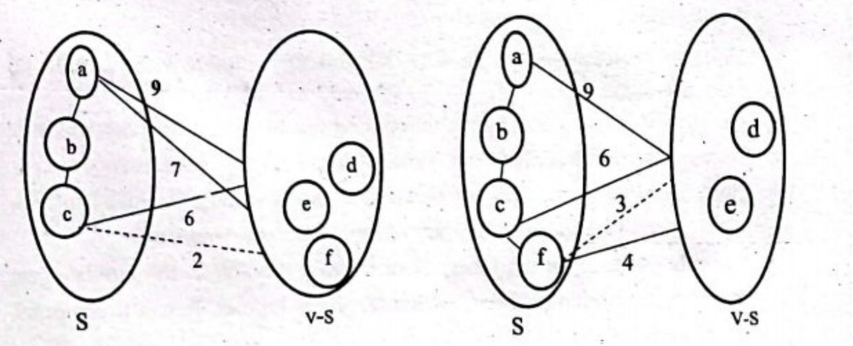
Prim's algorithm is a Greedy algorithm. It starts with an empty spanning tree. The idea is to maintain two sets of vertices. The first set contains the vertices already included in the MST, the other set contains the vertices not yet included. At every step, it considers all the edges that connect the two sets, and picks the minimum weight edge from these edges. After picking the edge, it moves the other endpoint of the edge to the set containing MST. Remember the cycle must be avoided.

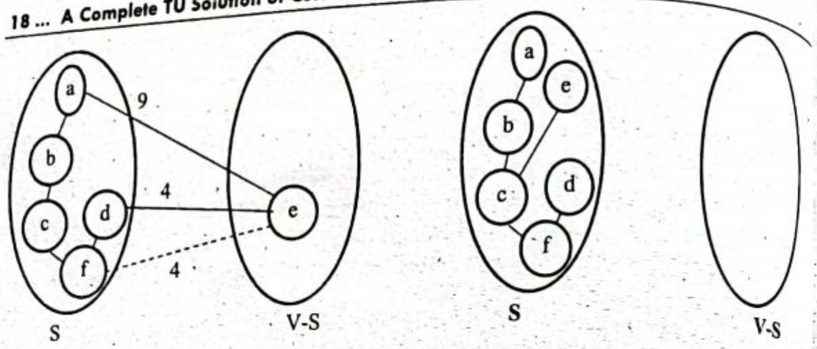
Example: find minimum spanning tree of given graph by using Prim's algorithm



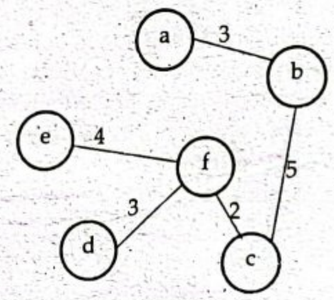
Solution:







Thus final MST given by Prim's algorithm is given below;



Thus the minimum spanning tree of weight 17 is shown in fig above.

Algorithm

1. Start
2. Initialize the minimum spanning tree with a vertex chosen at random.
3. Find all the edges that connect the tree to new vertices, find the minimum and add it to the tree
4. Keep repeating step 3 until we get a minimum spanning tree
5. Stop

Analysis

In the above algorithm while loop execute O(V). The edge of minimum weight incident on a vertex can be found in O(E), so the total time is O(EV). We can improve the performance of the above algorithm by choosing better data structures as priority queue and normally it will be seen that the running time of prim's algorithm is O(E log V)!.

11. Explain in brief about the classes P, NP and NP complete with example. (5)

Ans: Class P Problem

The class P consists of those problems that can be solved by a deterministic Turing machine in Polynomial time. P problems are obviously tractable. More specifically, they are problems that can be solved in time $O(n^k)$ for some constant k, where n is the size of the input to the problem.

Example: Adding two numbers is really easy. Surely, as the number gets larger the computation becomes harder to us human. But to a computer

adding large numbers are fairly simple. We can say computers can add two numbers in Polynomial time. These types of problem which can be solved in polynomial time by a computer are known as **P problems**.

NP-Class

NP is set of decision problems that can be solved by a Non-deterministic Turing Machine in Polynomial time. P is subset of NP (any problem that can be solved by deterministic machine in polynomial time can also be solved by non-deterministic machine in polynomial time).The class NP consists of those problems that are verifiable in polynomial time. NP is the class of decision problems for which it is easy to check the correctness of a claimed answer, with the aid of a little extra information. Hence, we aren't asking for a way to find a solution, but only to verify that an alleged solution really is correct. Every problem in this class can be solved in exponential time using exhaustive search.

Example: Let's take a little complex problem like prime factorization. We know that every composite number can be expressed as a product of two or more prime factors. Our normal PCs can handle this problem within seconds for numbers up to a billion. But as the numbers grow this problem becomes lot harder even for the fastest of computers. So this is not **solvable in Polynomial time.** So factorization is not solvable in polynomial time but the solution is verifiable in polynomial time. These problems are known as NP problems.

NP-Complete

NP-Complete problem is a complexity class which represents the set of all problems X in NP for which it is possible to reduce any other NP problem Y to X in polynomial time. NP-complete problems are the hardest problems in NP set. A decision problem L is NP-complete if:

- L is in NP (Any given solution for NP-complete problems can be verified quickly, but there is no efficient known solution).
- Every problem in NP is reducible to L in polynomial time

Intuitively this means that we can solve Y quickly if we know how to solve X quickly. Precisely, Y is reducible to X, if there is a polynomial time algorithm f to transform instances y of Y to instances x = f(y) of X in polynomial time, with the property that the answer to y is yes, if and only if the answer to f(y) is yes.

Examples: An interesting example is the graph isomorphism problem, the graph theory problem of determining whether a graph isomorphism exists between two graphs. Two graphs are isomorphic if one can be transformed into the other simply by renaming vertices'. Consider these two problems:

- **Graph Isomorphism:** Is graph G1 isomorphic to graph G2?

- **Sub-graph Isomorphism:** Is graph G1 isomorphic to a sub-graph of graph G2?

The Sub-graph Isomorphism problem is NP-complete. The graph isomorphism problem is suspected to be neither in P nor NP-complete, though it is in NP. This is an example of a problem that is thought to be hard, but is not thought to be NP-complete.

The easiest way to prove that some new problem is NP-complete is first to prove that it is in NP, and then to reduce some known NP-complete problem to it. Therefore, it is useful to know a variety of NP-complete problems. The list below contains some well-known problems that are NP-complete when expressed as decision problems.

- Boolean Satisfiability problem (SAT)
- Knapsack problem
- Hamiltonian path problem
- Traveling salesman problem (decision version)
- Sub-graph isomorphism problem
- Subset sum problem

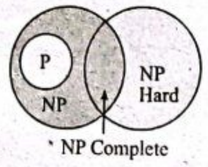


Fig: Relationships between classes P, NP, NP Complete and NP Hard

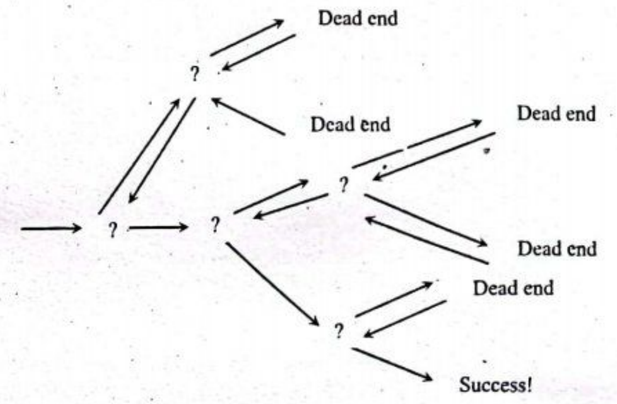
12. Write short notes on (2 x 2.5)

a. Backtracking strategy

Ans: The Backtracking is an algorithmic-method to solve a problem with an additional way. The Backtracking is an algorithmic-technique to solve a problem by an incremental way. It uses recursive approach to solve the problems. We can say that the backtracking is used to find all possible combination to solve an optimization problem.

Have you ever seen poor blind people walking in roads? If they find any obstacles in their way, they would just move backward. Then they will proceed in other direction. How a blind person could move backward when he finds obstacles? Simple answer by intelligence! Similarly, if an algorithm backtracks with intelligence, it is called backtracking algorithm.

Backtracking is general algorithm for finding solution to some computational problem. We have set of several choices. If one choice from set of choices proves incorrect, computation backtracks or restarts at the point of choice and tries another choice. In backtracking you use recursion in order to explore all the possibilities until you get the best result for the problem.



b. Tractable and Intractable problems

Ans: We call problems as **tractable** or easy, if the problem can be solved using polynomial time algorithms. The problems that cannot be solved in polynomial time but requires super-polynomial time algorithm are called **intractable** or hard problems. There are many problems for which no algorithm with running time better than exponential time is known some of them are, travelling salesman problem, Hamiltonian cycles, and circuit satisfiability, etc.

Here are examples of tractable problems (ones with known polynomial-time algorithms):

- Searching an unordered list
- Searching an ordered list
- Sorting a list
- Multiplication of integers (even though there's a gap)
- Finding a minimum spanning tree in a graph (even though there's a gap)

Here are examples of intractable problems (ones that have been proven to have no polynomial-time algorithm). Some of them require a non-polynomial amount of output, so they clearly will take a non-polynomial amount of time e.g.:

- **Towers of Hanoi**: we can prove that any algorithm that solves this problem must have a worst-case running time that is at least $2^n - 1$.
- List all permutations (all possible orderings) of n numbers.

TU QUESTIONS-ANSWERS 2078

Course Title: Design and Analysis of Algorithms
Course No: CSC 314
Nature of the Course: Theory + Lab
Semester: V

Full Marks: 60
Pass Marks: 24
Time: 3 hrs.

Section A

Attempt any two questions. (2 × 10 = 20)

1. What are the elementary properties of algorithm? Explain. Why do you need analysis of algorithm? Discuss about the RAM model for analysis of algorithm with suitable example. (2+2+6)

Ans: Properties

In order for an algorithm to be useful, it must help us find a solution to a specific problem. For that to happen, an algorithm must satisfy five properties.

- **Input**: The inputs used in an algorithm must come from a specified set of elements, where the amount and type of inputs are specified.
- **Output**: The algorithm must specify the output and how it is related to the input.
- **Definiteness**: The steps in the algorithm must be clearly defined and detailed.
- **Effectiveness**: The steps in the algorithm must be doable and effective.
- **Finiteness**: The algorithm must come to an end after a specific number of steps.

When an algorithm satisfies these five properties, it is a fail-proof way to solve the problem for which it was written.

Algorithm analysis is an important part of a broader computational complexity theory, which provides theoretical estimates for the resources needed by any algorithm which solves a given computational problem. These estimates provide an insight into reasonable directions of search for efficient algorithms.

RAM model

Algorithms can be measured in a machine-independent way using the Random Access Machine (RAM) model. This model assumes a single processor. In the RAM model, instructions are executed one after the other, with no concurrent operations. This model of computation is an abstraction that allows us to compare algorithms on the basis of performance. The assumptions made in the RAM model to accomplish this are:

- Each simple operation takes 1 time step.
- Loops and subroutines are not simple operations.
- Each memory access takes one time step, and there is no shortage of memory.

For any given problem the running time of an algorithm is assumed to be the number of time steps. The space used by an algorithm is assumed to be the number of RAM memory cells. In computing time complexity, one good approach is to count primitive operations. This approach of simply counting primitive operations gives rise to a computational model called the Random Access Machine (RAM). The RAM mode consists of following elements.

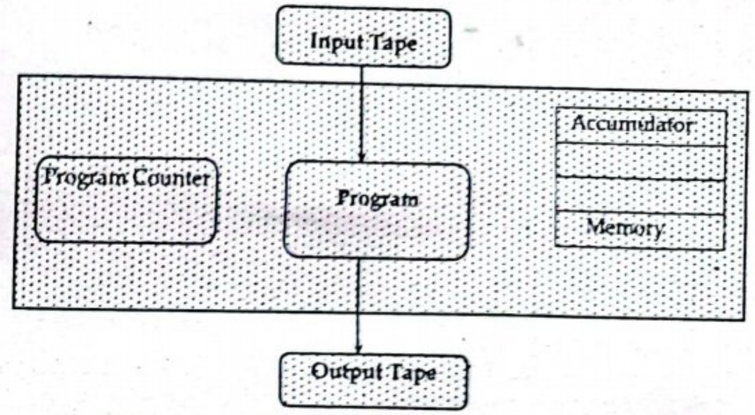


Fig: RAM Model

- **Input tape/output tape:** input tape consists of a sequence of squares, each of which can store integer. Whenever one square is read from the tape head moves one square to the right. The output tape is also a sequence of squares, in each square an integer can be written. Output is written on the square under the tape head and the after the writing, the tape head moves one square to the right over writing on the same square is not permitted.
- **Memory:** The memory consists of a sequence of registers, each of which is capable of holding an integer.

Program: Program for RAM contains a sequence of labeled instructions resembling those found in assembly language programs. All computations take place in the first register called accumulator. A RAM program defines a mapping from input tape to the output tape.

Example: Find detailed analysis of following factorial algorithm

```
#include <stdio.h>
int main()
{
    int i, n, fact = 1;
    printf("Enter a number to calculate its factorial\n");
    scanf("%d", &n);
    for (i = 1; i <= n; i++)
    fact = fact * i;
    printf("Factorial of %d = %d\n", n, fact);
    return 0;
}
```

Time complexity

The declaration statement takes 1 step
Printf statement takes 1 step time
Scanf statement takes 1 step
In for loop,
    i=1 takes 1 step
    i<=n takes (n+1) step
    i++ takes n step
    within for loop fact = fact*i takes n step
printf statement takes 1 step

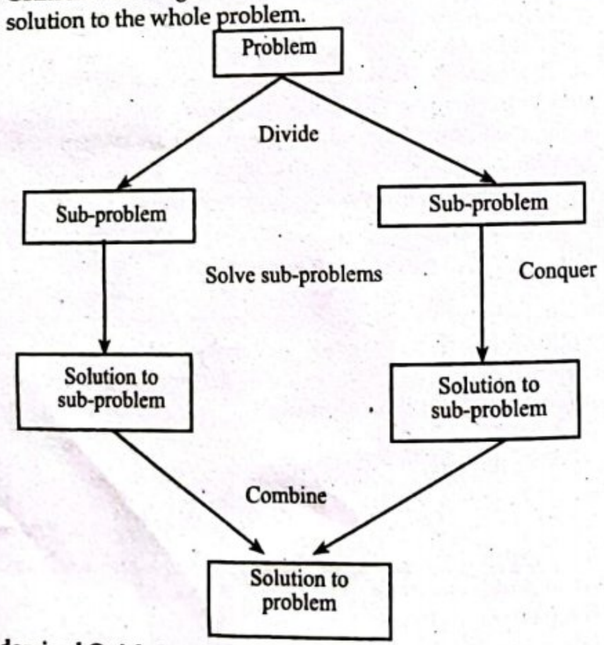
return statement takes 1 step
Total time complexity = 1+1+1+1+n+1+n+n+1+1
= 2n+7
= O(1) x O(n) +O(1)
= O(n) +O(1)
= O(n)

Similarly **space complexity** is,
Total memory references used = 3
= O(1)

2. **Explain about the divide and conquer paradigm for an algorithm design with suitable example. Write the Quick sort algorithm using randomized approach and explain its time complexity.** (4+6)

Ans: Divide and Conquer is an algorithmic pattern. In algorithmic methods, the design is to take a dispute on a huge input, break the input into minor pieces, decide the problem on each of the small pieces, and then merge the piecewise solutions into a global solution. This mechanism of solving the problem is called the Divide & Conquer Strategy. Divide and Conquer algorithm consists of a dispute using the following three steps.

- **Divide** the original problem into a set of sub-problems.
- **Conquer:** Solve every sub-problem individually, recursively.
- **Combine:** Put together the solutions of the sub-problems to get the solution to the whole problem.



Randomized Quick Sort

The algorithm is called randomized if its behavior depends on input as well as random value generated by random number generator. The beauty of the randomized algorithm is that no particular input can produce worst-case behavior of an algorithm. IDEA: Partition around a random element. Running time is independent of the input order. No assumptions need to be made about the input distribution. No specific input elicits the worst-case behavior. The worst case is determined only by the output of a random-number generator. Randomization cannot eliminate the worst-case but it can make it less likely!

Algorithm:

```
RandQuickSort(A,l,r)
{
    if(l<r)
    {
    m = RandPartition (A, l, r);
    RandQuickSort (A, l, m-1);
    RandQuickSort (A, m+1, r);
    }
}
RandPartition (A, l, r)
{
    k = random (l, r); //generates random number between i and j including both.
    swap(A[l],A[k]);
    return Partition(A, l, r);
}
Partition (A, l, r)
{
    x =l; y =r; p = A[l];
    while(x<y)
    {
    do {
    x++;
      }while(A[x] <= p);
    do {
    y--;
      } while(A[y] >=p);
    if(x<y)
    swap(A[x],A[y]);
    }
    A[l] = A[y]; A[y] = p;
    return y;  //return position of pivot
}
```

Time Complexity

Worst Case

T(n) = worst-case running time

Let k be the partition element then there are two sub problems of size k and (n-k), Since there are n elements so we need at most O(n) time for dividing.

Thus their recurrence relation can be defined as,

$T(n) = \max_{1 \le k \le n-1} (T(k) + T(n-k)) + O(n)$ .....(1)

Where, k is some partitioned point produced by random number generator

Now, by using substitution method to show that the running time of Quick sort is $O(n^2)$

Guess $T(n) = O(n^2)$

$\Rightarrow$ $T(n) \le cn^2$ .....(2)

Now proof this by using mathematical induction

Basic step: for n=1,

$T(1) \leq c.\ 1^2$

Or $1 \leq c$ which is true for $c > 0$

Inductive step:

Let's assume that it is true for all $k < n$

I.e. $T(k) \leq ck^2$ for any $k < n$

It is also true for $k=n-k$,

I.e. $T(n-k) \leq c\ (n-k)^2$

Now equation 1 becomes,

$T(n) \leq \max_{1 \leq k \leq n-1} (ck^2 + c(n-k)^2) + O(n)$

$= c \cdot \max_{1 \leq k \leq n-1} (k^2 + (n-k)^2) + O(n)$

The expression $k^2 + (n-k)^2$ achieves a maximum over the range $1 \leq k \leq n-1$ at one of the endpoints

$\max_{1 \leq k \leq n-1} (k^2 + (n-k)^2) = 1^2 + (n-1)^2 = n^2 - 2(n-1)$

$T(n) \leq cn^2 - 2c(n-1) + O(n)$

$\leq cn^2$

$\Rightarrow$ $T(n)=O(n^2)$

3. Explain in brief about the Dynamic Programming Approach for algorithm design. How it differs with recursion? Explain the algorithm for solving the 0/1 knapsack problem using the dynamic programming approach and explain its complexity. (2+2+6)

Ans: Dynamic Programming is the most powerful design technique for solving optimization problems. Dynamic Programming is used when the sub-problems are not independent, e.g. when they share the same sub-problems. In this case, divide and conquer may do more work than necessary, because it solves the same sub problem multiple times. Dynamic Programming solves each sub-problem just once and stores the result in a table so that it can be repeatedly retrieved if needed again. Dynamic Programming is a Bottom-up approach- we solve all possible small problems and then combine to obtain solutions for bigger problems. Dynamic Programming is a paradigm of algorithm design in which an optimization problem is solved by a combination of achieving sub-problem solutions and appearing to the "principle of optimality".

Characteristics of Dynamic Programming

Dynamic Programming works when a problem has the following features:

- **Optimal Substructure:** If an optimal solution contains optimal sub solutions then a problem exhibits optimal substructure.
- **Overlapping sub-problems:** When a recursive algorithm would visit the same sub-problems repeatedly, then a problem has overlapping sub-problems.

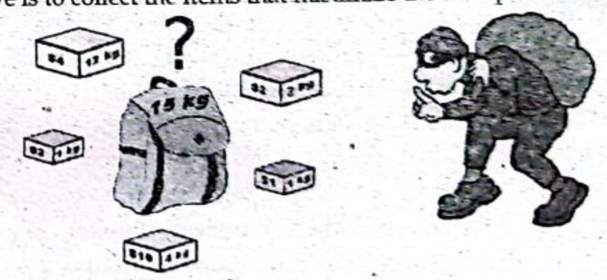
If a problem has optimal substructure, then we can recursively define an optimal solution. If a problem has overlapping sub-problems, then we can improve on a recursive implementation by computing each sub-problem only once

Divide and conquer algorithm vs. Dynamic Programming

Divide & Conquer Method	Dynamic Programming
1. It deals (involves) three steps at each level of recursion: • Divide the problem into a number of sub-problems. • Conquer the sub-problems by solving them recursively. • Combine the solution to the sub-problems into the solution for original sub-problems.	1. It involves the sequence of four steps: • Characterize the structure of optimal solutions. • Recursively defines the values of optimal solutions. • Compute the value of optimal solutions in a Bottom-up minimum. • Construct an Optimal Solution from computed information.
2. It is Recursive.	2. It is non Recursive.
3. It does more work on sub-problems and hence has more time consumption.	3. It solves sub-problems only once and then stores in the table.
4. It is a top-down approach.	4. It is a Bottom-up approach.
5. In this sub-problems are independent of each other.	5. In this sub-problems are interdependent.
6. For example: Merge Sort & Binary Search etc.	6. For example: Matrix Multiplication.

0/1 knapsack problem

A thief has a bag or knapsack that can contain maximum weight W of his loot. There are n items and the weight of i^{th} item is w_i and it worth v_i. An amount of item can be put into the bag is 0 or 1 i.e. x_i is 0 or 1. Here the objective is to collect the items that maximize the total profit earned.



Let W = Capacity of Knapsack

n = No. of items

w = {w1, w2... wn} = weights of items

V= {v1, v2, v3... vn} = value of items

C [i, w] = maximum profit earned with item i and with knapsack of capacity w

Then the recurrence relation for 0/1 knapsack problem is given as,

$$C[i,w] = \begin{cases} 0 & \text{if } = 0 \text{ or } w = 0 \\ C[i-1,w] & \text{if } wi > w \\ \max\{vi + C[i-1,w-wi], c[i-1,w]\} & \text{if } i > 0 \text{ and } wi \leq w \end{cases}$$

```
Algorithm
DynaKnapsack(W, n, v, w)
{
for(w=0; w<=W; w++)
C[0,w] = 0;
for(i=1; i<=n; i++)
C[i,0] = 0;
for(i=1; i<=n; i++)
{
for(w=1; w<=W;w++)
{
if(w[i]<w)
{
if v[i] +C[i-1,w-w[i]] > C[i-1,w]
C[i,w] = v[i] +C[i-1,w-w[i]];
else
C[i,w] = C[i-1,w];
}
else
C[i,w] = C[i-1,w];
}
}
}
```

Analysis

For run time analysis examining the above algorithm the overall run time of the algorithm is O(nW).

Section B

Attempt any eight questions : (8×5=40)

4. Explain the recursion tree method for solving the recurrence relation. Solve following recurrence relation using this method. (2+3)

T(n) = 2T(n/2)+1 for n>1, T(n) = 1 for n = 1

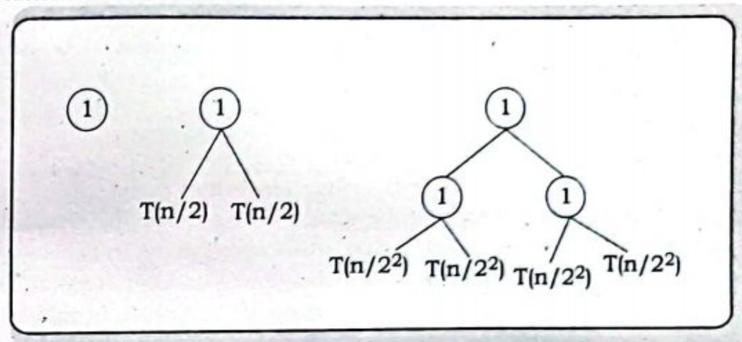
Ans: Recursion Tree

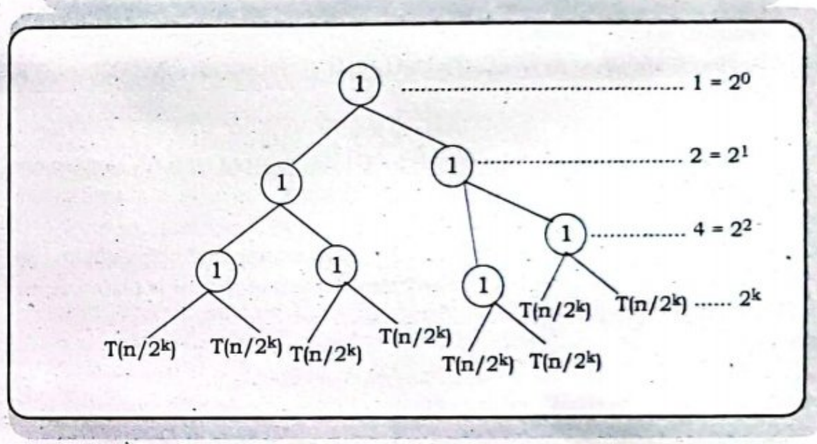
Recursion Tree Method is a pictorial representation of an iteration method which is in the form of a tree where at each level nodes are expanded. In general, we consider the second term in recurrence as root. It is useful when the divide & Conquer algorithm is used.

It is sometimes difficult to come up with a good guess. In Recursion tree, each root and child represents the cost of a single sub-problem. We sum the costs within each of the levels of the tree to obtain a set of pre-level costs and then sum all pre-level costs to determine the total cost of all levels of the recursion.

Solve following recurrence relation by using recursion tree method
T (n) = 2 T (n/2) + 1 when n>1

Solution:





Now $T(n) = 2^0 + 2^1 + 2^2 + \ldots\ldots\ldots\ldots + 2^k$
$= 1+2[2^k-1] / 2-1$
$= 1+ 2(2^k-1)$
$= 2*2^k -1$

For simplicity assume that $n/ 2^K=1$

⇨ $n=2^K$

Taking log on both sides,

⇨ $\log n=\log 2^K$

⇨ $k \log 2=\log n$

⇨ $k= \log n$

Now $T(n) = 2n - 1$

Hence, $T(n) = O(n)$

5. Write an algorithm to find the maximum element of an array and analyze its time complexity. (5)

Ans: Algorithm

1. Start
2. Create a local variable max to store the maximum among the list
3. Initialize max with the first element initially, to start the comparison.
4. Then traverse the given array from second element till end, and for each element:
   - Compare the current element with max

7. What do you mean by optimization problem? Explain the greedy strategy for algorithm design to solve optimization problems. (1+4)

Ans: An optimization problem is the problem of finding the best solution from all feasible solutions. Optimization problems can be divided into two categories depending on whether the variables are continuous or discrete. An optimization problem with discrete variables is known as a discrete optimization. In a discrete optimization problem, we are looking for an object such as an integer, permutation or graph from a countable set. Problems with continuous variables include constrained problems and multimodal problems.

An optimal solution is a feasible solution where the objective function reaches its maximum (or minimum) value – for example, the most profit or the least cost. A globally optimal solution is one where there are no other feasible solutions with better objective function values. A locally optimal solution is one where there are no other feasible solutions in the vicinity with better objective function values.

najikae chai xaenw

Greedy strategy for algorithm design to solve optimization problems

Among all the algorithmic approaches, the simplest and straightforward approach is the Greedy method. In this approach, the decision is taken on the basis of current available information without worrying about the effect of the current decision in future.

Greedy algorithms build a solution part by part, choosing the next part in such a way, that it gives an immediate benefit. This approach never reconsiders the choices taken previously. This approach is mainly used to solve optimization problems. Greedy method is easy to implement and quite efficient in most of the cases. Hence, we can say that Greedy algorithm is an algorithmic paradigm based on heuristic that follows local optimal choice at each step with the hope of finding global optimal solution.

In many problems, it does not produce an optimal solution though it gives an approximate (near optimal) solution in a reasonable time

Example: let's assume that there are 4 jobs

n = 4

J = (j1, j2, j3, j4)

D = (d1, d2, d3, d4) = (2, 1, 2, 1)

P = (100, 10, 15, 27)

Find the sequence due to which maximize the profit by using job sequencing with deadline algorithm

Solution: According to greedy algorithm for this problem, at first sort the jobs on the basis of profit in descending order as,

J = (j1, j4, j3, j2)

D = (d1, d4, d3, d2) = (2, 1, 2, 1)

P = (100, 27, 15, 10)

Job	Feasible/Non-feasible	Processing sequence	Total profit
j1	Feasible	{j1}	100
j4	Feasible	{j4, j1}	100+27=127
j3	Not feasible	{j4, j1}	127
j2	Not feasible	{j4, j1}	127

Thus the optimal profit=127 with the processing sequence= {j4, j1}

CamScanner

8. Explain the algorithm and its complexity for solving job sequencing with deadline problem using greedy strategy. (5)

Ans: The problem is the number of jobs, their profit and deadlines will be given and we have to find a sequence of jobs, which will be completed within their deadlines and it should yield a maximum profit.

Let there are a number of jobs $J = \{j_1, j_2, j_3, j_4 \ldots\ldots, j_n\}$

Deadline of jobs = $\{d_1, d_2, d_3, d_4 \ldots d_n\}$

The profit can be earned if job is completed within their deadline = $\{p_1, p_2 \ldots p_n\}$

Here every job can be completed in unit time (i.e. first job begins at time 0 and finished at time 1, the second job begins at time 1 and finished at time 2 and so on.) and we have a single machine (processor). The main aim of this problem is to find the feasible sequence of jobs that maximize the profit earned.

Features of algorithm

- There are n jobs to be processes on a machine
- Each job i has a deadline $d_i \geq 0$ and profit $P_i \geq 0$
- Pi is earned iff the job is completed by its deadline
- The job is completed if it is processed on a machine for unit time
- Only one machine is available for processing jobs
- Only one job is processed at a time on the machine

Example: let's assume that there are 4 jobs

n = 4

$J = (j_1, j_2, j_3, j_4)$

$D = (d_1, d_2, d_3, d_4) = (2, 1, 2, 1)$

$P = (100, 10, 15, 27)$

Find the sequence due to which maximize the profit by using job sequencing with deadline algorithm

Solution: According to greedy algorithm for this problem, at first sort the jobs on the basis of profit in descending order as,

$J = (j_1, j_4, j_3, j_2)$

$D = (d_1, d_4, d_3, d_2) = (2, 1, 2, 1)$

$P = (100, 27, 15, 10)$

Job	Feasible/Non-feasible	Processing sequence	Total profit
j_1	Feasible	$\{j_1\}$	100
j_4	Feasible	$\{j_4, j_1\}$	100+27=127
j_3	Not feasible	$\{j_4, j_1\}$	127
j_2	Not feasible	$\{j_4, j_1\}$	127

Thus the optimal profit = 127 with the processing sequence = $\{j_4, j_1\}$

Algorithm

Let us consider, a set of n given jobs which are associated with deadlines and profit is earned, if a job is completed by its deadline. These jobs need to be ordered in such a way that there is maximum profit. Assume, deadline of the job J_i is D_i and the profit received from this job is P_i. Hence, the optimal solution of this algorithm is a feasible solution with maximum profit.

```
Job-Sequencing-With-Deadline (D, J, n, k)
{
D(0) = J(0) = 0
k = 1
J(1) = 1  // means first job is selected
for i = 2 ... n do
 r = k
while D(J(r)) > D(i) and D(J(r)) ≠ r do
r = r – 1
if D(J(r)) ≤ D(i) and D(i) > r then
for l = k ... r + 1 by -1 do
J(l + 1) = J(l)
J(r + 1) = i
k = k + 1
}
```


Analysis

In this algorithm, we are using two loops, one is within another. Hence, the complexity of this algorithm is $O(n^2)$.

9. What do you mean by memoization strategy? Compare memoization with dynamic programming. (2+3)

Ans: Memoization means recording the results of earlier calculations so that we don't have to repeat the calculations later. If our code depends on the results of earlier calculations, and if the same calculations are performed over-and-over again, then it makes sense to store interim results so that we can avoid repeating the math.

So far we have talked about implementing dynamic programming in a bottom up fashion. Dynamic programming can also be implemented using memoization. With memoization, we implement the algorithm recursively, but we keep track of all of the sub solutions. If we encounter a sub-problem that we have seen, we look up the solution. If we encounter a sub-problem that we have not seen, we compute it, and add it to the list of sub solutions we have seen. Each subsequent time that the sub-problem is encountered, the value stored in the table is simply looked up and returned.

Memoization offers the efficiency of dynamic programming. It maintains the top-down recursive strategy.

Dynamic Programming vs. Memoization

Dynamic programming algorithm usually outperforms a top-down memoization algorithm by constant factor, because there is no over-head for recursion and fewer overheads for maintaining the table. In situations where not every sub-program is computed, memoization only solves those that are needed but dynamic programming solves all the sub-problems.

In summary, the matrix chain multiplication problem can be solved in $O(n^3)$ times by either a top-down, memorized algorithm or a bottom-up dynamic programming algorithm.

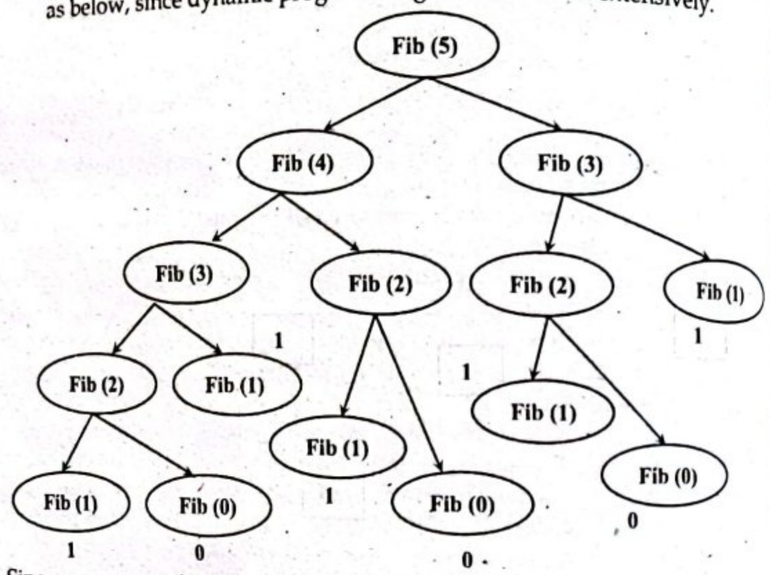
Let's take a problem of finding Fibonacci number by using recursion as below,

```
int Fibo(int n)
{
if(n<=1)
return n;
else
return Fibo(n-1)+Fibo(n-2);
}
```

Their recurrence relation is,

$$Fibo(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ Fibo(n-1) + Fibo(n-2) & \text{if } n \geq 2 \end{cases}$$

If we use dynamic programming for calculation Fibonacci number it work as below, since dynamic programming uses recursion extensively.



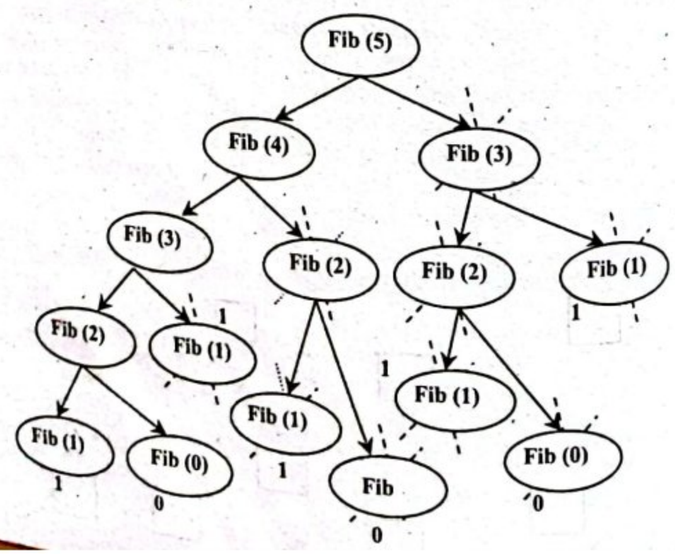
Since recurrence relation for this problem is,

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

By solving this we get,

$T(n)= O(2^n)$

Now using Memoization



0	1	1	-1	-1	-1
0	1	2	3	4	5

0	1	1	2	-1	-1
0	1	2	3	4	5

0	1	1	2	3	-1
0	1	2	3	4	5

0	1	1	2	3	5
0	1	2	3	4	5

From the above tracing of Fibonacci series by using Memoization there are only 6 calls are occur for n=5

Hence for a problem of size n there are (n+1) calls may occur.

Thus total time complexity,

$T(n) = n + 1 = O(n) + O(1) = O(n)$

⇨ $T(n) = O(n)$

10. Explain the concept of backtracking. How it differ with recursion? (2+3)

Ans: Same as TU 2076 Q No. 12 a.

11. Explain in brief about the complexity classes P, NP and NP Complete. (5)

Ans: Same as TU 2076 Q. No. 11

12. Write short notes on: (2*2.5)

a. NP Hard Problems and NP Completeness

Ans: Same as TU 2076 Q. No. 11

b. Problem Reduction

Ans: Reduction

Problem decomposition refers to the problem-solving process that computer scientists apply to solve a complex problem by breaking it down into parts that can be more easily solved. Oftentimes, this involves modifying algorithm templates and combining multiple algorithm ideas those to solve the problem at hand: all of the design work in the projects have been designed with this process in mind.

However, modifying an algorithm template is just one way that we can solve a problem. An alternative approach is to represent it as a different problem, one that can be solved without modifying an algorithm. Reduction is a problem decomposition approach where an algorithm designed for one problem can be used to solve another problem.

- Modify the input so that it can be framed in terms of another problem.
- Solve the modified input using a standard (unmodified) algorithm.
- Modify the output of the standard algorithm to solve the original problem.

MODEL QUESTIONS SETS FOR PRACTICE

MODEL SET 1

Course Title: Design and Analysis of Algorithms
Course No: CSC 314
Semester: V

Full Marks: 6
Pass Marks: 2
Credit Hrs: 3

Section A

Attempt any TWO questions. (2 × 10 = 20)

1. Why do you need the algorithm analysis? Discuss about RAM model for analysis of algorithms. Also discuss about Big Oh, Big Omega and Big theta with examples. (2+3+5
2. Discuss the order statistics. Explain about the worst case linear time selection algorithm and analyze its time complexity. (2+8
3. Explain in brief about the Dynamic Programming Approach for algorithm design. How it differs with recursion? Explain the Floyd Warshall algorithm to compute the all pair shortest path in graph and analyze its time complexity. (4+6

Section B

Attempt any EIGHT questions. (8 × 5 = 40)

4. Write the algorithm for Binary Search with divide and conquer approach and explain its complexity. (5
5. Solve the following recurrence relations using master method. (2.5+2.5
   a. $T(n) = 3T(n/2) + n$
   b. $T(n) = 2T(n/4) + \sqrt{n}$
6. What is prefix code? Explain Huffman algorithm to compute the prefix codes. (5
7. Write the algorithm for insertion sort and explain its time complexity (5
8. What do you mean by memoization strategy? Compare memoization with dynamic programming. (5
9. Explain backtracking with suitable example. (5
10. Explain the Euclid's method to solve the modular linear equations with example (5
11. Explain in brief about the complexity classes P, NP and NP Complete. (5
12. Write short notes on
    a. Tractable and Intractable Problems
    b. Approximation Algorithms

MODEL SET 2

Section A

Attempt any two questions. (2 × 10 = 20)

1. What is a backtracking? Describe the backtracking 0/1 Knapsack Problem. Find an optimal solution for the backtracking 0/1 knapsack instance for n = 3, m = 6, profits are $(p_1, p_2, p_3) = (1, 2, 5)$, weights are $(w_1, w2, w_3) = (2, 3, 4)$.
2. Define greedy algorithm. Describe their characteristics. let's assume that there are 4 jobs
   n = 4
   $S = (p_1, p_2, p_3, p_4)$
   $D = (d_1, d_2, d_3, d_4) = (2, 1, 3, 4)$
   P = (100, 10, 15, 27)
   Find the sequence due to which maximize the profit by using job sequencing with deadlines.
3. Define heap sort. Write down the algorithm for heap sort and analyze it. Also sort following data items by using heap sort algorithm
   A [ ] = {3, 5, 2, 66, 4, 11, 9, 34}

Section B

Attempt any eight questions. (8 × 5 = 40)

4. Define binary search. Writhe down the recursive algorithm for binary search algorithm and analyze it.
5. Given a set S = {6, 4, 5, 6, 9} and X = 11. Obtain the subset sum using backtracking approach.
6. What is shortest path problem? Explain Dijkstra algorithm with suitable example.
7. What is the importance of asymptotic notations in DAA? Write down the algorithm for selection sort then find their tighter big oh by using RAM model.
8. Define selection sort algorithm. Write down algorithm for selection sort and analyze it.
9. Find optimal sequence of matrices A, B, C and D of order respectively 3 x 4, 4 x 2, 2 x 3 and 3 x 4 by using matrix chain multiplication.
10. Write down the applicable examples for class P, class NP and class NP hard problems.
11. What is the purpose of Euclid's algorithm? Explain with suitable example.
12. Argue that the solution to the recurrence $T(n) = T(n/3) + T(2n/3) + n$ is $(n \log n)$ by appealing to a recursion tree.

MODEL QUESTIONS SETS FOR PRACTICE

MODEL SET 1

Course Title: Design and Analysis of Algorithms
Course No: CSC 314
Semester: V

Full Marks: 60
Pass Marks: 24
Credit Hrs: 3

Section A

Attempt any TWO questions. **(2 × 10 = 20)**

1. Why do you need the algorithm analysis? Discuss about RAM model for analysis of algorithms. Also discuss about Big Oh, Big Omega and Big theta with examples. (2+3+5)
2. Discuss the order statistics. Explain about the worst case linear time selection algorithm and analyze its time complexity. (2 + 8)
3. Explain in brief about the Dynamic Programming Approach for algorithm design. How it differs with recursion? Explain the Floyd Warshall algorithm to compute the all pair shortest path in graph and analyze its time complexity. (4 + 6)

Section B

Attempt any EIGHT questions. **(8 × 5 = 40)**

4. Write the algorithm for Binary Search with divide and conquer approach and explain its complexity. (5)
5. Solve the following recurrence relations using master method. (2.5+2.5)
   a. $T(n) = 3T(n/2) + n$
   b. $T(n) = 2T(n/4) + \sqrt{n}$
6. What is prefix code? Explain Huffman algorithm to compute the prefix codes. (5)
7. Write the algorithm for insertion sort and explain its time complexity (5)
8. What do you mean by memoization strategy? Compare memoization with dynamic programming. (5)
9. Explain backtracking with suitable example. (5)
10. Explain the Euclid's method to solve the modular linear equations with example (5)
11. Explain in brief about the complexity classes P, NP and NP Complete. (5)
12. Write short notes on
    a. Tractable and Intractable Problems
    b. Approximation Algorithms

MODEL SET 2

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. What is a backtracking? Describe the backtracking 0/1 Knapsack Problem. Find an optimal solution for the backtracking 0/1 knapsack instance for n = 3, m = 6, profits are $(p_1, p_2, p_3) = (1, 2, 5)$, weights are $(w_1, w2, w_3) = (2, 3, 4)$.
2. Define greedy algorithm. Describe their characteristics. let's assume that there are 4 jobs
   n = 4
   $S = (p_1, p_2, p_3, p_4)$
   $D = (d_1, d_2, d_3, d_4) = (2, 1, 3, 4)$
   P = (100, 10, 15, 27)
   Find the sequence due to which maximize the profit by using job sequencing with deadlines.
3. Define heap sort. Write down the algorithm for heap sort and analyze it. Also sort following data items by using heap sort algorithm
   A [ ] = {3, 5, 2, 66, 4, 11, 9, 34}

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. Define binary search. Writhe down the recursive algorithm for binary search algorithm and analyze it.
5. Given a set S = {6, 4, 5, 6, 9} and X = 11. Obtain the subset sum using backtracking approach.
6. What is shortest path problem? Explain Dijkstra algorithm with suitable example.
7. What is the importance of asymptotic notations in DAA? Write down the algorithm for selection sort then find their tighter big oh by using RAM model.
8. Define selection sort algorithm. Write down algorithm for selection sort and analyze it.
9. Find optimal sequence of matrices A, B, C and D of order respectively 3 x 4, 4 x 2, 2 x 3 and 3 x 4 by using matrix chain multiplication.
10. Write down the applicable examples for class P, class NP and class NP hard problems.
11. What is the purpose of Euclid's algorithm? Explain with suitable example.
12. Argue that the solution to the recurrence $T(n) = T(n/3) + T(2n/3) + n$ is $(n \log n)$ by appealing to a recursion tree.

MODEL SET 3

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Write down the elements of dynamic programming. Also mention their advantages and disadvantages. Give the recursive definition of LCS problem. Find LCS between sequences S1="Dinesh", S2 ="Dikshya"
2. Define divide and conquer algorithm. How it is differ from dynamic programming? Write Divide and Conquer recursive Merge sort algorithm and derive the time complexity of this algorithm.
3. Define RAM model. Why do you need the algorithm analysis? Write recurrence relation for following segment of code and then find their big oh.

```
void main ( )
{
int L, M, N, i, j, k, a[ ], b[ ], c[ ];
printf ("Enter value of L, M and N");
scanf ("%d %d %d", &L, &M, &N);
for (i=0; i<=L; i++)
{
for (j=1; j<=M; j++)
{
b[j]=2*j;
}
}
for (k=N; k>=0; k--)
c[k]=-k;
}
```

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. Differentiate between Euclid's algorithm and extended Euclid's algorithm with suitable example.
5. Explain approximation algorithm with an appropriate example.
6. Differentiate between dynamic programming vs. memorization with suitable example.
7. Trace the insertion sort algorithm for following algorithm,
   A[ ] = {1, 23, 21, 66, 22, 14, 98, 45, 78}
8. The running time of an algorithm A is described by the recurrence $T(n) = 7T(n/2) + n2$. A competing algorithm A' has a running time of $T'(n) = a\ T'(n/4) + n^2$. What is the largest integer value for 'a' such that A' is asymptotically faster than A?
9. Define build heap operation. Construct heap of any 10 elements by using build heap operation.
10. Write an algorithm for N – queen's problem. Give time and space complexity for 8 – queen's problem.
11. Differentiate between Preem's algorithm and Kruskal's algorithm for finding minimum spanning tree of given graph.
12. Find big oh and big omega of following function,
    $F(x) = 5n^3 + 6n^2 + 9n + 3$

MODEL SET 4

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. What are the characteristics of problem that can be solved by using dynamic programming algorithm? Give the recursive definition of solving 0/1 knapsack problem. Trace the algorithm for w = {3, 4, 2, 2, 3}, v = {12, 14, 6, 5, 6} and knapsack of capacity 12.
2. Write Divide and Conquer recursive Quick sort algorithm and merge sort algorithm and analyze the algorithms.
3. What is a backtracking? Give the explicit and implicit constraints in 8 queen's problem.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. Write down the algorithm for Miller-Rabin Randomized Primality Test and analyze it.
5. Show that the decision version of the set –covering problem is NP-complete by reduction from the vertex cover problem.
6. What is all pair shortest path problem? Find all pair shortest path of given weighted graph by using Floyd Warshall algorithm.
7. What is detailed analysis of algorithm? Detailed analyze the sequential search algorithm.
8. Give the statement of sum –of subsets problem. Find all sum of subsets for n = 4, (w_1, w_2, w_3, w_4) = (11, 13, 24, 7) and M=31. Draw the portion of the state space tree using fixed – tuple sized approach.
9. Define heap sort. Sort following data items by using heap sort algorithm
   A[ ] = {3, 5, 2, 66, 4, 11, 9, 34}
10. Bubble sort is called one of the worst sorting algorithms. Justify
11. Define the terms "Class P", "Class NP" and "NP-Completeness".
12. Differentiate between Preem's algorithm and Kruskal's algorithm for finding minimum spanning tree of given graph.

MODEL SET 5

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Why asymptotic notations are important in algorithm analysis? Describe big-Oh, big-omega and big- theta notation with suitable examples.
2. What is recurrence relation? Prove that the complexity of the recurrence relation "$T(n) = 8T(n/2) + n^2$" is $O(n^3)$ by using substitution method.
3. Given the following block of code, write a recurrence relation for it and also find asymptotic upper bound (Assume that all dotted code takes constant time)

```
Fun(int n)
{
.....
```

```
if(condition1)
x=Fun(n/2)
else if(condition2)
x=Fun(2n/3)
else
x= Fun(n/4)
.....
}
```

Section B

Attempt any eight questions. (8 × 5 = 40)

4. What is the concept behind randomized quick sort? Write down its algorithm and give its average case analysis.
5. What is meant by medial order statistics? Write the algorithm for expected liner time selection and analyze it.
6. Devise a divide and conquer algorithm for finding minimum and maximum element among a set of given elements. Write recurrence relation for your algorithm and give its big-O estimate.
7. What are the characteristics of problem that can be solved by using dynamic programming algorithm? Give the recursive definition of solving 0/1 knapsack problem. Trace the algorithm
8. for w={3,4,2,2,3}, v={12,14,6,5,6} and knapsack of capacity 12. (2+1+5)
9. Write the recurrence relation for Longest Common subsequence problem (LCS). Trace the algorithm to find LCS of X={a, b, c, b, d, a, b} and Y={b, d, c, a, b, a}. Use master method to find the big-O estimates of the recurrences: (4+4)
   a. $T(n) = 3T(n/2) + n$
   b. $T(n) = 4T(n/2) + n_2$
10. Show all the steps required for sorting an array of size 10 by using Heap sort. a[10]={5, 3, 2, 4, 7, 8, 1, 11, 9, 15}.
11. What is Minimum Spanning Trees? Write down algorithm for prem's algorithm for MST and analyze it.
12. Write the EUCLID'S GCD algorithm. Compute gcd(99, 78) with EXTENDED-EUCLID.

MODEL SET 6

Section A

Attempt any two questions. (2 × 10 = 20)

1. When and how Dynamic Programming Approach is applicable? Discuss the matrix chain multiplication with respect to dynamic programming technique.
2. Define minimum cost spanning tree. Write Prim's algorithm to generate a minimum cost spanning tree for any given weighted graph. Generate minimum cost spanning tree for a graph with 5 vertices and 9 weighted edges using Prim's algorithm.
3. Explain different constraints used to solve a problem by Backtracking. Solve sum of subset problem using backtracking.

Section B

Attempt any eight questions. (8 × 5 = 40)

4. What is an optimal Huffman code for the following set of frequencies, based on the first 8 Fibonacci numbers? A: 1, B: 1, C: 2, D: 3, E: 5, F: 8, G: 13, H: 21.
5. State the master method for solving the recurrence relation. Solve the recurrence relation $T(n) = 2T(n/2) + n \log n$ by using master method as well as recursion tree method.
6. What is heap? Sort following data items by using heap sort. A[ ]={4, 55, 6, 9, 11, 2, 44}
7. Define a Knapsack problem and describe its formulation. Find the optimal solution to the knapsack instance n = 5, W = {20, 30, 40, 10, 7}, P = {7, 8, 9, 1, 6} and C=80 using greedy method.
8. Define main characteristics of an algorithm. Discuss time and space complexity of bubble sort algorithm by using RAM model.
9. Solve the recurrence relation $T(n) = T(n-1) + T(n-2) + 1$ when $T(0) = 0$ and $T(1) = 1$.
10. Describe the Warshall's and Floyd's algorithm for finding all pairs shortest path.
11. Explain NP hard and NP complete problems and also define the polynomial time problems and write a procedure to solve NP problems.
12. How to find complexity of iterative algorithm? Also how to find complexity of recursive algorithms? Write down the algorithm for randomised quick sort and analyze it.

MODEL SET 7

Section A

Attempt any two questions. (2 × 10 = 20)

1. What is the importance of asymptotic notations in DAA? Write down the algorithm for selection sort then find their tighter big oh by using RAM model.
2. Make tight big oh analysis of following code segment.

```
void main ( )
{
int sum=0,i,j, a[ ][ ];
for(i=0; i<n;i++)
{
for(j=1; j<n-i; j++)
{
sum=sum+a[i][j];
}
}
}
```

3. What do you mean by n^{th} order statistics? Describe expected linear time selection problem and give its big-O estimate.

Section B

Attempt any EIGHT questions. **(8 × 5 = 40)**

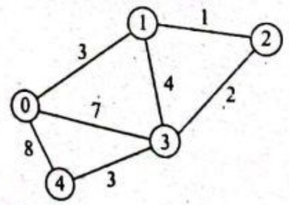
4. What is heap? How it is differ from BST? Sort following data items by using heap sort. A[ ] = {4, 5, 22, 12, 6, 45, 89, 33, 1}
5. What are the advantages of dynamic programming? Give the recursive definition of LCS problem. Find LCS between sequences S_1 = "Dinesh", S_2 = "Dikshya" Define greedy paradigm. How can you define Huffman code in greedy algorithm? Explain with a suitable example.
6. What do you mean by spanning tree? Define Prim's algorithm for MST with suitable example.
7. Compare Dijkstra algorithm and DAG algorithm for finding shortest path problem. How DAG works for fining single source shortest path of given directed acyclic graph?
8. Why asymptotic notations are important in algorithm analysis? Given the function prove that : f(n)=O(g(n)), f(n)= Ω(g(n)), f(n)=Θ(g(n))
   $f(n) = 9n^3 + 2n + 5$, $g(n) = n^3$
9. Define class P, class NP and class NP complete. How vertex covering problem work? Describe with suitable example.
10. Define substitution method of solving recurrence relation. Write down recurrence relation for finding nth term of Fibonacci series. Prove that the complexity of the recurrence relation T (n) = 3T (n/2) +n 2 is O (n 2) by using substitution method.
11. Define greedy paradigm. How can you define Huffman code in greedy algorithm? Explain with a suitable example.
12. Find GCD (198, 128) and value of x and y by using extended Euclidean's algorithm.

MODEL SET 8

Section A

Attempt any TWO questions. **(2 × 10 = 20)**

1. Define greedy algorithm. Describe their characteristics. Find minimum spanning tree of following graph by using Kruskal's algorithm.



2. Differentiate between internal and external sort algorithm. Write down the algorithm for insertion sort then find their best case, average case and worst case time complexity.
3. Write down the elements of dynamic programming. Also mention their advantages and disadvantages. Find optimal sequence of matrices A, B, C and D of order respectively 3 x 4, 4 x 2, 2 x 3 and 3 x 4 by using matrix chain multiplication.

Section B

Attempt any EIGHT questions. **(8 × 5 = 40)**

4. Distinguish between backtracking and branch - and bound techniques. Discuss the 4 - queen's problem. Draw the portion of the state space tree for n= 4 queens using backtracking algorithm.
5. Explain approximation algorithm with an appropriate example.
6. What is selection? How it is differ from searching? Explain linear time selection algorithm with example.
7. Define binary search. Writhe down the recursive algorithm for binary search algorithm and analyze it.
8. Does Prim's and Kruskal's algorithm work if negative weights are allowed? Explain
9. What is a Backtracking and give the 4 - Queens's solution.
10. Devise a divide and conquer algorithm for finding minimum and maximum element among a set of given elements. Write recurrence relation for your algorithm and give its big-O estimate.
11. What is all pair shortest path problem? Explain Floyd's Warshall algorithm for all pair shortest path problem with a suitable example.
12. What is the main concept behind greedy algorithm? State fractional knapsack problem. By using greedy algorithm find the optimal (maximum) profit earned for following item sets. Use knapsack capacity (W)=90
    Items (I) = {I1, I2, I3, I4, I5}
    Weight (w) = {5, 10, 20, 30, 40}
    Value (v) = {30, 20, 100, 90, 160}

□□□

New Syllabus

Course Title: Cryptography
Course No: CSC316
Nature of the Course: Theory + Lab
Semester: V

Full Marks: 60 + 20 + 20
Pass Marks: 24 + 8 + 8
Credit Hrs: 3

Course Description: The course introduces the underlying the principles and design of cryptosystems. The course covers the basics concepts of cryptography including: traditional ciphers, block ciphers, stream ciphers, public and private key cryptosystems. The course also includes the theory of hash functions, authentication systems, network security protocols and malicious logic.

Course Objectives: The objectives of this course are to familiarize the students with cryptography and its applications. The students will be able to develop basic understanding of cryptographic mechanisms.

Course Contents:

Unit I: Introduction and Classical Ciphers (7 hr)

1.4. Security: Computer Security, Information Security, Network Security, CIA Triad, Cryptography, Cryptosystem, Cryptanalysis, Security Threats and Attacks, Security Services, Security Mechanisms
1.5. Classical Cryptosystems: Substitution Techniques: Ceasar, Monoalphabetic, Playfair, Hill, Polyalphabetic ciphers, One-time pad Transposition Techniques: Rail Fence Cipher
1.6. Modern Ciphers: Block vs. Stream Ciphers, Symmetric vs. Asymmetric Ciphers

Unit II: Symmetric Ciphers (10 hr)

2.4. Fiestel Cipher Structure, Substitution Permutation Network (SPN)
2.5. Data Encryption Standards (DES), Double DES, Triple DES
2.6. Finite Fields: Groups Rings, Fields, Modular Arithmetic, Euclidean Algorithm, Galois Fields (GF(p) & GF(2^n)), Polynomial Arithmetic
2.7. International Data Encryption Standard (IDEA)
2.8. Advanced Encryption Standards (AES) Cipher AES KEY EXPANSION PADNA XA
2.9. Modes of Block Cipher Encryptions (Electronic Code Book, Cipher Block Chaining, Cipher Feedback Mode, Output Feedback Mode, Counter Mode)

Unit III: Asymmetric Ciphers (8 hr)

3.4. Number Theory: Prime Numbers, Fermat's Theorem, Euler's Theorem, Primility Testing, Miller-Rabin Algorithm, Extended Euclidean Theorem, Discrete Logarithms
3.5. Public Key Cryptosystems, Applications of Public Key Cryptosystems
3.6. Distribution of public key, Distribution of secret key by using public key cryptography, Diffie- Helman Key Exchange, Man-in-the-Middle Attack
3.7. RSA Algorithm
3.8. Elgamal Cryptographic System

Unit IV: Cryptographic Hash Functions and Digital Signatures (8 hr)

4.4. Message Authentication, Message Authentication Functions, Message Authentication Codes
4.5. Hash Functions, Properties of Hash functions, Applications of Hash Functions
4.6. Message Digests: MD4 and MD5
4.7. Secure Hash Algorithms: SHA-1 and SHA-2
4.8. Digital Signatures: Direct Digital Signatures, Arbitrated Digital Signature
4.9. Digital Signature Standard: The DSS Approach, Digital Signature Algorithm
4.10. Digital Signature Standard: The RSA Approach

Unit V: Authentication (3 Hrs)

5.4. Authentication System,
5.5. Password Based Authentication, Dictionary Attacks,
5.6. Challenge Response System,
5.7. Biometric System
5.8. Needham-Schroeder Scheme, Kerberos Protocol

Unit VI: Network Security and Public Key Infrastructure (6 Hrs)

6.1. Overview of Network Security
6.2. Digital Certificates and X.509 certificates, Certificate Life Cycle Management
6.3. PKI trust models, PKIX
6.4. Email Security: Pretty Good Privacy (PGP)
6.5. Secure Socket Layer (SSL) and Transport Layer Security (TLS)
6.6. IP Security (IPSec)
6.7. Firewalls and their types

Unit VI: Malicious Logic (3 Hrs)

7.1. Malicious Logic, Types of Malicious Logic: Virus, Worm, Trojan Horse, Zombies, Denial of Service Attacks,
7.2. Intrusion, Intruders and their types, Intrusion Detection System

Laboratory Works:

The laboratory work includes implementing and simulating the concepts of cryptographic algorithms, hash functions, digital signatures, network security protocols and malicious logic. Students are free to use any of the language and platform as per the skills.

Text Book:

1. W. Stallings, *Cryptography and Network Security*, Pearson Education.

Reference Books:

1. William Stallings, *Network Security, Principles and Practice.*
2. Matt Bishop, *Computer Security, Art and Science.*
3. Mark Stamp, *Information Security: Principles and Practices.*
4. Bruce Schneier, *Applied Cryptography.*
5. Douglas. R. Stinson. *Cryptography: Theory and Practice.*
6. B. A. Forouzan, *Cryptography & Network Security*, Tata Mc Graw Hill.

TRIBHUVAN UNIVERSITY

Institution of Science and Technology

TU QUESTIONS-ANSWERS 2076

Bachelor Level/Third Year/Fifth Semester/Science
Computer Science and Information Technology [CSC 317]

Full Marks: 60
Pass Marks: 24
Time: 3 hrs.

Section A
Long Answer Questions

Attempt any TWO questions. [2×10=20]

1. **Among monoalphabetic and polyalphabetic cipher, which one is more vulnerable? Justify your statement. Which types of keys are considered as weak keys in DES? Explain the round operation in IDEA.** [2+2+6]

Ans: Among monoalphabetic and polyalphabetic cipher, monoalphabetic cipher is more vulnerable because of their fixed key substitution which open to many attacks. This type of encryption can be easily broken down using the "Brute Force Algorithm" and frequency analysis attacks.This BFA tries to decrypt the message by trying all the possible combinations. Thus, to prevent this type of attack, the words should be long enough, which is impossible for every word in a sentence.

DES with a cipher key of 56 bit is not safe enough to be used comfortably. Four out of 2^{56} possible keys are called **weak keys**. A weak key is theone that, after parity drop operation, consists either of all 0s, all 1s,or half 0s and half 1s. These keys are shown in figure below.

Keys before parties drop (64 bits)				Actual key (56 bits)	
0101	0101	0101	0101	0000000	0000000
1F1F	1F1F	0E0E	0E0E	0000000	FFFFFFF
E0E0	E0E0	F1F1	F1F1	FFFFFFF	0000000
FEFE	FEFE	FEFE	FEFE	FFFFFFF	FFFFFFF

Figure: Weak Keys

The round keys created from any of these weak keys are the same and have thesame pattern as the cipher key.If we encrypt a block with a weak keyand subsequently encrypt the result with the same weak key, we get the original block.The process creates the same original block if we decrypt the block twice. In other words,each weak key is the inverse of itself $E_k(E_k(P)) = P$..

IDEA uses a 128-bit key and operates on 64-bit blocks. Essentially, it encrypts a 64-bit block of plaintext into a 64-bit block of ciphertext. This input plaintext block is divided into four subblocks of 16 bits each. It consists of a series of eight identical transformations, where each transformation is known as a round, as well as an output transformation, which is known as a half-round. Similar to the 16-bit plaintext block, the ciphertext block is also the exact same size.

A block cipher operates in round blocks, with part of the encryption key, known as **round key**, applied to each round, followed by other mathematical operations. After a certain number of rounds, the ciphertext for that block is generated.

Encryption in IDEA

By using a 128-bit key, IDEA encrypts a 64-bit block of plaintext into a 64-bit block of ciphertext. One process partitions the plaintext block into four 16-bit subblocks for each of the eight complete rounds, namely X1, X2, X3 and X4.

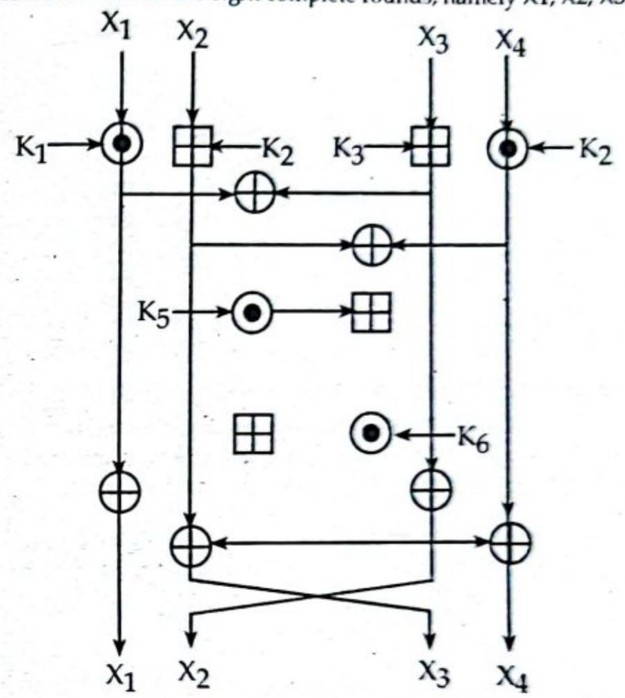


Figure: An Encryption Round of IDEA

Another process produces six 16-bit key subblocks for each of the encryption rounds, namely K1, K2, K3, K4, K5 and K6. For subsequent output transformation, a further four 16-bit key subblocks are required. Thus, from a 128-bit key, a total of 52 16-bit subblocks are generated.

In each complete round, three algebraic operations are performed:

- ☐ XOR Operation (⊕)
  - ✓ Use Bitwise XOR Logic
- ☐ Modulo Multiplication (⊙)
  - ✓ Calculate 32 Bit Product
  - ✓ Result = Product 32 Bit Mod $2^{16}+1$
- ☐ Module Addition (⊞)
  - ✓ Add Two 16 Bit Numbers
  - ✓ Result = Sum Mod 2^{16} (if Carry Occurs)

The 14 steps for a complete round are the following:

1. Multiply X1 and the First subkey K1.
2. Add X2 and the Second subkey K2.
3. Add X3 and the third subkey K3.
4. Multiply X4 and the fourth subkey K4.
5. Bitwise XOR the results of steps 1 and 3.
6. Bitwise XOR the results of steps 2 and 4.

7. Multiply the result of step 5 and the Fifth subkey K5.
8. Add the results of steps 6 and 7.
9. Multiply the result of step 8 and the Sixth subkey K6.
10. Add the results of steps 7 and 9.
11. Bitwise XOR the results of steps 1 and 9.
12. Bitwise XOR the results of steps 3 and 9.
13. Bitwise XOR the results of steps 2 and 10.
14. Bitwise XOR the results of steps 4 and 10

Six subkeys are used in each of the eight rounds, and the final 4 subkeys are used in the ninth half-round final transformation.

Swapping occurs for every round until the final complete round (round 8). After eight complete rounds, the final half-round transformation occurs. The steps involved are the following:

1. Multiply X1 with the first subkey.
2. Add X2 with the second subkey.
3. Add X3 with the third subkey.
4. Multiply X4 with the fourth subkey.

The concatenation of the four blocks is the encrypted output.

2. State the Fermat's theorem with example. Given the prime number p=29 and its primitive root g=8, private key of sender with X=9 and random integer K=11, encrypt the message m=13 using ElGamal cryptosystem. [5+5]

Ans: Fermat's theorem states the following: If p is prime and a is a positive integer not divisible by p, then $a^{p-1} \equiv 1 \pmod p$ [where, a and p are relatively prime]. Therefore, $3^{10} \equiv 1 \pmod{11}$ where p=11. Alternative form of Fermat's theorem is $a^p \equiv a \pmod p$

Given,

p = 29, g=8, private key (X)=9, random integer (K) = 11 and message (m)=13

Now,

Public Key = $(p, g, y=g^x \bmod p)$

$= (29, 8, 8^9 \bmod 29)$

$= (29, 8, 15)$

Again, encrypting the message (m)

$E(m) = (g^k \bmod p, m.y^k \bmod p)$

$= (8^{11} \bmod 29, 13.15^{11} \bmod 29)$

$= (3, 12)$

3. Compare the SHA parameters between SHA-1 and SHA-2 family. Decrypt the cipher textDRJI with the key $\begin{pmatrix} 7 & 8 \\ 11 & 11 \end{pmatrix}$ using the Hill cipher. [3+7]

Ans:

SHA 1	SHA 2
1. SHA 1 generates 160 bits hash value.	1. SHA 2 generates 224-, 256-, 384- or 512-bits hash values.
2. The length output value of SHA 1 is 40 digits.	2. The length output value of SHA 2 is 64 digits
3. Its structure is based on Merkle-Damgard construction.	3. Its structure is based on Merkle-Damgard structure with Davies-Meyer compression function.

Given,

$$k = \begin{pmatrix} 7 & 8 \\ 11 & 11 \end{pmatrix}$$

Step 1: Find multiplicative inverse of determinant of key matrix $\begin{pmatrix} 7 & 8 \\ 11 & 11 \end{pmatrix}$.

$= 7 \times 11 - 8 \times 11 = 15 \bmod 26$

Now, multiplicative inverse of 15 mod 26 is

$15 \times a = 1 \bmod 26$

$\therefore \quad a = 7$

Step 2: find the adj. matrix of Key matrix $\begin{pmatrix} 7 & 8 \\ 11 & 11 \end{pmatrix}$

$$= \begin{pmatrix} 11 & -8 \\ -11 & 7 \end{pmatrix} \bmod 26 = \begin{pmatrix} 11 & 8 \\ 15 & 7 \end{pmatrix}$$

Step 3: Multiply multiplicative inverse of determinant by adj. matrix

$$= 7 \times \begin{pmatrix} 11 & 18 \\ 15 & 7 \end{pmatrix} = \begin{pmatrix} 25 & 22 \\ 1 & 23 \end{pmatrix} \bmod 26$$

Now, cipher text = "DRJI"

$$\begin{pmatrix} D \\ R \end{pmatrix} = \begin{pmatrix} 3 \\ 17 \end{pmatrix} \text{ and } \begin{pmatrix} J \\ I \end{pmatrix} = \begin{pmatrix} 9 \\ 8 \end{pmatrix}$$

Step 4: multiply cipher text (c) by inverse key matrix

$$= \begin{pmatrix} 25 & 22 \\ 1 & 23 \end{pmatrix} \begin{pmatrix} 3 \\ 17 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 75 + 374 \\ 3 + 391 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 449 \\ 394 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 7 \\ 4 \end{pmatrix} = \begin{pmatrix} H \\ E \end{pmatrix}$$

Similarly,

$$\begin{pmatrix} 25 & 22 \\ 1 & 23 \end{pmatrix} \begin{pmatrix} 9 \\ 8 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 225 + 176 \\ 9 + 184 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 401 \\ 193 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 11 \\ 11 \end{pmatrix} = \begin{pmatrix} L \\ L \end{pmatrix}$$

$\therefore$ Plain Text (P) = HELL

SectionB

Short Answer Questions

Attempt any EIGHT questions. [8×5=40]

4. Define discrete logarithm. Explain the procedure of sharing the secret key in Diffie Hellman? [2+3]

Ans: If a is an arbitrary integer relatively prime to n and g is a primitive root of n then there exists among the numbers 0, 1, 2, ..., $\phi(n)$-1, where $\phi(n)$ is the totient function, exactly one number μ such that

$a = g^{\mu} \pmod{n}$.

The number μ is then called the discrete logarithm of a with respect to the base g modulo n and is denoted

$\mu = \text{ind}_g a \pmod{n}$.

Figure below shows the procedure of sharing the secret key in Diffie Hellman

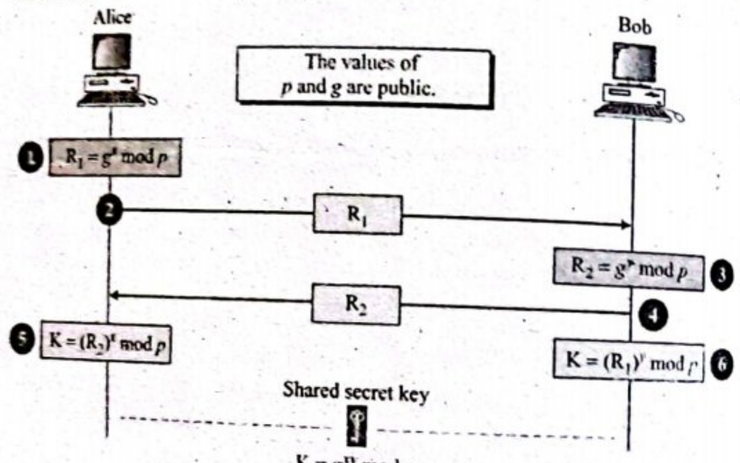


Figure: Diffie-Hellman Method

Steps are as follows:

1. Alice chooses a large random number x such that $0 \le x \le p-1$ and calculates
   $R_1 = g^x \bmod p$.
2. Bob chooses another large random number y such that $0 \le y \le p-1$ and calculates
   $R_2 = g^y \bmod p$.
3. Alice sends R_1 to Bob. Note that Alice does not send the value of x; she sends only R_1.
4. Bob sends R_2 to Alice. Again, note that Bob does not send the value of y, he sendsonly R_2.
5. Alice calculates $K = (R_2)^x \bmod p$.
6. Bob also calculates $K = (R_1)^y \bmod p$.

K is the symmetric key for the session.Bob has calculated $K = (R_1)^y \bmod p = (g^x \bmod p)^y \bmod p = g^{xy} \bmod p$. Alice hascalculated $K = (R_2)^x \bmod p = (g^y \bmod p)^x \bmod = g^{xy} \bmod p$. Both have reachedthe same value without Bob knowing the value of x and without Alice knowing thevalue of y.

For example: Let us give a trivial example to make the procedure clear. Our example uses small numbers, butnote that in a real situation, the numbers are very large. Assume that g = 7 and p = 23. The stepsare as follows:

1. Alice chooses $x = 3$ and calculates $R_1 = 7^3 \bmod 23 = 21$.
2. Bob chooses $y = 6$ and calculates $R_2 = 7^6 \bmod 23 = 4$.
3. Alice sends the number 21 to Bob.
4. Bob sends the number 4 to Alice.
5. Alice calculates the symmetric key $K = 4^3 \bmod 23 = 18$.
6. Bob calculates the symmetric key $K = 21^6 \bmod 23 = 18$.

The value of K is the same for both Alice and Bob; $g^{xy} \bmod p = 7^{18} \bmod 35 = 18$.

5. Distinguish between stream cipher and block cipher. Encrypt the message WE ARE IN SAME RACE UNTILL OUR LIVE END using Rail fence cipher using 4 as number of rails. [2+3]

Answer:

Stream Cipher	Block Cipher
1. Stream Cipher operates on smaller units of Plaintext	1. Block cipher operates on larger block of data
2. Faster than Block Cipher	3. Slower than Stream Cipher
3. Stream Cipher processes the input element continuously producing output one element at a time	3. Block Cipher processes the input one block of element at a time, producing an output block for each input block
4. Require less code	4. Requires more code
5. Only one time of key used. **Example:** One time pad **Application:** SSL (Secure connection on the web)	5. Reuse of key possible **Example:** DES (Data Encryption Standard) **Application:** Database, file encryption
6. Stream Cipher is more suitable for hardware implementation.	6. Easier to implement in software.

Given,

Number of rails = 4

Plaintext = **WE ARE IN SAME RACE UNTILL OUR LIVE END**

W					N				A				I					L							D	
	E			I		S				R		C			T		L			R		I			N	
		A			E			A			E			E			N			L		U			V		E	
			R						M							U					O						E		

Now,

Take characters row-wise,

Ciphertext = WNAILDEISRCTLRINAEAEENLUVERMUOE

6. Define digital signature. Describe the approaches of DSS. [2+3]

Ans: A **digital signature** is a mathematical technique used to validate the authenticity and integrity of a message, software or digital document. As the digital equivalent of a handwritten signature or stamped seal, a digital signature offers far more inherent security, and it is intended to solve the problem of tampering and impersonation in digital communications.

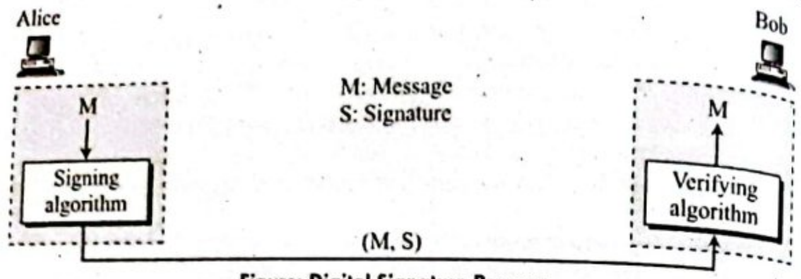


Figure: Digital Signature Process

Digital signatures can provide the added assurances of evidence of origin, identity and status of an electronic document, transaction or message and can acknowledge informed consent by the signer. In many countries, digital signatures are considered legally binding in the same way as traditional document signatures. The digital signature must have the following properties:

- **Authentication:** A digital signature must give the receiver reason to believe the message was created and send by the claimed sender.
- **Non-repudiation:** With digital signature, the sender cannot deny having sent the message later on.
- **Integrity:** A digital Signature must ensure that the message was not altered in transit.

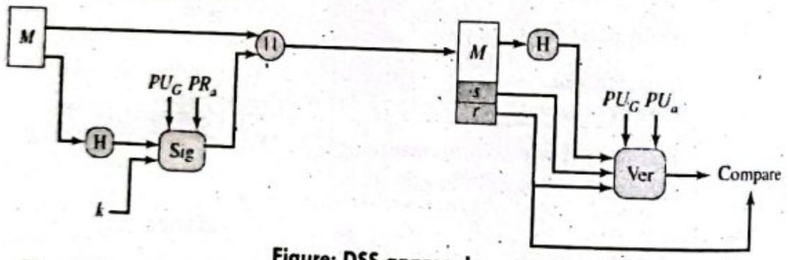


Figure: DSS approach

The DSS approach also makes use of a hash function. The hash code is provided as input to a signature function along with a random number generated for this particular signature. The signature function also depends on the sender's private key and a set of parameters known to a group of communicating principals. We can consider this set to constitute a global public key. The result is a signature consisting of two components, labeled s and r.

At the receiving end, the hash code of the incoming message is generated. This plus the signature is input to a verification function. The verification function also depends on the global public key as well as the sender's public key, which is paired with the sender's private key. The output of the verification function is a value that is equal to the signature component if the signature is valid. The signature function is such that only the sender, with knowledge of the private key, could have produced the valid signature.

7. **What is the task of firewall? List the elements of X.509.** **[2+3]**

Ans: A "firewall," is a tool that provides a filter of both incoming and outgoing packets. Thus, the main task of the firewall itself is to monitor and control all incoming or outgoing access to network connections based on predetermined security rules.Most firewalls perform two basic security functions:

- Packet filtering based on accepts or deny policy that is itself based on rules of the security policy.
- Application proxy gateways that provide services to the inside users and at the same time protect each individual host from the "bad" outside users.

List of the elements of X.509 are as follows:

Elements	Purpose
1. Versión number	1. Most certificates use X.509 version 3.
2. Serial number	2. Unique number set by a CA
3. Issuer	3. Name of the CA
4. Subject issued certificate	4. Name of a receiver of the certificate
5. Validity period	5. Period in which certificate will valid
6. Public-key algorithm information of the subject of the certificate	6. Algorithm used to sign the certificate with digital signature
7. Digital signature of the issuing authority	7. Digital signature of the certificate signed by CA
8. Public key	8. Public key of the subject
9. Extension	9. Optional Extensions (e.g., Key usage)

8. **How does the nature of worms differ with virus? Define PKI with its architecture model.** **[1+4]**

Ans: The primary difference between a virus and a worm is that viruses must be triggered by the activation of their host; whereas worms are stand-alone malicious programs that can self-replicate and propagate independently as soon as they have breached the system. Worms do not require activationor any human interventionto execute or spread their code.

Public-key infrastructure (PKI) is defined asthe set of hardware, software, people, policies, and procedures needed to create,manage, store, distribute, and revoke digital certificates based on asymmetric cryptography. The principal objective for developing a PKI is to enable secure, convenient, and efficient acquisition of public keys.

The Internet Engineering Task Force(IETF) Public Key Infrastructure X.509 (PKIX) working group has been the driving force behind setting up a formal (and generic) model based on X.509 that issuitable for deploying a certificate-based architecture on the Internet. Figure below shows the interrelationship among the key elements of the PKIXmodel. These elements are

- **End entity:** A generic term used to denote end users, devices (e.g., servers,routers), or any other entity that can be identified in the subject field of a public key certificate. End entities typically consume and/or support PKI-relatedservices.
- **Certification authority (CA):** The issuer of certificates and (usually) certificate revocation lists (CRLs). It may also support a variety of administrativefunctions, although these are often delegated to one or more registrationauthorities.
- **Registration authority (RA):** An optional component that can assume a number of administrative functions from the CA. The RA is often

associated withthe end entity registration process, but can assist in a number of other areas aswell.

- **CRL issuer:** An optional component that a CA can delegate to publish CRLs.
- **Repository:** A generic term used to denote any method for storing certificatesand CRLs so that they can be retrieved by end entities.

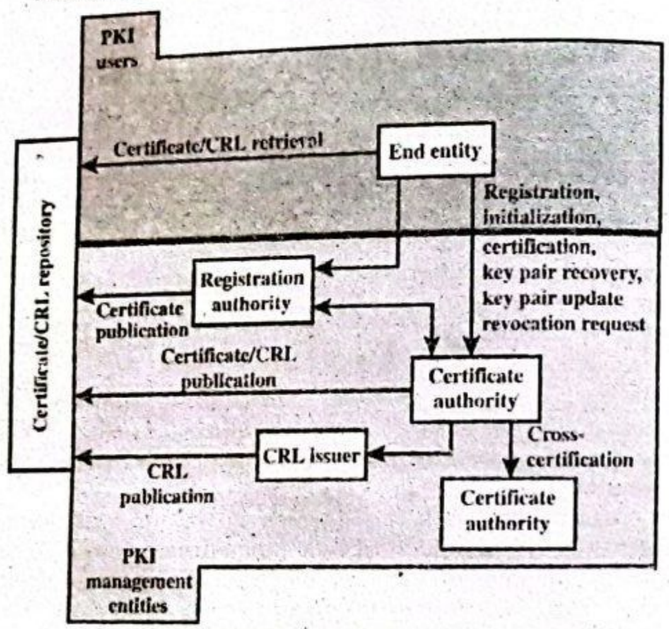


Figure: PKIX architectural model

9. Explain the procedure of mix column transformation in AES with an example. [5]

Ans: This method messes with columns instead of rows. MixColumns transformation takeseach column in the state and perform linear transformation on it. Since eachcolumn has 4 rows, the matrix transformation has to be 4 × 4 and is definedas follow

$$M = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix}$$

And since linear transformation has an inverse, this operation is invertible.In fact, the matrix used to revert the state back to its original standing isgiven by

$$M^{-1} = \begin{bmatrix} 14 & 11 & 13 & 9 \\ 9 & 14 & 11 & 13 \\ 13 & 9 & 14 & 11 \\ 11 & 13 & 9 & 14 \end{bmatrix}$$

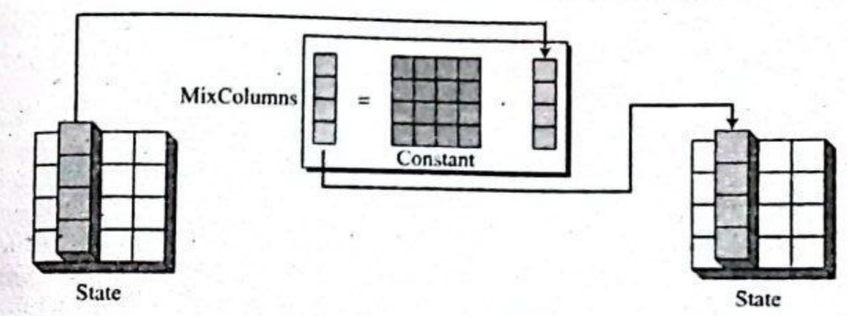
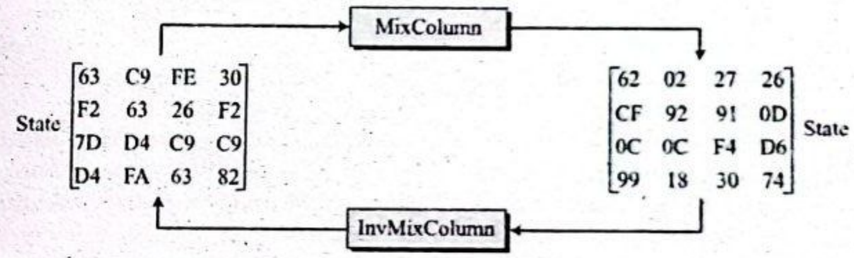


Figure: MixColumns transformation

Figure below shows how a state is transformed using the MixColumns transformation. The figure also shows that the InvMixColumns transformation creates the original one.



10. What is the role of prime number in Euler totient Function? Find the GCD of 12 and 16 using Euclidean algorithm. [2.5+2.5]

Ans: Euler's totient function one may use to know how many prime numbers are coming up to the given integer 'n.' It is also called an arithmetic function. Two things are important for an application or use of Euler's totient function. **One** is that the gcd formed from the given integer 'n' should be multiplicative. The **other** is that the numbers of gcd should be the prime numbers only. The integer 'n' in this case should be more than 1. Calculating the Euler's totient function from a negative integer is impossible. The principle, in this case, is that for $\phi(n)$, the multiplicators called m and n should be greater than 1. Hence, denoted by $1<m<n$ and gcd (m, n) = 1. Sign ϕ is the sign used to denote the totient function.

Euclidean algorithm for GCD

$a = q_0 \times b + r_0$

or, $b = q_1 \times r_0 + r_1$ and so on

Here, a= 16 and b = 12

or, $16 = 1 \times 12 + 4$

or, $12 = 3 \times 4 + 0$

Hence, GCD of 12 and 16 using Euclidean algorithm is 4.

11. Write down any two limitations of MAC? What does policy and mechanism mean in cryptography? Describe with a scenario. [2+3]

Ans: There are two major limitations of MAC, both due to its symmetric nature of operation

- **Establishment of Shared Secret:** It can provide message authentication among pre-decided legitimate users who have shared key.This requires establishment of shared secret prior to use of MAC.
- **Inability to Provide Non-Repudiation:** Non-repudiation is the assurance that a message originator cannot deny any previously sent messages and commitments or actions.MAC technique does not provide a non-repudiation service. If the sender and receiver get involved in a dispute over message origination, MACs cannot provide a proof that a message was indeed sent by the sender.Though no third party can compute the MAC, still sender could deny having sent the message and claim that the receiver forged it, as it is impossible to determine which of the two parties computed the MAC.

Security Policies and Mechanisms

Simply stating that a system should be able to protect itself against all possible security threats is not the way to actually build a secure system. What is first needed is a description of security requirements, that is, a **security policy**. A **security policy** describes precisely which actions the entities in a system are allowed to take and which ones are prohibited. Entities include users, services, data, machines, and so on. Once a security policy has been laid down, it becomes possible to concentrate on the **security mechanisms** by which a policy can be enforced i.e., *security mechanisms implement security policies*. Important security mechanisms are:

1. **Encryption:** It provides a means to implement data confidentiality. In addition, it allows user to verify data modification so, it also provides support for integrity checks.
2. **Authentication:** It is used to verify the claimed identity of a user, client, server, host or other entity are authentic. Typically, users are authenticated by password, but there are many other ways to authenticate clients.
3. **Authorization:** After a client has been authenticated, authorization is to check as a weather the client is authorized to perform specific task.
4. **Auditing:** Auditing tools are used to trace which client accessed what information, when and in which way they did so. Although auditing does not provide any protection against security threats. Audit logs can be useful for the analysis of a security breach, and subsequently taking measures against intruders.

12. Write Short Notes On (Any Two)

a. Classes of Intruder [2.5+2.5]

Ans: An Intruder is a person who attempts to gain unauthorized access to a system, to damage that system, or to disturb data on that system. In summary, this person attempts to violate Security by interfering with system Availability, data Integrity or data Confidentiality.

The three classes of intruders are;

- **Masquerader:** An individual who is not authorized to use the computer and who penetrates a system's access controls to exploit a legitimate user's account. Masquerader is likely to be outsider.
- **Misfeasor:** A legitimate user who accesses data, programs, or resources for which such access is not authorized, or who is authorized for such access but misuses his or her privileges. Misfeasor is normally insider.
- **Clandestine user:** An individual who seizes supervisory control of the system and uses this control to evade auditing and access controls or to suppress audit collection. The clandestine user can be either an outsider or an insider.

b. SSL

Ans: Netscape originated SSL. Version 3 of the protocol was designed with public review and input from industry and was published as an Internet draft document. Subsequently, when a consensus was reached to submit the protocol for Internet standardization, the TLS working group was formed within IETF to develop a common standard. This first published version of TLS can be viewed as essentially an SSLv3.1 and is very close to and backward compatible with SSLv3.

SSL Architecture

SSL is designed to make use of TCP to provide a reliable end-to-end secure service. SSL is not a single protocol but rather two layers of protocols, as illustrated in figure 6.3. The SSL Record Protocol provides basic security services to various higher-layer protocols. In particular, the, Hypertext Transfer Protocol (HTTP), which provides the transfer service for Web client/server interaction, can operate on top of SSL. Three higher-layer protocols are defined as part of SSL: the Handshake Protocol, The Change Cipher Spec Protocol, and the Alert Protocol. These SSL specific protocols are used in the management of SSL exchanges and are examined later in this section.

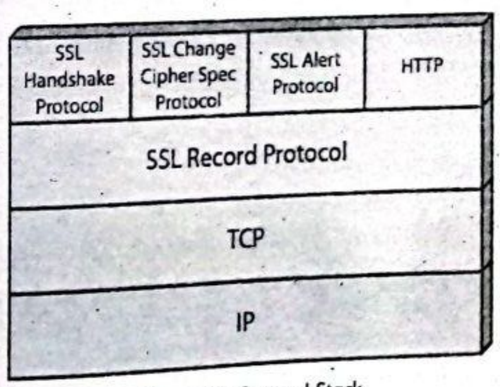


Figure: SSL Protocol Stack

Two important SSL concepts are the SSL session and the SSL connection, which are defined in the specification as follows.

- **Connection:** A connection is a transport (in the OSI layering model definition) that provides a suitable type of service. For SSL, such connections are peer-to-peer relationships. The connections are transient. Every connection is associated with one session.
- **Session:** An SSL session is an association between a client and a server. Sessions are created by the Handshake Protocol. Sessions define a set of cryptographic security parameters which can be shared among multiple connections. Sessions are used to avoid the expensive negotiation of new security parameters for each connection.

c. **DoS Attack**

Ans: A **denial-of-service (DoS)** attack is a type of cyber-attack in which a malicious actor aims to render a computer or other device unavailable to its intended users by interrupting the device's normal functioning. DoS attacks typically function by overwhelming or flooding a targeted machine with requests until normal traffic is unable to be processed, resulting in denial-of-service to addition users. A DoS attack is characterized by using a single computer to launch the attack.

DoS attacks generally take one of two forms. They either flood web services or crash them.

- **Flooding attacks:** Flooding is the more common form DoS attack. It occurs when the attacked system is overwhelmed by large amounts of traffic that the server is unable to handle. The system eventually stops.

  An **ICMP flood** – also known as a ping flood – is a type of DoS attack that sends spoofed packets of information that hit every computer in a targeted network, taking advantage of misconfigured network devices.

  A **SYN flood** is a variation that exploits vulnerability in the TCP connection sequence. This is often referred to as the three-way handshake connection with the host and the server. Here's how it works:

  The targeted server receives a request to begin the handshake. But, in a SYN flood, the handshake is never completed. That leaves the connected port as occupied and unavailable to process further requests. Meanwhile, the cybercriminal continues to send more and more requests, overwhelming all open ports and shutting down the server.
- **Crash attacks:** Crash attacks occur less often, when cybercriminals transmit bugs that exploit flaws in the targeted system. The result? The system crashes.

  Crash attacks – and flooding attacks – prevent legitimate users from accessing online services such as websites, gaming sites, email, and bank accounts.

TU QUESTIONS-ANSWERS 2078

Bachelor Level/Third Year/Fifth Semester/Science — Full Marks: 60
Computer Science and Information Technology [CSC 317] — Pass Marks: 24

1. **Define CIA triad. State the encryption process of double and triple DES. What is the task of S-Box in DES? Discuss with an example. [3+ 2 + 5]**

Ans: The CIA triad concept in information security which guides an organization's efforts towards ensuring data security. CIA stands for **confidentiality, integrity** and **availability** is also known as three foundations of information systems security.

Figure: CIA Triad

- **Confidentiality:** Confidentiality prevents the disclosure of information to unauthorized people, resources and processes. Another term for confidentiality is privacy. Organizations restrict access to ensure that only authorized operators can use data or other network resources. For example, a programmer should not have access to the personal information of all employees.
- **Integrity:** Integrity is the accuracy, consistency, and trustworthiness of data during its entire life cycle. Another term for integrity is quality. Data undergoes a number of operations such as capture, storage, retrieval, update, and transfer. Data must remain unaltered during all of these operations by unauthorized entities.
- **Availability:** Data availability is the principle used to describe the need to maintain availability of information systems and services at all times. Cyber-attacks and system failures can prevent access to information systems and services. For example, interrupting the availability of the website of a competitor by bringing it down may provide an advantage to its rival. These denial-of-service (DoS) attacks threaten system availability and prevent legitimate users from accessing and using information systems when needed.

Encryption in double DES

Given a plaintext and two encryption keys K_1 and K_2, a cipher text can be generated as,

Ciphertext = $E(K_2, (E(K_1, Plaintext))$

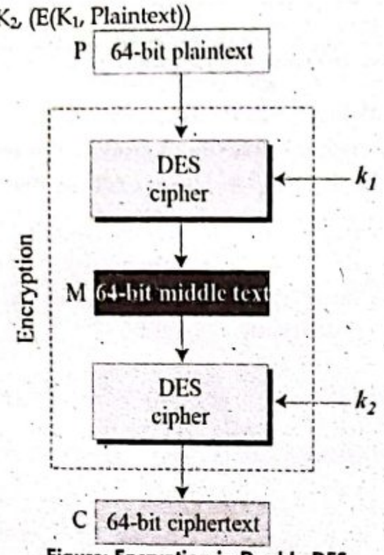


Figure: Encryption in Double DES

Encryption in Triple DES

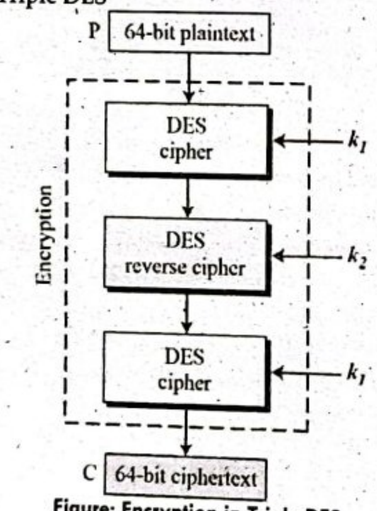


Figure: Encryption in Triple DES

Given a plaintext and two encryption and decryption keys K_1, $K_2$and K_3, a cipher text can be generated as,

Ciphertext = $E(K_3, D(K_2, E(K_1, Plaintext)))$

An **S-box** is a substitution box and it is the only non-linear component in the cipher. Its main purpose is to obscure the relationship between the key, the plaintext, and the ciphertext.The role of the S-boxes in the function F is that the substitution consists of a set of eight S-boxes,each of which accepts 6 bits as input and produces 4 bits as follows:

The first and last bits of the input to box Si form a 2-bit binary number to select one of four substitutions defined by the four rows in the table for Si.The middle four bits select one of the sixteen columns.The decimal value in the cell selected by the row and column is then converted to its 4-bit

representation to produce the output. For example, in S1,for input 011001,the row is 01 and the column is 1100.The value in row 1,column 12 is 9,so the output is 1001.

2. Explain the generic model of digital signature process. Consider the two prime numbers 7 and 19. Select 29 as public key and 41 as private key. Encrypt the plain text 4 and decrypt the cipher text 3 using RSA. [5+5]

Answer:

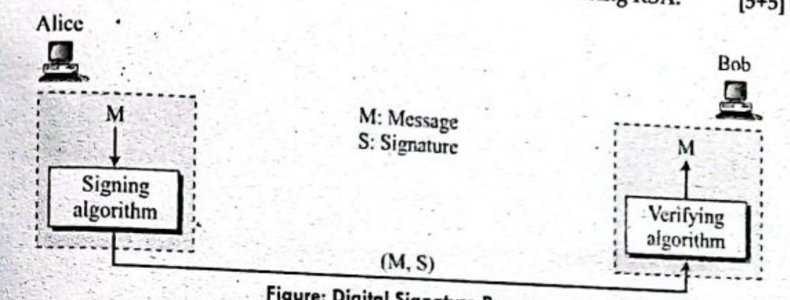


Figure: Digital Signature Process

Above figure shows the digital signature process. The sender uses a **signing algorithm** to sign the message. The sender uses a signing algorithm to sign the message. The message and the signature are sent to the receiver. The receiver receives the message and the signature and applies the verifying algorithm to the combination. If the result is true, the message is accepted otherwise, it is rejected.

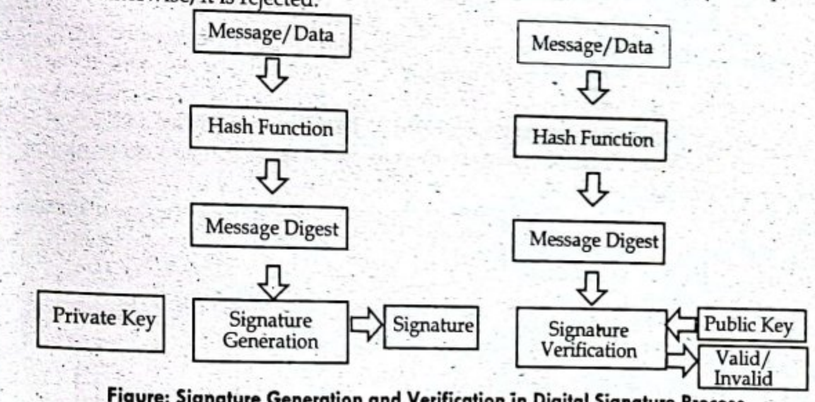


Figure: Signature Generation and Verification in Digital Signature Process

Given,

Two prime numbers i.e., p = 7 and q = 19

Public key (e) ='29

Private key (d) = 41

Now,

$n = p \times q = 7 \times 19 = 133$

then,

Public Key pair = (e, n) = (29, 133)

Private Key pair = (d, n) = (41, 133)

Encrypting the plain text (m) = 4

Cipher text = m^e MOD n

$= 4^{29}$ MOD 133 = 16

Decrypting the cipher text (c) = 3

Plain text $= c^d \text{ MOD } n$

$= 3^{11} \text{ MOD } 133$

$= 110$

3. Define Galois field with an example. Explain any two modes of block cipher encryptions. Determine the quadratic residues of 7. [2+4+4]

Ans: The elements of Galois Field $gf(p^n)$ is defined as

$gf(p^n) = (0, 1, 2, \ldots\ldots, p-1) \cup (p, p+1, p+2, \ldots, p+p-1) \cup$

$(p^2, p^2+1, p^2+2, \ldots\ldots, p^2+p-1) \cup \ldots \cup (p^{n-1}, p^{n-1}+1, p^{n-1}+2, \ldots\ldots, p^{n-1}+p-1)$

where $p \in \mathbb{P}$ and $n \in \mathbb{Z}^+$. The order of the field is given by p^n while p iscalled the characteristic of the field. On the other hand, gf stands for Galois Field. Also note that the degree of polynomialof each element is at most n – 1.

Example: $gf(5) = (0,1,2,3,4)$which consists of 5 elements where each of them is a polynomial of degree 0(a constant) while

$gf(2^3) = (0, 1, 2, 2+1, 2^2, 2^2+1, 2^2+2, 2^2+2+1) = (0, 1, 2, 3, 4, 5, 6, 7)$

which consists of $2^3 = 8$ elements where each of them is a polynomial ofdegree at most 2 evaluated at 2.

Modes of block cipher encryptions are:

- Electronic codebook (ECB) mode
- Cipher block chaining (CBC) mode
- Cipher feedback (CFB) mode
- Output feedback (OFB) mode
- Counter (CTR) mode

Electronic Codebook (ECB) mode

The simplest mode of operation is called the electronic codebook (ECB) mode. Theplaintext is divided into N blocks. The block size is n bits. If the plaintext size is not multiple of the block size, the text is padded to make the last block the same size as theother blocks. The same key is used to encrypt and decrypt each block. Figure below showsthe encryption and decryption in this mode.

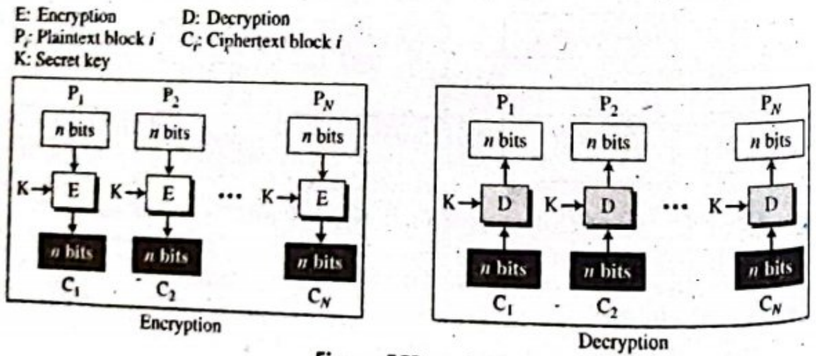


Figure: ECB mode

The relation between plaintext and ciphertext block is shown below:

Encryption: $C_i = E_K (P_i)$

Decryption: $P_i = D_K (C_i)$

Cipher Block Chaining (CBC) Mode

The next evolution in the operation mode is the **cipher block chaining (CBC) mode**.In CBC mode, each plaintext block is exclusive-ored with the previous ciphertextblock before being encrypted. When a block is enciphered, the

block is sent, but acopy of it is kept in memory to be used in the encryption of the next block. The readermay wonder about the initial block. There is no ciphertext block before the first block.In this case, a phony block called the initialization vector (IV) is used.The senderand receiver agree upon a specific predetermined IV. In other words, an IV isused instead of the nonexistent C_0. Figure below shows CBC mode. At the sender side, exclusive-oring is done before encryption; at the receiver site, decryption is done before exclusive-oring.

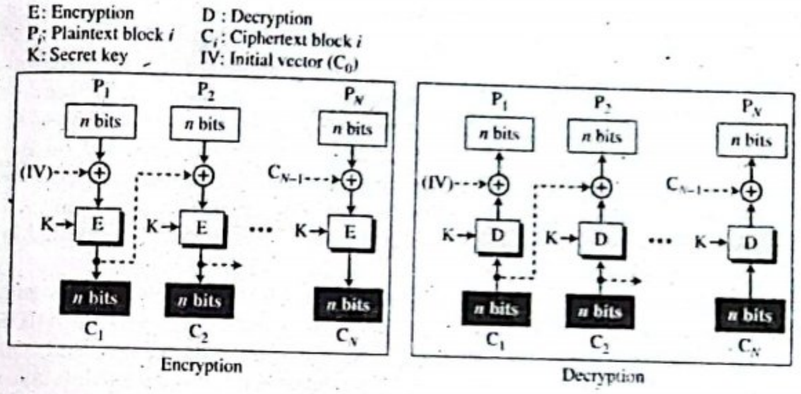


Figure: CBC mode

The relation between plaintext and ciphertext blocks is shown below:

Encryption:

$C_0 = IV$

$C_i = E_K (P_i \oplus C_{i-1})$

Decryption:

$C_0 = IV$

$P_i = D_K (C_i) \oplus C_{i-1}$

For the quadratic residue of 7

Here, $p = 7$

$\phi(7) = 6$

We have,

$y^2 \equiv a \pmod p$

or, $1^2 \equiv 1 \pmod 7$

or, $2^2 \equiv 4 \pmod 7$

or, $3^2 = 9 \equiv 2 \pmod 7$

or, $4^2 = 16 \equiv 2 \pmod 7$

or, $5^2 = 25 \equiv 4 \pmod 7$

or, $6^2 = 36 \equiv 1 \pmod 7$

Hence, 1, 2, 4 are the quadratic residue of 7.

SectionB

Short Answer Questions

Attempt any EIGHT questions. [8×5=40]

4. What does intrusion mean? How the system detect intrusion? List any four types of firewalls. [1+2+2]

Ans: Intrusion means any set of actions that attempts to compromise the integrity, confidentiality or availability of a resource.

An Intrusion Detection System (IDS) is a system that monitors network traffic for suspicious activity and issues alerts when such activity is discovered. It is a software application that scans a network or a system for harmful activity or policy breaching. Any malicious venture or violation is normally reported either to an administrator or collected centrally using a security information and event management (SIEM) system. A SIEM system integrates outputs from multiple sources and uses alarm filtering techniques to differentiate malicious activity from false alarms.

Intrusion detection systems come in different flavors and detect suspicious activities using different methods, including the following:

- **Network Intrusion Detection System (NIDS):** Network intrusion detection systems (NIDS) are set up at a planned point within the network to examine traffic from all devices on the network. It performs an observation of passing traffic on the entire subnet and matches the traffic that is passed on the subnets to the collection of known attacks. Once an attack is identified or abnormal behavior is observed, the alert can be sent to the administrator. An example of a NIDS is installing it on the subnet where firewalls are located in order to see if someone is trying to crack the firewall.
- **Host Intrusion Detection System (HIDS):** Host intrusion detection systems (HIDS) run on independent hosts or devices on the network. A HIDS monitors the incoming and outgoing packets from the device only and will alert the administrator if suspicious or malicious activity is detected. It takes a snapshot of existing system files and compares it with the previous snapshot. If the analytical system files were edited or deleted, an alert is sent to the administrator to investigate. An example of HIDS usage can be seen on mission critical machines, which are not expected to change their layout.

A **firewall** may act as a packet filter. It can operate as a positive filter, allowing to pass only packets that meet specific criteria, or as a negative filter, rejecting any packet that meets certain criteria. Depending on the type of firewall, it may examine one or more protocol headers in each packet, the payload of each packet, or the pattern generated by a sequence of packets. The principal types of firewalls and they are:

1. Packet Filtering Firewalls
2. Circuit Level Gateway Firewalls
3. Application-Level Gateway Firewalls
4. Stateful Multilayer Inspection Firewalls

5. Decrypt the message "GVPJ" using Hill cipher taking the key as key $\begin{pmatrix} 3 & 7 \\ 5 & 12 \end{pmatrix}$. [5]

Ans: Key $\begin{pmatrix} 3 & 7 \\ 5 & 12 \end{pmatrix}$

Step 1: Find multiplicative inverse of determinant of key matrix $\begin{pmatrix} 3 & 7 \\ 5 & 12 \end{pmatrix}$

$= (3 \times 12 - 7 \times 5) \bmod 26 = 1 \bmod 26$

Now, multiplicative inverse of 1 mod 26 is

$1 \times a = 1 \bmod 26$

$\therefore \quad a = 1$

Step 2: find the adj. matrix of Key matrix $\begin{pmatrix} 3 & 7 \\ 5 & 12 \end{pmatrix}$

$$= \begin{pmatrix} 12 & -7 \\ -5 & 3 \end{pmatrix} \bmod 26 = \begin{pmatrix} 3 & 7 \\ 5 & 12 \end{pmatrix}$$

Step 3: Multiply multiplicative inverse of determinant by adj. matrix

$$= 1 \times \begin{pmatrix} 12 & -7 \\ -5 & 3 \end{pmatrix} = \begin{pmatrix} 12 & 19 \\ 21 & 3 \end{pmatrix} \bmod 26$$

Now, cipher text = "GVPJ"

$$\begin{pmatrix} F \\ V \end{pmatrix} = \begin{pmatrix} 6 \\ 21 \end{pmatrix} \text{ and } \begin{pmatrix} P \\ J \end{pmatrix} = \begin{pmatrix} 15 \\ 9 \end{pmatrix}$$

Step 4: multiply cipher text (c) by inverse key matrix

$$= \begin{pmatrix} 12 & 19 \\ 21 & 3 \end{pmatrix} \begin{pmatrix} 6 \\ 21 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 72 + 399 \\ 126 + 63 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 471 \\ 189 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 3 \\ 7 \end{pmatrix} = \begin{pmatrix} D \\ H \end{pmatrix}$$

Similarly,

$$\begin{pmatrix} 12 & 19 \\ 21 & 3 \end{pmatrix} \begin{pmatrix} 15 \\ 9 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 180 + 171 \\ 315 + 27 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 351 \\ 342 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 13 \\ 4 \end{pmatrix} = \begin{pmatrix} N \\ E \end{pmatrix}$$

$\therefore$ Plain Text (P) = DHNE

6. Describe the PKI trust model. [5]

Ans: To help ensure trust, a PKI relies on a standard trust model that assigns to a third party the responsibility of establishing a trust relationship between any two communicating entities. The model used by a PKI is a strict hierarchical model. At the top is a publicly (or privately) recognized source (authority) that everyone using the PKI recognizes and trusts to validate (authorize and certify) the identities that are part of the PKI. Under this authority might exist subordinate authorities that rely on the top (root) authority as the ultimate source of authorization and certification.

The mechanism that is typically used to convey or validate this authorized identity is the digital certificate. The authority that is entrusted with issuing digital certificates for the purpose of authorizing and validating identity is a Certification Authority (CA, also often referred to as a Certificate Authority). Again, CAs can be organized in a hierarchy of authority with the ultimate

authority at the top being the root CA of that CA hierarchy. The strength of a CA rests entirely on the agreement between the holder of an identity that is authorized by the CA on one side and those who communicate with the holder of that identity on the other side to trust the integrity of the CAs authorization of that identity. Among other requirements, the most important is that this identity must be unique to the identity holder, and all parties involved must trust the CA to guarantee this to the extent possible.

7. **Define authentication system. Illustrate the need of mutual authentication over one way authentication with an example.** **[2+3]**

Ans: Authentication system is to guarantee that an entity attempting to access protected resources in genuine. In concrete terms, this consists of preventing two main types of attaches, both of which may have serious consequences:

- fraudulent access to a system, allowing access to sensitive data;
- identity theft, which may result in an innocent individual being considered to be responsible for the actions of the attacker.

Mutual authentication is when two sides of a communications channel verify each other's identity, instead of only one side verifying the other. Mutual authentication is also known as "two-way authentication" because the process goes in both directions.

Mutual authentication is used in verification schemes that transmit sensitive data, in order to ensure data security& can be accomplished with two types of credentials: usernames &passwords, and public key certificates.

Mutual authentication verifies both parties in a digital communications channel. For example, a client and a server using mutual authentication take steps to independently verify each other's identity, instead of only the client authenticating the server. Device-to-device connections, like those between Internet of Things (IoT) devices, often use mutual authentication as well.

One-way authentication happens all the time on the Internet. Every time someone loads a website that uses HTTPS, their device authenticates the identity of the web server by checking the server's TLS certificate. Another example would be a person signing in to their account on an application, in this case, the application is authenticating the person, there is a higher vulnerability to hackers because the password is human-made rather than a computer-generated certificate. Where mutual authentication help to stop different types of attacks as Replay attack, **Spoofing and impersonation attacks,Man in the middle attack.**

Example:

In one-way authentication, only one party verifies the identity of the other party.

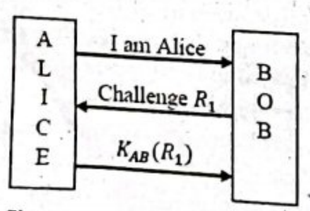


Figure: one-way authentication

Assume that Alice and Bob share a secret key K_{AB}

In mutual authentication, both communicating parties verify each other's identity.

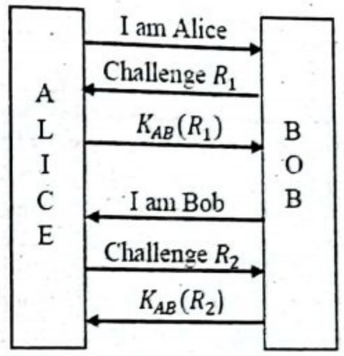


Figure: Mutual authentication

8. **Find the value of 7^{2019} MOD 13using Fermat's Little theorem. Define Euler totient function with an example.** **[2+3]**

Ans: Now,

7^{2019} MOD 13

From Fermat's Little theorem,

$7^{13-1} \equiv 1 \pmod{13}$

or, $7^{12} \equiv 1 \pmod{13}$

and, $7^{13} \equiv 7 \pmod{13}$

Also,

$= 7^{2019}$

$= (7^{12})^{168} \times 7^3 (\text{MOD } 13)$

$= (1)^{168} \times 7^3 (\text{MOD } 13)$

$= 7^3 (\text{MOD } 13)$

$= 343 (\text{MOD } 13)$

$= 5 (\text{MOD } 13)$

Euler's Totient function $\phi(n)$ for an input n is count of positive integers in {1, 2, 3, ..., n-1} that are relatively prime ton, i.e., the positive integers whose GCD with n is 1.

Examples:

$\phi(1) = 1$;gcd(1, 1) is 1

$\phi(2) = 1$; gcd(1, 2) is 1, but gcd(2, 2) is 2.

$\phi(3) = 2$; gcd(1, 3) is 1 and gcd(2, 3) is 1

$\phi(4) = 2$; gcd(1, 4) is 1 and gcd(3, 4) is 1

$\phi(5) = 4$; gcd(1, 5) is 1, gcd(2, 5) is 1, gcd(3, 5) is 1 and gcd(4, 5) is 1

9. **List the properties of hash function. Discuss the first pass of MD4.** **[2+3]**

Ans: **Properties of Cryptographic Hash Functions**

The hash computation on a message m is mathematically represented by $H(m) = y$. The computation of y from m must be easy and fast. The fundamental security properties of H are defined below:

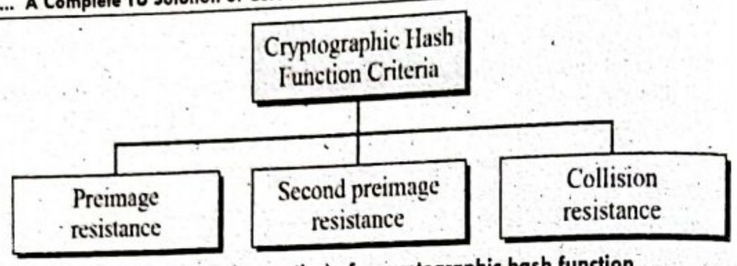


Figure: Criteria (properties) of a cryptographic hash function

- **Preimage resistance:** H is preimage resistant if for any given hash value y of H, it is "computationally infeasible" to find a message m such that $H(m) = y$. That is, it must be hard to invert H from y to get an m corresponding to y. This property is also called one-wayness. For an ideal H, it takes about 2^n evaluations of H to find a preimage.
- **Second pre-image resistance:** His second preimage resistant if for any given message m, it is "computationally infeasible" to find another message m^* such that $m^* = m$ and $H(m) = H(m^*)$. For an ideal H, it takes about 2^n evaluations of H to find a second preimage.
- **Collision resistance:** H is collision resistant if it is "computationally infeasible" to find any two messages m and m^* such that $m \neq m^*$ and $H(m) = H(m^*)$. Due to the birthday paradox, for an ideal H, it takes about $2^{n/2}$ evaluations of H to find a collision.

MD4 Message Digest Pass 1:

A function $F(x, y, z) = (x \wedge y) \vee (\sim x \wedge z)$; takes three 32-bit words x, y, and z, and produces an output 32-bit word. This function is sometimes known as the **selection function**, because if the n^{th} bit of x is a 1 it selects the n^{th} bit of y for the n^{th} bit of the output. Otherwise (if the n^{th} bit of x is a 0) it selects the n^{th} bit of z for the n^{th} bit of the output.

Each of the 16 words of the messages is separately processed using the following relation, where i move from 0 to 15.

$d_{(-i)\wedge 3} = (d_{(-i)\wedge 3} + F(d_{(1-i)\wedge 3}, d_{(2-i)\wedge 3}, d_{(3-i)\wedge 3}) + m_i) \lfloor S_1(i \wedge 3)$

Where, $S_1(i) = 3 + 4i$, so ⌊s cycle over the values 3, 7, 11, 15.

The "$\wedge$ 3" that appears several times in the above equation means that only the bottom two bits are used (because we're doing a *bitwise* and with 11_2). So $i \wedge 3$ cycles 0, 1, 2, 3, 0, 1, 2, 3, ... while $(-i) \wedge 3$ cycles 0, 3, 2, 1, 0, 3, 2, 1,... and $(1-i) \wedge 3$ cycles 1, 0, 3, 2, 1, 0, 3, 2,.... We can write out the first few steps of the pass as follows:

$d_0 = (d_0 + F(d_1, d_2, d_3) + m_0) \lfloor 3;$

$d_3 = (d_3 + F(d_0, d_1, d_2) + m_1) \lfloor 7;$

$d_2 = (d_1 + F(d_3, d_0, d_1) + m_2) \lfloor 11;$

$d_1 = (d_1 + F(d_2, d_3, d_1) + m_3) \lfloor 15;$

$d_0 = (d_0 + F(d_1, d_2, d_3) + m_4) \lfloor 3;$ and so on.

10. Differentiate between Symmetric and Asymmetric cipher. Encrypt the message "HELL" using the key "FAIL" usingVernam Cipher. [3+2]

Ans:

	Symmetric Encryption		Asymmetric Encryption
1.	Smaller cipher text compares to original plain text file.	1.	Larger cipher text compares to original plain text file.
2.	Used to transmit big data.	2.	Used to transmit small data.
3.	Symmetric key encryption works on low usage of resources.	3.	Asymmetric encryption requires high consumption of resources.
4.	128 or 256-bit key size.	4.	RSA 2048-bit or higher key size.
5.	Less secured due to use a single key for encryption.	5.	Much safer as two keys are involved in encryption and decryption.
6.	Symmetric Encryption uses a single key for encryption and decryption.	6.	Asymmetric Encryption uses two keys for encryption and decryption
7.	It is an old technique.	7.	It is a modern encryption technique.
8.	A single key for encryption and decryption has chances of key compromised.	8.	Two keys separately made for encryption and decryption that removes the need to share a key.
9.	Symmetric encryption is fast technique	9.	Asymmetric encryption is slower in terms of speed.
10.	RC4, AES, DES, 3DES, and QUAD.	10.	RSA, Diffie-Hellman, ECC algorithms.

Encrypt the message using Vernam Cipher.

Now,

Plaintext = HELL

Key = FAIL

Assign numbers, A= 0, B=1, C=2 ....., Z=25

	7	4	11	11
+	5	0	8	11
	12	4	19	22

Ciphertext= METW

11. Divide $3x^2 + 4x + 3$ by $5x + 6$ over GF(7) [5]

Answer:

$5x + 6 \;\big|\; 3x^2 + 4x + 3 \;\big|\; 2x + 4$

$3x^2 + 5x$

$- \quad -$

$-x + 3$

$\equiv 6x + 3 \pmod 7$

$20x + 24$

$- \quad -$

$-14x + 14$

$\equiv 0 \pmod 7$

$Q = 2x + 4$

$R = 0$ Ans

12. Define SSL protocol. Mention the services provided by PGP. [2+3]

Ans: Secure Socket Layer (SSL), is an encryption-based internet security protocol. It was first developed by Netscape in 1995 for the purpose of ensuring

privacy, authentication and data integrity in internet communications. SSL is the predecessor to the modern TLS encryption used today.

The SSL Record Protocol provides two services for SSL connections:

- **Confidentiality:** The Handshake Protocol defines a shared secret key that is used for conventional encryption of SSL payloads.
- **Message Integrity:** The Handshake Protocol also defines a shared secret key that is used to form a message authentication code (MAC).

The actual operation of PGP is based on five services: **authentication, confidentiality, compression, e-mail compatibility, and segmentation.**

- **Authentication:** PGP provides authentication via a digital signature scheme. The hash code (MAC) is created using a combination of SHA-1 and RSA to provide an effective digital signature. It can also create an alternative signature using DSS and SHA-1. The signatures are then attached to the message or file before sending. PGP, in addition, supports unattached digital signatures. In this case, the signature may be sent separately from the message.
- **Confidentiality:** PGP provides confidentiality by encrypting messages before transmission. PGP encrypts messages for transmission and storage using conventional encryption schemes such as CAST-128, IDEA, and 3DES. In each case, a 64-bit cipher feedback mode is used. As in all cases of encryption, there is always a problem of key distribution; So, PGP uses a conventional key once.
- **Compression:** PGP compresses the message after applying the signature and before encryption. The idea is to save space.
- **E-mail Compatibility:** As we have seen above, PGP encrypts a message together with the signature (if not sent separately) resulting into a stream of arbitrary 8-bit octets. But since many e-mail systems permit only use of blocks consisting of ASCII text, PGP accommodates this by converting the raw 8-bit binary streams into streams of printable ASCII characters using a radix-64 conversion scheme. On receipt, the block is converted back from radix-64 format to binary. If the message is encrypted, then a session key is recovered and used to decrypt the message. The result is then decompressed. If there is a signature, it has to be recovered by recovering the transmitted hash code and comparing it to the receiver's calculated hash before acceptance.
- **Segmentation:** To accommodate e-mail size restrictions, PGP automatically segments email messages that are too long. However, the segmentation is done after all the housekeeping is done on the message, just before transmitting it. So, the session key and signature appear only once at the beginning of the first segment transmitted. At receipt, the receiving PGP strip off all e-mail headers and re-assembles the original mail.

MODEL QUESTIONS SETS FOR PRACTICE

MODEL SET 1

Course Title: Cryptography **Full Marks: 60**
Course No: CSC316 **Pass Marks: 24**
Semester: V **Time: 3 hrs**

Candidates are required to give their answers in their own words as far as practicable.

Group A

Attempt any TWO questions: **(2x10=20)**

1. Let's go back to the first step of processing in each round of AES. How does one look up the 16 × 16 s-box table for the byte-by-byte substitution? Generate the public and private key pair of RSA algorithm. Use two prime numbers are p= 7 and q= 19. [5+5]
2. Differentiate Kerberos version 4 and Kerberos version 5. Explain Kerberos version 4operations. [4+6]
3. Discuss the five principle services provided by PGP protocol. Define IPSec. Also, explain Transport and Tunnel Modes. [5+1 +4]

Section B

Short Answer Questions

Attempt any EIGHT questions. **[8×5=40]**

4. What are the classes of Message Authentication Functions? Explain the properties of Cryptographic Hash Functions. [2+3]
5. Explain SSL Handshake Protocol? [5]
6. Encrypt the message "help" using the Hill cipher with the key $\begin{pmatrix} 3 & 3 \\ 2 & 5 \end{pmatrix}$. Show your calculations and the result. [5]
7. Express gcd(252,198)=18 as a linear combination of 252 and 198. [5]
8. Define Computer Worm along with its types. [5]
9. Describe PKIX Management Protocols. Differentiate between PKI and Kerberos. [2+3]
10. Explain the Digital Signature Algorithm? [5]
11. Miller-Rabin test says that if a candidate integer n is prime, it must satisfy one of two special conditions. What are those two conditions? [5]
12. Write Short Notes On (Any Two) [2.5+2.5]
    a. Network Security
    b. Demilitarized Zone (DMZ) Networks
    c. Intruders Detection Approaches

MODEL SET 2

Group A

Attempt any TWO questions: **(2x10=20)**

1. Mention the families SHA-2? Describe how 160-bit of hash value is generated by taking an input message of Variable size using SHA-1? [2+8]
2. Discuss how encryption and decryption is done using RSA? In a RSA system, a user Named Ram has chosen the prime 5 and 7 to create a key pair. The public key is (e_{Ram}, n) and the private key is (d_{Ram}, n). Compute the private and public key pairs. Suppose another user Sita knows public key of Ram and want to send the plaintext "hi" to Ram using RSA Scheme. Show how Sita has encrypted the plaintext and Ram has decrypted the ciphertext. [5+5]
3. Describe the working principle of Feistel Cipher Structure. Give the encryption and decryption procedure for 2-DES and 3-DES. Find the multiplicative inverse of 7 in Z_{11} using Extended Euclidean Algorithm. [4+2+4]

Section B

Short Answer Questions

Attempt any EIGHT questions. **[8×5=40]**

4. Define authentication system. How challenge response systems can be used as authentication approach? [1+4]
5. Define SSL. How SSL Record Protocol Provides Security in Secure Socket Layer Protocol? [1+4]
6. Decrypt the message "CMAL" using the Hill cipher with the key $\begin{pmatrix} 7 & 8 \\ 11 & 11 \end{pmatrix}$. Show your calculations and the result. [5]
7. Divide $5x^2 + 4x + 6$ by $2x + 1$ over GF(7). [5]
8. Differentiate between virus, worm and trojan horse. [5]
9. Describe the purpose of PKI trust model? List any four types of firewall. [3+2]
10. What is digital signature? How DSS Approach is used to generate digital signature? [1+4]
11. Define Euler totient function. Find out whether 3 is primitive root of 7? [1+4]
12. Write Short Notes On (Any Two) [2.5+2.5]
    a. Vernam Cipher
    b. Kerberos Protocol
    c. Intrusion Detection System

MODEL SET 3

Section A

Long Answer Questions

Attempt any TWO questions. **[2×10=20]**

1. What are the characteristics of a stream cipher? Explain the key elements of public key encryption. [2+8]
2. Explain SHA – 512 algorithm with a neat diagram. [10]
3. What do you mean by one-time password? Explain Diffie-Helman Key agreement. [2+8]

Section B

Short Answer Questions

Attempt any EIGHT questions. **[8×5=40]**

4. Explain symmetric key encryption model with a neat diagram.
5. Explain Euclid's algorithm with example.
6. Explain transpositional Cipher with an example.
7. Explain CBC mode of operation.
8. Compare SSL and TLS protocols
9. Differentiate equality and congruence with examples.
10. Explain Fermat's theorem of primality test.
11. What is trapdoor one-way function? Write any three applications of RSA algorithm. [2+3]
12. Write Short Notes On (Any Two) [2.5+2.5]
    a. Cryptosystem
    b. Dictionary Attack
    c. Denial of Service Attack

MODEL SET 4

Section A

Long Answer Questions

Attempt any TWO questions. **[2×10=20]**

1. Explain steps in DES algorithm. [10]
2. State and explain Chinese Remainder Theorem with an example. [10]

3. What do you mean by Fabrication? Illustrate ElGamal Encryption and decryption algorithm. [2+8]

Section B

Short Answer Questions

Attempt any EIGHT questions. **[8×5=40]**

4. Discuss Extended Euclidean Algorithm.
5. Explain biometric system along with its types.
6. Explain Caesar Cipher with an example.
7. Discuss the classification of security goals.
8. Find GCD (2740, 1760) using Euclidean Algorithm.
9. Differentiate between block cipher and a stream cipher
10. What is primality test? Explain in brief.
11. Explain various security mechanisms.
12. Write Short Notes On (Any Two) [2.5+2.5]
    a. IPSec
    b. Passive Attack
    c. Man-In-Middle Attack

MODEL SET 5

Section A

Long Answer Questions

Attempt any TWO questions. **[2×10=20]**

1. Construct a playfair matrix with the key "KEYWORD" then use the matrix to encrypt the message "WHYDON'TYOU". What is the purpose of s-box in DES and Prove that DES satisfies complementation property? [4+6]
2. Which four tasks are performed in each round of AES cipher? Explain. [10]
3. What do you mean by avalanche effect? Using RSA algorithm, Find n, d if p=11, q=3, e=3. Encrypt "cipher"Message. [2+8]

Section B

Short Answer Questions

Attempt any EIGHT questions. **[8×5=40]**

4. What is replay attack? What is the counter measure for it?
5. Explain the key expansion process in AES.
6. What are the properties of discrete logarithm?
7. Differentiate between tunnel mode and transport mode of IPSec.
8. Discuss five principle services provided by PGP protocol.
9. How digital signatures can be enforced using encryptions? Illustrate with an example.
10. Determine whether the integers 105 and 295 are relatively prime. Explain your answer using Euclidean algorithm.

11. What is a certificate and why are certificates needed in public key cryptography?
12. Write Short Notes On (Any Two) [2.5+2.5]
    a. Security Mechanism
    b. Multifactor Authentication
    c. Prime Numbers

MODEL SET 6

Section A

Long Answer Questions

Attempt any TWO questions. **[2×10=20]**

1. Give a detailed description of key generation and encryption of IDEA algorithm. [10]
2. Describe signing and verification in digital signature algorithm. [10]
3. Give an example for a situation that compromise in confidentiality leads to compromise in integrity.Write about the usage of session keys, public and private keys in PGP. [2+8]

Section B

Short Answer Questions

Attempt any EIGHT questions. **[8×5=40]**

4. What is a Cyber Threat? Write about Most Common Sources of CyberThreats in detail.
5. Mention the strengths and weakness of DES algorithm.
6. Configure a Vigenere table for the characters from A-H. Use the table to encrypt the text DAD CAFÉ EACH BABE using the key FADE.
7. Perform decryption and encryption using RSA algorithm with p=3, q=11, e=7 and N=5.
8. Discuss any two Substitution Technique and list their merits and demerits.
9. Differentiate between Active attacks and Passive Attacks.
10. How hash function is differ from MAC? Discuss how data integrity can be achieved from either of them?
11. What do you mean by man-in-middle attack? Is man-in-middle attack possible in Diffie-Hellman? How?
12. Write Short Notes On (Any Two) [2.5+2.5]
    a. Zombies
    b. Firewall
    c. Rail-Fence Cipher

MODEL SET 7

Section A
Long Answer Questions

Attempt any TWO questions. [2×10=20]

1. Give a detailed description of key generation and encryption of IDEA algorithm. [10]
2. Describe signing and verification in digital signature algorithm. [10]
3. Give an example for a situation that compromise in confidentiality leads to compromise in integrity.Write about the usage of session keys, public and private keys in PGP. [2+8]

Section B
Short Answer Questions

Attempt any EIGHT questions. [8×5=40]

4. What is a Cyber Threat? Write about Most Common Sources of CyberThreats in detail.
5. Mention the strengths and weakness of DES algorithm.
6. Configure a Vigenere table for the characters from A-H. Use the table to encrypt the text DAD CAFÉ EACH BABE using the key FADE.
7. Perform decryption and encryption using RSA algorithm with p=3, q=11, e=7 and N=5.
8. Discuss any two Substitution Technique and list their merits and demerits.
9. Differentiate between Active attacks and Passive Attacks.
10. How hash function is differ from MAC? Discuss how data integrity can be achieved from either of them?
11. What do you mean by man-in-middle attack? Is man-in-middle attack possible in Diffie-Hellman? How?
12. Write Short Notes On (Any Two) [2.5+2.5]
    a. Zombies
    b. Firewall
    c. Rail-Fence Cipher

MODEL SET 8

Section A
Long Answer Questions

Attempt any TWO questions. [2×10=20]

1. What sort of malware is known as computer virus? Briefly explain encapsulating IPSec Payload? [2+8]
2. Demonstrate DSA digital signature algorithm, with the help of prime divisor q=11 and prime modulus p=23. [10]

3. What do you mean by cryptographic hash function? Discuss Needham-Schroeder Protocol in detail. [2+8]

Section B
Short Answer Questions

Attempt any EIGHT questions. [8×5=40]

4. Write and explain the digital signature algorithm.
5. Explain Extended Euclid's algorithm with example.
6. Explain the methods used for statistical anomaly detection.
7. What are the services provided by IPSec? Where can be the IPSec located on anetwork?
8. Compare stream cipher with block cipher with an example.
9. Calculate the result of the following if the polynomial are over GF(2)
   a. $(x^4 + x^2 + x + 1) + (x^3 + 1)$
   b. $(x^4 + x^2 + x + 1) - (x^3 + 1)$
   c. $(x^4 + x^2 + x + 1) \times (x^3 + 1)$
   d. $(x^4 + x^2 + x + 1)/(x^3 + 1)$
10. Differentiate between SSL Session and SSL Connection. How SSL Record protocol provides confidentiality and message integrity.
11. Briefly describe about Mix Columns and Add Round Key stage in AES.
12. Write Short Notes On (Any Two) [2.5+2.5]
    a. Cryptanalysis
    b. Encryption and Decryption
    c. Diffusion and Confusion.

❑❑❑

New Syllabus

Course Title: Simulation and Modeling
Full Marks: 60 + 20 + 20
Pass Marks: 24 + 8 + 8
Course No: CSC317
Credit Hrs: 3
Nature of the Course: Theory + Lab
Semester: V Course

Description: The syllabus consists of introduction to system, modeling and simulation of different types of systems. It includes the modeling of systems, its validation, verification and analysis of simulation output. It comprises the concept of queuing theory, random number generation as well as study of some simulation languages.

Course Objective: To make students understand the concept of simulation and modeling of real time systems.

Course Contents:

Unit 1: Introduction to Simulation **(6 Hours)**

System and System Environment, Components of System, Discrete and Continuous System, System Simulation, Model of a System, Types of Model, Use of Differential and Partial differential equations in Modeling, Advantages, Disadvantages and Limitations of Simulation, Application Areas, Phases in Simulation Study

Unit 2: Simulation of Continuous and Discrete System **(7 Hours)**

Continuous System Models, Analog Computer, Analog Methods, Hybrid Simulation, Digital-Analog Simulators, Feedback Systems, Discrete Event Simulation, Representation of time, Simulation Clock and Time Management, Models of Arrival Processes - Poisson Processes, Non-stationary Poisson Processes, Batch Arrivals; Gathering statistics, Probability and Monte Carlo Simulation

Unit 3: Queuing System **(6 Hours)**

Characteristics and Structure of Basic Queuing System, Models of Queuing System, Queuing notation, Single server and Multiple server Queueing Systems, Measurement of Queueing System Performance, Elementary idea about networks of Queuing with particular emphasis to computer system, Applications of queuing system

Unit 4: Markov Chains **(2 Hours)**

Features, Process Examples, Applications

Unit 5: Random Numbers (7 Hours)

Random Numbers and its properties, Pseudo Random Numbers, Methods of generation of Random Number, Tests for Randomness, Uniformity and independence, Random Variate Generation

Unit 6: Verification and Validation **(4 Hours)**

Design of Simulation Models, Verification of Simulation Models, Calibration and Validation of the models, Three-Step Approach for Validation of Simulation Models, Accreditation of Models

Unit 7: Analysis of Simulation Output **(4 Hours)**

Confidence Intervals and Hypothesis Testing, Estimation Methods, Simulation run statistics, Replication of runs, Elimination of initial bias 57

Unit 8: Simulation of Computer Systems **(9 Hours)**

Simulation Tools, Simulation Languages: GPSS, Case Studies of different types of Simulation Models and Construction of sample mathematical models **Laboratory Work:** Practical should include the simulation of some real time systems (continuous and discrete event systems), Queuing Systems, Random Number generations as well as study of Simulation Tools and Language

Text Book:

1. Jerry Banks, John S. Carson, Barry L. Nelson, David M. Nicole, "Discrete Event system simulation", 5th Edition, Pearson Education

Reference Books:

1. Geoffrey Gordon: System Simulation
2. Law, "Simulation Modeling and Analysis", 5th Edition, McGraw-Hill

TRIBHUVAN UNIVERSITY

Institution of Science and Technology

Bachelor Level/Third Year/Fifth Semester/Science — Full Marks: 60
Computer Science and Information Technology [CSC 317] — Pass Marks: 24
Time: 3 hrs.

TU QUESTIONS-ANSWERS 2076

Bachelor Level/Third Year/Fifth Semester/Science — Full Marks: 60
Course Title: Simulation and Modeling — Pass Marks: 24
Course Code: CSC 317 — Time: 3 hrs.

Candidates are required to give their answers in their own words as far as practicable. The figures in the margin indicate full marks.

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Define queuing system. Explain different queuing disciplines. Also explain different performance measures for evaluation of queuing system.

Ans: Queuing systems are simplified mathematical models to explain congestion. Broadly speaking, a queuing system occurs any time 'customers' demand 'service' from some facility; usually both the arrival of the customers and the service times are assumed to be random. If all of the 'servers' are busy when new customers arrive, these will generally wait in line for the next available server. Simple queuing systems are defined by specifying the following

(a) the arrival pattern,
(b) the service mechanism, and
(c) queue discipline.

The queue discipline indicates the order in which members of the queue are selected for service. It is most frequently assumed that the customers are served on a first come first serve basis. This is commonly referred to as FIFO (first in, first out) system. Occasionally, a certain group of customers receive priority in service over others even if they arrive late. This is commonly referred to as priority queue. The queue discipline does not always take into account the order of arrival. The server chooses one of the customers to offer service at random. Such a system is known as service in random order (SIRO).

Different queue discipline are listed below:

- FIFO: first-in-first-out
- LIFO: last-in-first-out
- SIRO: service in random order
- SPT: shortest processing time first
- PR: service according to priority
- RR: round robin

The primary long-run measures of performance of queueing systems are the long-run-average number of customers in the system (L) and in the queue (L_Q), the long-run average time spent in system (w) and in the queue (w_Q) per customer utilization, or proportion of time that a server in busy (ρ). the

term "system" usually refers to the waiting line plus the service mechanism, but in general, can refer to any subsystem of the queueing of the queueing system; whereas the term "queue" refers to the waiting line alone. Other measures of performance of interest include the long-run proportion of customers who are delayed in queue longer than t_0 time units, the long-run proportion of customers turned away because of capacity constraints, and the long-run proportion of time the waiting line contains more than k_0 customers.

This section defines the major measures of performance for a general G/G/c/N/K queueing system, discusses their relationships, and shows how they can be estimated from a simulation run. There are two types of estimators: an ordinary sample average, and a time-integrated (or time-weighted) sample average.

Time-Average Number in System L

Consider a queuing system over a period of time T, and L(t) denote the number of customers in the system at time t. A simulation of such a system is shown in figure below.

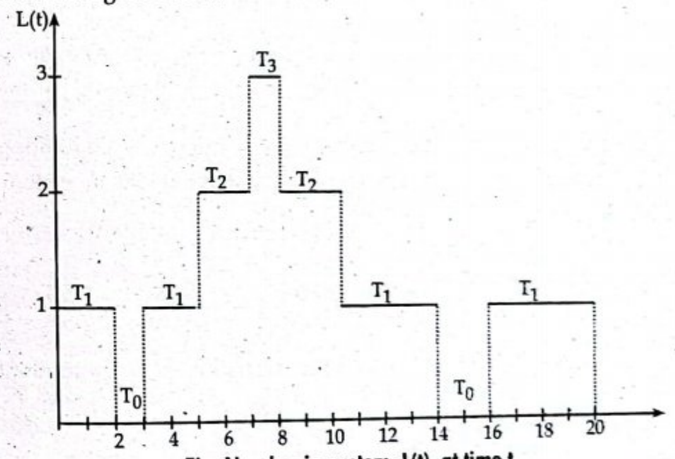


Fig: Number in system, L(t), at time t

Let T_1 denote the total time during [0, T] in which the system contained exactly i customers. In figure above, it is seen that $T_0 = 3$, $T_1 = 12$, $T_1 = 4$ and $T_3 = 1$. (The line segments whose lengths total $T_1 = 12$ are labeled "T_1" in figure etc.) In general, $\sum_{i=0}^{\infty} T_i = T$. The time-weighted-average number in a system is defined by

$$\hat{L} = \frac{1}{T}\sum_{i=0}^{\infty} iT_i = \sum_{i=0}^{\infty} i\left(\frac{T_i}{T}\right) \quad \text{..... (1)}$$

For figure above, $\hat{L} = \frac{[0(3) + 1(12) + 2(4) + 3(1)]}{20} = \frac{23}{20} = 1.15$ customers. Notice that $\frac{T_i}{T}$ is the proportion of time the system contains exactly i customers. The estimator $\hat{L}$ is an example of a time-weighted average. By considering figure

above, it can be seen that the total area under the function L(t) can be decomposed into rectangles of height I and length T_i. For example, the rectangle of area $3 \times T_3$ has base running from t = 7 to t = 8 (thus $T_3 = 1$); however, most of the rectangles are broken into parts, such as the rectangle of area $2 \times T_2$ which has part of its base between t = 5 and t = 7 and the remainder from t = 8 to t = 10 (thus $T_2 = 2 + 2 = 4$). It follows that the total area is given by $\sum_{i=0}^{\infty} i^*T_i = \int_0^T L(t)\,dt$, and therefore that

$$\hat{L} = \frac{1}{T}\sum_{i=0}^{\infty} iT_i = \frac{1}{T}\int_0^T L(t)\,dt \qquad \ldots\ldots (2)$$

The expression in equations (1) and (2) are always equal for any queueing system, regardless of the number of servers. the queue discipline, or any other special circumstances. Equation (2) justifies the terminology time-integrated average.

Many queueing systems exhibit a certain a kind of long-run stability in terms of their average performance. For such systems, as time T gets large, the observed time-average number in the system $\hat{L}$ approaches a limiting value, say L, which is called the long-run time-average number in system-that is with probability 1,

$$\hat{L} = \frac{1}{T}\sum_{i=0}^{\infty} L(t)dt \to L \text{ as } T \to \infty \qquad \ldots\ldots\ldots (3)$$

The estimator $\hat{L}$ is said to be strongly consistent for L. If simulation run length T is sufficiently long, the estimator $\hat{L}$ becomes arbitrarily close to L. Unfortunately, for $T < \infty$, $\hat{L}$ depends on the initial conditions at time 0.

2. Differentiate between chi-square test and KS test for uniformity. Use KS test to check for the uniformity for the input set of random numbers given below.

0.54, 0.73, 0.98, 0.11, 0.68, 0.45. Assume level of significance to be D_α=0.05=> 0.565.

Ans: The **Chi Square test** is used to test whether the distribution of nominal variables is same or not as well as for other distribution matches and on the other hand the **Kolmogorov Smirnov (K-S)** test is only used to test to the goodness of fit for a continuous data.

Kolmogorov Smirnov (K-S) test compares the continuous cdf, F(X), of the uniform distribution to the empirical cdf, SN(x), of the sample of N observations. By definition,

$F(x) = x,\ 0 <= x <= 1$

If the sample from the random-number generator is $R_1, R_2, \ldots\ldots, R_N$, then the empirical cdf, $S_N(X)$, is defined by

$S_N(X)$ = (Number of $R_1, R_2, \ldots\ldots, R_N$ which are <= x)/N

As N becomes larger, SN(X) should become a better approximation to F(X), provided that the null hypothesis is true. The Kolmogorov-Smirnov test is based on the largest absolute deviation or difference between F(x) and $S_N(X)$ over the range of the random variable. I.e. it is based on the statistic

$D = \max | F(x) - S_N(x)|$

The **chi-square test** uses the sample statistic

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$$

Where O_i is the observed number in the i^{th} class, E_i is the expected number in the i^{th} class, and n is the number of classes. For the uniform distribution, E_i i^{st} the expected number in each class is given by: $E_i = N/n$, N is the total number of observation.

Now,

Given sequence of number,

0.54, 0.73, 0.98, 0.11, 0.68 and 0.45

Arranging the given number in ascending order:

0.11, 0.45, 0.54, 0.68, 0.73, 0.98

Here, N = 6

Calculation table for Kolmogorov-Smirnov test :

i	$R_{(i)}$	i/N	i/N- $R_{(i)}$	$R_{(i)}$- (i-1)/N
1	0.11	0.17	0.06	0.11
2	0.45	0.33	-	0.28
3	0.54	0.5	-	0.21
4	0.68	0.67	0.07	0.18
5	0.73	0.83	0.1	0.06
6	0.98	1	0.02	0.15

Now, calculating

$$D^+ = \max_{1 \le i \le N} \left\{ \frac{i}{N} - R_{(i)} \right\} = 0.1$$

$$D^- = \max_{1 \le i \le N} \left\{ R_{(i)} - \frac{i-1}{N} \right\} = 0.28$$

$D = \max(D^+, D^-) = 0.28$

Given, Critical value $D\alpha = 0.565$

Since the computed value, $D\alpha = 0.28$, is less than the tabulated critical value, $D\alpha = 0.565$, the hypothesis of no difference between the distribution of the generated numbers and the uniform distribution is not rejected.

3. What do you understand by static mathematical model? Explain with example. Differentiate between stochastic and deterministic activities.

Ans: Static Mathematical Model

A static model gives the relationships between the system attributes when the system is in equilibrium. If the point of equilibrium is changed by altering any o the attribute values, the model enables the new values for all the attributes to be derived but does not show the way in which they changed to their new values.

For example, in marketing a commodity there is a balance between the supply and demand for the commodity. Both factors depend upon price: a simple market mode! will show what is the price at which the balance occurs.

Demand for the commodity will be low when the price is high, and it will increase as the price drops. The relationship between demand, denoted by Q, and price, denoted by P, might be represented by the straight line marked "Demand" in Fig. below On the other hand, the supply can be expected to increase as the price increases, because the suppliers see an opportunity for more revenue. Suppose supply, denoted by S, is plotted against price, and the relationship is the straight line marked "Supply" in Fig. below. If conditions remain stable, the price will settle to the point at which the two lines cross, because that is where the supply equals the demand.

Since the relationships have been assumed linear, the complete market model can be written mathematically as follows:

$Q = a - bP$

$S = c + dP$

$S = Q$

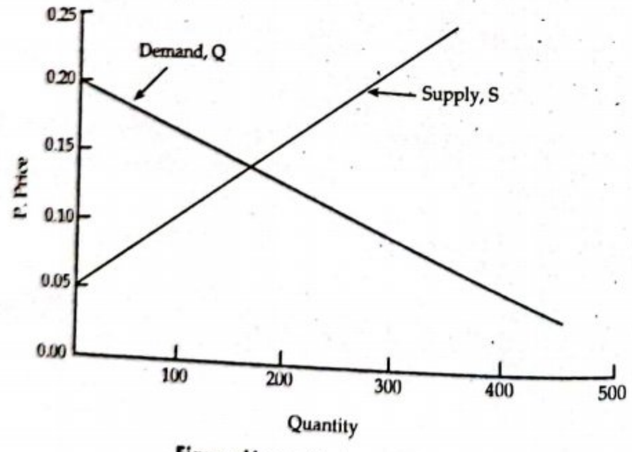


Figure: Linear Market model

The last equation states the condition for the market to be cleared; it says supply equals demand and, so, determines the price to which the market will settle.

For the model to correspond to normal market conditions in which demand goes down and supply increases as price goes up the coefficients b and d need to be positive numbers. For realistic, positive results, the coefficient a must also be positive. Figure above has been plotted for the following values of the coefficients:

$a = 60$

$b = 3{,}000$

$c = -100$

$d = 2{,}000$

The fact that linear relationships have been assumed allows the model to be solved analytically. The equilibrium market price, in fact, is given by the following expression:

$$P = \frac{a - c}{b + d}$$

With the chosen values, the equilibrium price is 0.14, which corresponds to a supply of 180.

More usually, the demand will be represented by a curve that slopes downwards, and the supply by a curve that slopes upwards, as illustrated in Fig below. It may not then the possible to express the relationship by equations that can be solved. Some numeric method is then needed to solve the equations. Drawing the curves to scale and determining graphically where they intersect is one such method.

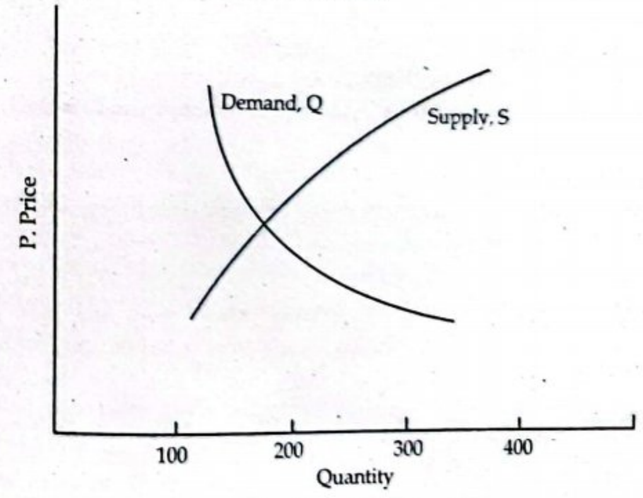


Figure: Non-linear Market model

Stochastic vs. deterministic simulations

A model is deterministic if its behavior is entirely predictable. Given a set of inputs, the model will result in a unique set of outputs. A model is stochastic if it has random variables as inputs, and consequently also its outputs are random.

Consider the donut shop example. In a deterministic model we would for instance assume that a new customer arrives every 5 minutes and an employee takes 2 minutes to serve a customer. In a stochastic model we would on the other hand assume that the arrival times and the serving time follows some random variables: for instance, normal distributions with some mean and variance parameters.

Section B

Attempt any EIGHT questions. **(8 × 5 = 40)**

4. Discuss the merits and demerits of system simulation.

Ans: Advantages

1. **System behavior analysis**
   Simulation software analyzes the behavior of a system without building prototypes. Engineers can work on alternative solutions and simulate designs. Therefore simulations during product design has the advantage of finalizing the best solution for prototyping before actually building it. It helps in reducing the number of design iterations.
2. **Reduces manufacturing cost**
   Simulation software helps engineers in iterating and testing designs very quickly. As a result you can design first time right products. In this way, manufacturing and testing cost reduces significantly.
3. **Faster products to market**
   Advantages of simulation studies include a reduced number of design iterations. As a result, product design cycle time reduces significantly.
4. **Design Analysis: Visual Output**
   It helps engineers to analyze system behavior and product performance more effectively.
5. **Problem solving**
   Simulation software is used to analyze a problem at various levels. This leads to faster and effective problem solving.
6. **Value Engineering**
   Simulation studies can help in reducing the existing and new product manufacturing cost. For example, mold-flow analysis can reduce the injection molding cycle time.

Disadvantages

1. **Higher initial investment**
   The disadvantages of simulation software are high initial investment cost in software license and computing power requirements. Therefore small companies cannot afford them.
2. **Accurate Boundary conditions and input data**
   Simulation results accuracy depends on input data and boundary conditions. System boundary conditions include environmental temperature, pressure and material. Therefore boundary conditions need to be defined accurately to achieve accurate result.

3. **Not 100% accurate**
   Simulation software is not 100% accurate. You should expect some discrepancies in simulation results and tested products. But with experience, engineers can refine their simulation results.

5. Explain Markov's chain with a suitable example.

Ans: A Markov chain is a mathematical process that transitions from one state to another within a finite number of possible states. It is a collection of different states and probabilities of a variable, where its future condition or state is substantially dependent on its immediate previous state.

Markov chains are a fundamental part of stochastic processes. They are used widely in many different disciplines. A Markov chain is a stochastic process that satisfies the Markov property, which means that the past and future are independent when the present is known. This means that if one knows the current state of the process, then no additional information of its past states is required to make the best possible prediction of its future. This simplicity allows for great reduction of the number of parameters when studying such a process.

In mathematical terms, the definition can be expressed as follows:

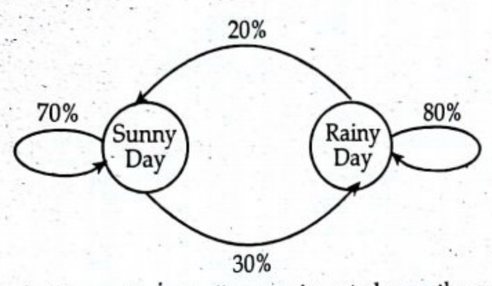
A stochastic process $X = \{X_n, n \in N\}$ in a countable space S is a discrete-time Markov chain if:

For all $n \geq 0$, $Xn \in S$

For all $n \geq 1$ and for all $i_0, \ldots i_{n-1}$, in ε S, we have :

$P\{ X_n = in \mid X_{n-1} = in-1, \ldots, X_0 = i_0 \} = P\{ X_n = in \mid Xn-1 = in-1 \}$

Markov chains are used to compute the probabilities of events occurring by viewing them as states transitioning into other states, or transitioning into the same state as before. We can take weather as an example: If we arbitrarily pick probabilities, a prediction regarding the weather can be the following: If it is a sunny day, there is a 30% probability that the next day will be a rainy day, and a 20% probability that if it is a rainy day, the day after will be a sunny day. If it is a sunny day, there is therefore a 70% chance that the next day will be another sunny day, and if today is a rainy day, there is 80% chance that the next day will be a rainy day as well. This can be summarized in a transition diagram, where all of the possible transitions of states are described:



To approach this mathematically one views today as the current state, S_0, which is a $1 \times m$ vector. The elements of this vector will be the current state of the process. In our weather example, we define S = [*Sunny Rainy*]. Where S

is called our state space, in which all the elements are all the possible states that the process can attain. If, for example, today is a sunny day, then the so vector will be S_0 = [1 0], because there is 100% chance of a sunny day and zero chance of it being a rainy day. To get to the next state, the transition probability matrix is required, which is just the state transition probabilities summarized in a matrix. In this case it will be as follows:

	S	R
S	0.7	0.3
R	0.2	0.8

or more generally in this case:

P =	S	R
S	α	1 – α
R	β	1 – β

To get to the next state, S1, you simply calculate the matrix product S1 = S0P. Since calculations for successive states of S is only of the type $S_n = S_{n-1}P$, the general formula for computing the probability of a process ending up in a certain state is

$Sn = S0P_n$.

This allows for great simplicity when calculating the probabilities far into the future. For example, if today is a sunny day then the state vector 120 days from now, S^{120}, is

S^{120} = [0.4 0.6] . [3]

So, A Markov chain is a mathematical model for a process which moves step by step through various states. In a Markov chain, the probability that the process moves from any given state to any other particular state is always the same, regardless of the history of the process. It consists of states and transition probabilities. Each transition probability is the probability of moving from one state to another in one step. The transition probabilities are independent of the past and depend only on the two states involved. The matrix of transition probabilities is called the transition matrix.

Key Features of Markov Chains

A sequence of trials of an experiment is a Markov chain if

- the outcome of each experiment is one of a set of discrete states.
- the outcome of an experiment depends only on the present state, and not on any past states.
- the transition probabilities remain constant from one transition to the next.

6. Define arrival pattern. Explain non-stationary Poisson process.

Ans: Solution

Arrival/defines the way customers enter the system. Mostly the arrivals are random with random intervals between two adjacent arrivals. Typically the arrival is described by a random distribution of intervals also called Arrival Pattern. Arrivals may occur at scheduled times or at random times. When at random times, the inter arrival times are usually characterized by a probability distribution and most important model for random arrival is the poisson process. In schedule arrival interarrival time of customers are constant.

The non-stationary Poisson process is a Poisson process for which the arrival rate varies with time. More specifically, it can be defined as follows:

The counting process N(t) is a non-stationary Poisson process if:

a) The process has independent increments.

b) $\Pr[N(t+dr) - N(t)] \begin{cases} = 0] = 1 - \lambda(t)dt \\ = 1] = \Lambda(t)dt \\ > 1] = 0 \end{cases}$

where, $\lambda(t)$ = the arrival rate at time t

dt = differential sized interval

7. Differentiate between validation and calibration. How can we perform validation of model?

Ans: Calibration is a process or action that compares the measurement values of a measuring device or equipment against a reference standard and certifies the measurement accuracy.

On the other hand, **validation** is a detailed documented process of confirming that the equipment or machine is installed correctly, operating effectively, and performing without any error.

Calibration is an operation to confirm that the equipment or measuring device is performing accurately. Whereas, validation is an operation to confirm that the analytical method is performing accurately. Calibration provides: Is equipment or measuring device accurate? Validation provides: Is equipment or measuring device system function satisfactorily?

Difference between calibration and validation

Calibration	Validation
1. Calibration is a process or action that compares the measurement values of a measuring device or equipment against a reference standard and certifies the measurement accuracy.	1. Validation is a detailed documented process of confirming that the equipment or machine is installed correctly, operating effectively, and performing without any error.
2. It is system performance checking.	2. It is a performance analysis methodology.
3. To nullify or remove the deviation by comparing with a reference or know standard.	3. To verify the consistency of predictable results of the method, process, or procedure.
4. To confirm that the equipment or measuring device is performing accurately.	4. To confirm that the analytical method is performing accurately.
5. It is done by comparing with a reference standard	5. There are no known standard or reference standards used.
6. Executed according to calibration SOP.	6. Executed according to the validation protocol.

7. To ensure: • Specificity • Linearity • Accuracy • Sensitivity	7. a. It will minimize rejection loss b. Reduction in utility cost c. Help timely corrective action d. Assure consistent production performance e. Ensure achievement of quality goals f. Reduce extensive finished product testing g. Easier maintenance of the equipment. h. Allow parametric release i. Sort out the main risk of production liability j. More rapid automation

8. Use maxed congruential method to generate a sequence of random numbers with $X_0 = 27$, a=17,m=100 and c=43.

Ans: Given,
$X_0 = 27$,
$a = 17$,
$c = 43$ &
$m = 100$
We have,
$X_{i+1} = (a X_i + c) \bmod m$
Mixed Congruential Method: $c \neq 0$
& $R_i = X_i/m$,
The sequence of random numbers are calculated as follows:
$X_0 = 27$
$R_0 = 27/100 = 0.27$
$X_1 = (a X_0 + c) \bmod m = (17*27+43) \bmod 100 = 502 \bmod 100 = 2$
$R_1 = 2/100 = 0.02$
$X_2 = (a X_1 + c) \bmod m = (17*2+43) \bmod 100 = 77 \bmod 100 = 77$
$R_2 = 77/100 = 0.77$
$X_3 = (a X_2 + c) \bmod m = (17*77+43) \bmod 100 = 1352 \bmod 100 = 52$
$R_3 = 52/100 = 0.52$
$X_4 = (a X_3 + c) \bmod m = (17*52+43) \bmod 100 = 927 \bmod 100 = 27$
$R_4 = 27/100 = 0.27$
Therefore,
The sequence of random integers are 27, 02, 77, 52, 27 & so on.
The sequence of random numbers are 0.27, 0.02, 0.77, 0.52, 0.27 & so on.

9. What do you understand by replication of runs. Why is it necessary?

Ans: Replication of run is used to obtain independent results by repeating the simulation. Repeating the experiment with different random numbers for the same sample size n gives a set of independent determinations of the sample mean $\bar{x}$. The mean of the means and the mean of the variances are then used to estimate the confidence interval.

Suppose the experiment is repeated p times with independent random values of n sample sizes. Let x_{ij} be the i^{th} observation in j^{th} run and let the sample mean and the variance for the j^{th} run is denoted by $\bar{x}_j(n)$ and $S_j^2(n)$ respectively. Then for j^{th} run, the estimates are;

$$\bar{x}_j(n) = \frac{1}{n}\sum_{i}^{n} x_{ij}$$

$$s_j^2(n) = \frac{1}{n-1}\sum_{i=1}^{n} [x_{ij} - \bar{x}_j(n)]^2$$

Combining the result of p independent measurement gives the following estimate for the mean $\bar{x}$ and variance s^2 of the populations as:

$$\bar{x} = \frac{1}{p}\sum_{j=1}^{p} \bar{x}_j$$

$$s^2 = \frac{1}{p}\sum_{j=1}^{p} s_j^2$$

Replication of runs are necessary for running experiments based on scenarios with stochastic parameters. It is used to obtain independent results by repeating the simulation. If replications are not used, a single run of an experiment will not produce statistically significant results and will not allow for proper calculation of statistical data.

10. Explain generation of non-uniform random number generation using inverse method.

Ans: Non-uniform random variate generation is concerned with the generation of random variables with certain distributions. Such random variables are often discrete, taking values in a countable set, or absolutely continuous, and thus described by a density. The methods used for generating non-uniform random variate are inverse transformation technique and acceptance/rejection technique.

Non Uniform transformation method/Inverse transform method

The inverse transform technique can be used to sample from the exponential, uniform, triangular distribution etc. by inversing the CDF of those probability distributions. The inverse transform technique can be utilized for any distribution when the cdf, F(x), is of a form that its inverse F^{-1} can be computed easily.

The random variate generation process using exponential distribution:

1. Compute the cdf of the random variable X for exponential distribution.
2. Set $F(X) = R$ on the range of X i.e. $1-1-e^{-\lambda x} = R$
3. Solve the equation, $1-e^{-\lambda x} = R$ in terms of R

$X = -1/\lambda \ln(1-R)$

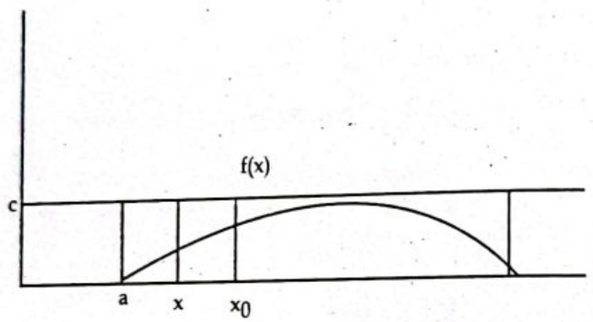
This equation is called random variate generator for the exponential; distribution. In general equation it is written as $X = F^{-1}(R)$

Acceptance/Rejection Method

The rejection method for obtaining samples of random numbers forms a given non-uniform distribution works by generating uniform random numbers repeatedly and accepting only those numbers that meet certain conditions.

The rejection method is applied when the probability density function, f(x), has a lower and upper limit to its range, 'a' and 'b', respectively, and an upper bound 'c'. The method can be specified as follows:

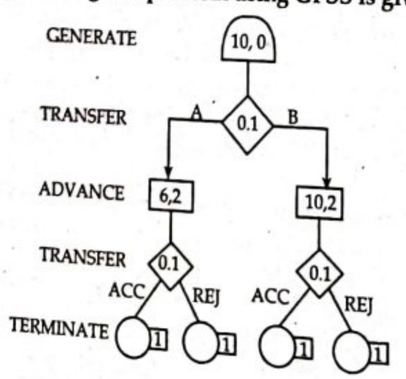
Let $\Delta x = x - x_0$



1. Find the maximum value c of f(x) on $a \le x \le b$.
   $f(X) \le c \; \forall \; x \in [a, b]$
2. Compute two values μ_1, μ_2 of the uniformity distributed variables, both defined on [a, b] = [0, 1].
3. Compute $x_0 = a + \mu_1(b - a)$
4. Compute $y_0 = c\,\mu_2$
5. If $y_0 \le f(x_0)$, accept x_0 as desired output; otherwise reject x0 and repeat the process with two next values μ_1 & μ_2.

11. Parts are being made at the rate of one every 10 minutes. They are of two types, A and B. And are mixed randomly with about 10% being type B. A separate inspector is assigned to examine each part. Inspection of part A takes 6 ± 2 minutes while B takes 10 ± 2 miutes. Both inspector rejects 10% of parts they inspect. Draw GPSS block diagram to simulate the above problem for 100 parts.

Ans: The block diagram for given problem using GPSS is given below:



Code for simulating the given problem using GPSS:

```
GENERATE  10, 0
TRANSFER  0.1  A  B
A  ADVANCE  6, 2
B  ADVANCE  10, 2
A  TRANSFER  0.1  ACC  REJ
B  TRANSFER  0.1  ACC  REJ
A  ACC TERMINATE  1
   REJ TERMINATE  1
B  ACC TERMINATE  1
   REJ TERMINATE  1
START 100
```

12. Write short notes on (any two): (2 x2.5=5)

a. System and its environment

Ans: A system is defined as an aggregation or assemblage of objects joined in some regular interaction or interdependence toward the accomplishment of some purpose.

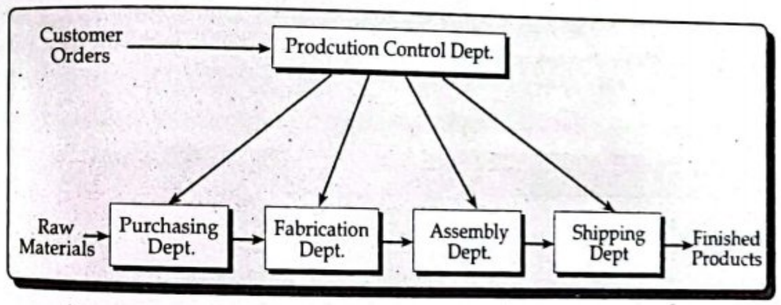


Fig: Production system

Example: Production System

In the above system there are certain distinct objects, each of which possesses properties of interest. There are also certain interactions occurring in the system that cause changes in the system.

Example: An aircraft under autopilot control

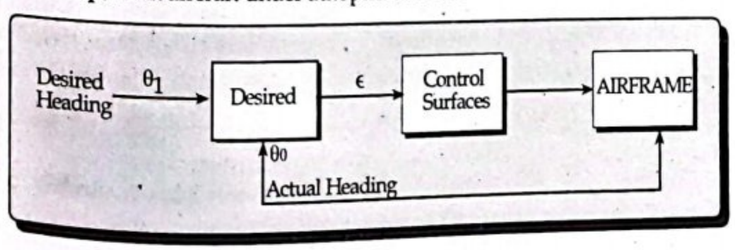


Fig: An Aircraft under autopilot control

System Environment

The external components which interact with the system and produce necessary changes are said to constitute the system environment.

In modeling systems, it is necessary to decide on the boundary between the system and its environment. This decision may depend on the purpose of the study.

Ex: In a factory system, the factors controlling arrival of orders may be considered to be outside the factory but yet a part of the system environment. When, we consider the demand and supply of goods, there is certainly a relationship between the factory output and arrival of orders. This relationship is considered as an activity of the system.

b. Simulation run statistics

Ans: In most of the simulation study, the assumptions of stationary and mutually independent observations do not apply. An example of such case is queuing system. Correlation is necessary to analyze such scenario. In such cases, simulation run statistics method is used.

Example:

Consider a system with Kendall's notation M/M/1/FIFO (i.e. a single server system in which the inter-arrival time is distributed exponentially ;and service time has an exponential and queue discipline is FIFO) and the objective is to measure the mean waiting time.

In simulation run approach, the mean waiting time is estimated by accumulating the waiting time of n successive entities and then it is divided by n. This measures the sample mean such that:

$$\bar{x}(n) = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Whenever a waiting line forms, the waiting time of each entity on the line clearly depends upon the waiting time of its predecessors. Such series of data in which one value affect other values is said to be autocorrelated. The sample mean of autocorrelated data can be shown to approximate a normal distribution as the sample size increases.

A simulation run is started with the system in some initial state, frequently the idle state, in which no service is being given and no entities are waiting. The early arrivals then have a more than normal probability of obtaining service quickly, so a sample mean that includes the early arrivals will be biased.

TU QUESTIONS-ANSWERS 2078

Bachelor Level/Third Year/Fifth Semester/Science
Course Title: Simulation and Modeling — Full Marks: 60
Course Code: CSC 317 — Pass Marks: 24
Time: 3 hrs.

Candidates are required to give their answers in their own words as far as practicable. The figures in the margin indicate full marks.

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Define queuing system. Explain the Kendall's notation for queuing system. What are the various performance measure in single server queuing system? Explain which of the them determine system stability and how?

Ans: Queuing systems are simplified mathematical models to explain congestion. Broadly speaking, a queuing system occurs any time 'customers' demand 'service' from some facility; usually both the arrival of the customers and the service times are assumed to be random. If all of the 'servers' are busy when new customers arrive, these will generally wait in line for the next available server. Simple queuing systems are defined by specifying the following

(a) the arrival pattern,
(b) the service mechanism, and
(c) queue discipline.

The queue discipline indicates the order in which members of the queue are selected for service. It is most frequently assumed that the customers are served on a first come first serve basis. This is commonly referred to as FIFO (first in, first out) system. Occasionally, a certain group of customers receive priority in service over others even if they arrive late. This is commonly referred to as priority queue. The queue discipline does not always take into account the order of arrival. The server chooses one of the customers to offer service at random. Such a system is known as service in random order (SIRO).

Different queue discipline are listed below:

- FIFO: first-in-first-out
- LIFO: last-in-first-out
- SIRO: service in random order
- SPT: shortest processing time first
- PR: service according to priority
- RR: round robin

Kendall's notation for queuing system

Kendall's Notation is a system of notation according to which the various characteristics of a queuing model are identified. Kendall (Kendall, 1951) has introduced a set of notations which have become standard in the literature of queuing models. A general queuing system is denoted by (a/b/c): (d/e) where,

a	=	probability distribution of the inter-arrival time.
b	=	probability distribution of the service time.
c	=	number of servers in the system.
d	=	maximum number of customers allowed in the system.
e	=	queue discipline

Single-Server Queue

A queueing system is described by its calling population, the nature of the arrivals, the service mechanism, the system capacity, and the queueing discipline. A single-channel queueing system is portrayed in figure below.

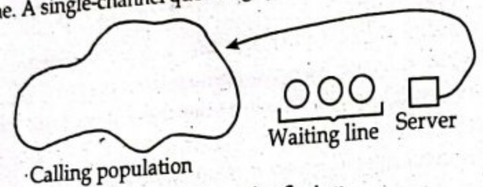


Figure: Queueing System

In the single-channel queue, the calling population is infinite; that is, if a unit leaves the calling population and joins the waiting line or enters service, there is no change in the arrival rate of other units that may need service.

Arrivals for service occur one at a time in a random fashion; once they join the waiting line, they are eventually served. In addition, service times are of some random length according to a probability distribution which does not change over time.

The system capacity; has no limit, meaning that any number of units can wait in line.

Finally, units are served in the order of their arrival by a single server or channel.

Arrivals and services are defined by the distributions of the time between arrivals and the distribution of service times, respectively.

For any simple single or multi-channel queue, the overall effective arrival rate must be less than the total service rate, or the waiting line will grow without bound. When queues grow without bound, they are termed **"explosive"** or **unstable.**

The state of the system is the number of units in the system and the status of the server, busy or idle.

An event is a set of circumstances that cause an instantaneous change in the state of the system. In a single –channel queueing system there are only two possible events to affect the state of the system.

2. **Define true random numbers and pseudo random numbers with its properties. The sequence of numbers 0.64, 0.50, 0.25, 0.58, 0.72, 0.90 has been generated. Use KS Test with Dα = 0.050=>0.512 to determine if the hypothesis that they are uniformly distributed on interval [0, 1] can be rejected.**

Ans: Random number

As the term suggests, a random number is a number chosen by chance – i.e., randomly, from a set of numbers. All the numbers in a specified distribution have equal probability of being chosen randomly.

A random number occurs in a specified distribution only when two conditions are met: The values are uniformly distributed over a defined interval or set, and it is impossible to predict future values based on past or present ones.

Properties of Random Numbers

A sequence of random numbers, R_1, R_2, R_3, ... must have two important properties:

- **uniformity,** i.e. they are equally probable every where
- **independence,** i.e. the current value of a random variable has no relation with the previous values

Pseudo Random Number

Software-generated random numbers only are pseudorandom. They are not truly random because the computer uses an algorithm based on a distribution, and are not secure because they rely on deterministic, predictable algorithms.

Pseudo Random Number Generator (PRNG) refers to an algorithm that uses mathematical formulas to produce sequences of random numbers. PRNGs generate a sequence of numbers approximating the properties of random numbers. A PRNG starts from an arbitrary starting state using a seed state. Many numbers are generated in a short time and can also be reproduced later, if the starting point in the sequence is known. Hence, the numbers are deterministic and efficient.

Characteristics of PRNG

- **Efficient:** PRNG can produce many numbers in a short time and is advantageous for applications that need many numbers
- **Deterministic**: A given sequence of numbers can be reproduced at a later date if the starting point in the sequence is known.Determinism is handy if you need to replay the same sequence of numbers again at a later stage.
- **Periodic:** PRNGs are periodic, which means that the sequence will eventually repeat itself. While periodicity is hardly ever a desirable characteristic, modern PRNGs have a period that is so long that it can be ignored for most practical purposes

Arranging the given number in ascending order:

0.11, 0.45, 0.54, 0.68, 0.73, 0.98

Here, N = 6

Calculation table for Kolmogorov-Smirnov test :

I	$R_{(i)}$	i/N	i/N- $R_{(i)}$	$R_{(i)}$- (i-1)/N
1	0.11	0.17	0.06	0.11
2	0.45	0.33	-	0.28
3	0.54	0.5	-	0.21
4	0.68	0.67	0.07	0.18
5	0.73	0.83	0.1	0.06
6	0.98	1	0.02	0.15

Now, calculating

$$D^+ = \max_{1 \le i \le N} \left\{ \frac{i}{N} - R_{(i)} \right\} = 0.1$$

$$D^- = \max_{1 \le i \le N} \left\{ R_{(i)} - \frac{i-1}{N} \right\} = 0.28$$

$$D = \max(D^+, D^-) = 0.28$$

Given, Critical value $D\alpha = 0.565$

Since the computed value, $D\alpha = 0.28$, is less than the tabulated critical value, $D\alpha = 0.565$, the hypothesis of no difference between the distribution of the generated numbers and the uniform distribution is not rejected.

3. **What do you understand by dynamic mathematical model? Explain with example. Differentiate it with static mathematical model.**

Ans: Dynamic mathematical model allows the changes of system attributes to be derived as a function of time.

The derivation may be made with an analytical solution or with a numerical computation, depending upon the complexity of the model. The equation to describe the behavior of the car wheel is an example of a dynamic mathematical model. It is not possible to find analytic solution of this equation and one has to adopt the numerical methods.

We divide equation by M and write in the following form,

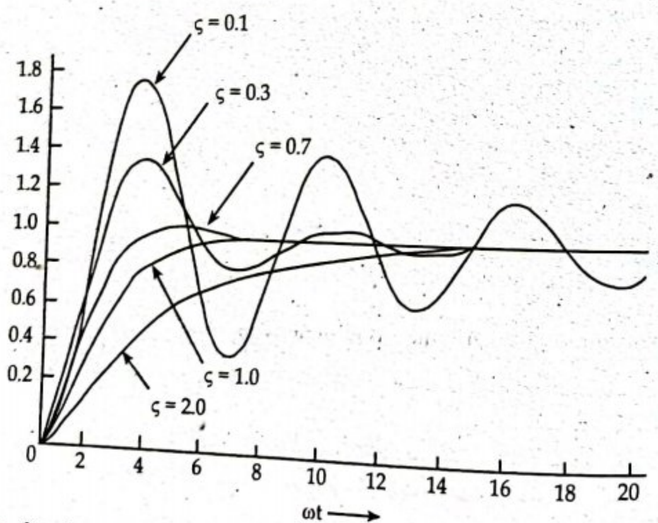
$$\ddot{x} + 2\varsigma w\dot{x} + w^2x = w^2F(t)$$

Where, $2\varsigma w = \frac{D}{M}$ and $w^2 = \frac{k}{M}$

The solution can be given in terms of the variable w(t), where w is the oscillation given by,

$$w = 2\pi f$$

And f is the number of cycles per second



The fig. shows how x varies in response to a steady force applied at time t =0. It can be seen that when $\zeta < 1$, the motion is oscillatory.

A dynamic model accounts for time-based adjustments withinside the state of the system, even as a static (or steady-state) version calculates the system in equilibrium, and hence is time-invariant.

Any system may be defined using a mathematical model that contains mathematical symbols and concepts. Mathematical modeling is the name of the system that is undertaken to broaden a version for a specific system. It isn't always just life sciences however additionally social sciences that make heavy use of these mathematical models. In fact, it's far in an artwork subject like economics that these mathematical models are used extensively. There are many kinds of mathematical models however there's no tough and rapid rule and there's a pretty a bit of overlapping in exceptional models. One manner to categorize mathematical models is to place them into static modelling and dynamic modelling.

Section B

Attempt ANY EIGHT questions. (8 × 5 = 40)

4. **Describe the pases in simulation.**

Ans: Although simulations vary in complexity from situation to situation, in general one would have to go through the following steps:

Step 1→Define the problem or system you intended to simulate.
Step 2→Formulate the model you intend to use.
Step 3→ Test the model; compare its behaviour with the behaviour of the actual problem.
Step 4→Identify and collect the data needed to test the model.
Step 5→ Run the simulation
Step 6→ Analyze the results of the simulation and, if desired, change the solution you are evaluating.
Step 7→ Rerun the simulation to test the new solution.
Step 8→Validate the simulation; this involves increasing the chances of the inferences you may draw about the real situation

5. **Explain the concept of discrete event simulation. Explain poisson's arrival pattern.**

Ans: Discrete event simulation (DES) is a method used to model real world systems that can be decomposed into a set of logically separate processes that autonomously progress through time. Each event occurs on a specific process, and is assigned a logical time (a timestamp). The result of this event can be an outcome passed to one or more other processes. The content of the outcome may result in the generation of new events to be processed at some specified future logical time. The underlying statistical paradigm that supports DES is based in queuing theory. The approach has been used historically to evaluate telephone scheduling of and more recently computer network job allocation.

Poison arrival Patterns

Second condition says that arrival of a customer is completely random. This means that an arrival can occur at any time and the time of next arrival is independent of the previous arrival. With this assumption it is possible to show that the distribution of the inter-arrival time is exponential. This is

equivalent to saying that the number of arrivals per unit time is a random variable with a Poisson's distribution. This distribution is used when chances of occurrence of an event out of a large sample is small.

That is if X = number of arrivals per unit time, then, probability distribution function of arrival is given as,

$$f(x) = Pr(X = x) = \frac{e^{-\lambda}\lambda^x}{x!}, \begin{cases} x = 0, 1,2,3,... \\ \lambda > 0 \end{cases}$$

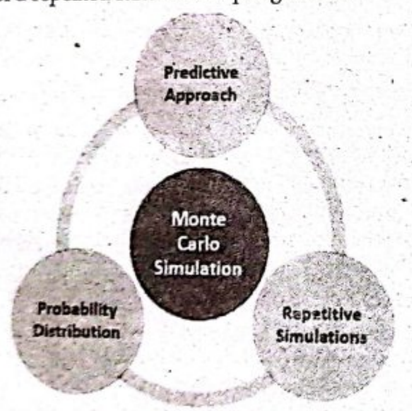
$E(X) = \lambda$

Where λ is the average number of arrivals per unit time $(1/\tau)$, $E(X)$ is the expected number, and x is the number of customers per unit time. This pattern of arrival is called Poisson's arrival pattern. τ is inter arrival time.

6. Explain Monte Carlo simulation method with an example.

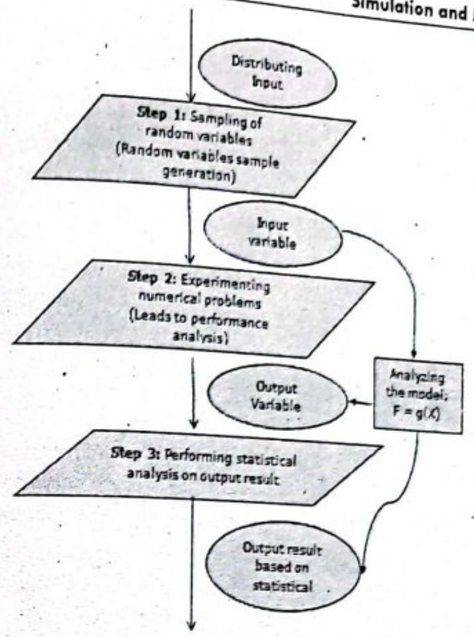
Ans: Monte Carlo simulation is a computerized mathematical technique to generate random sample data based on some known distribution for numerical experiments. This method is applied to risk quantitative analysis and decision making problems. This method is used by the professionals of various profiles such as finance, project management, energy, manufacturing, engineering, research & development, insurance, oil & gas, transportation, etc.

The three most prominent ways of reaching the best possible outcomes evolving out of a repeated random sampling include:



- **Predictive Approach** – It determines dependent and independent variables to obtain the desired range of findings.
- **Probability Distribution** – This method identifies independent variables as they are responsible for different possibilities of multiple outcomes that would occur. It is one of the best ways to find uncertainties accurately and getting prepared for the same accordingly.
- **Repeated Simulations** – This technique allows repeating the simulation 'n' number of times. It gives results that are more likely to affect a particular process.

The following illustration shows a generalized flowchart of Monte Carlo simulation



7. Define the terms verification, calibration, validation and accreditation of models.

Ans: Verification, Validation, and Accreditation (VV&A) are three interrelated but distinct processes that gather and evaluate evidence to determine whether a model or simulation should be used in a given situation and establishing its credibility. The decision to use the simulation will depend on the simulation's capabilities and correctness, the accuracy of its results, and its usability in the specified application.

Verification: The process of determining that a model implementation and its associated data accurately represent the developer's conceptual description and specifications.

Validation: The process of determining the degree to which a model and its associated data provide an accurate representation of the real world from the perspective of the intended uses of the model.

On the other hand, **validation** is a detailed documented process of confirming that the equipment or machine is installed correctly, operating effectively, and performing without any error.

Accreditation: The official certification that a model, simulation, or federation of models and simulations and its associated data is acceptable for use for a specific purpose.

Calibration is a process or action that compares the measurement values of a measuring device or equipment against a reference standard and certifies the measurement accuracy.

Calibration is an operation to confirm that the equipment or measuring device is performing accurately. Whereas, validation is an operation to confirm that the analytical method is performing accurately. Calibration provides: Is equipment or measuring device accurate? Validation provides: Is equipment or measuring device system function satisfactorily?

8. Use Multiplicative congruential method to generate a sequence of random numbers with X0=7, a=11, m=16.

Ans: X1= (aX0 + c) mod m = 11*7 mod 16 = 77 mod 16 = 13

R1 = X1 /m = 13/16 = 0.8125

X2 = (aX1 + c) mod m = 11*13 mod 16 = 143 mod 16 = 15

R2 = X2 /m = 15/16 = 0.9375

X3= (aX2 + c) mod m = 11*15 mod 16 = 165 mod 16 = 5

R3 = X3 /m = 5/16 = 0.3125

X4= (aX3 + c) mod m = 11*5 mod 16 = 55 mod 16 = 7

R4 = X4 /m = 7/16 = 0.4375

And so on....

9. Why is estimation methods used in simulation? Explain.

Ans: Whenever a random variable is introduced to the simulation model, all the system variables that describe its behaviour become random or stochastic. The values of the variables involved in the system will fluctuate as the simulation proceeds. So, arbitrary measurement of the values of these variables can not represent the true value. For this, some conditions about the probability of the true value falling within a given interval about the estimated value must be made. Such interval is known as confidence interval. In simulation study, it is assumed that the observations being made are mutually independent. But, in most of the real world problems, simulation results are mutually dependent. The various methods used to analyze simulation results are as follows:

1. Estimation Methods
2. Simulation Run Statistics
3. Replication of Runs
4. Elimination of Initial Bias

Estimation Method

It is assumed that the random variables are stationary and independent drawn from an infinite population with a finite mean and finite variance. Such random variables are independently and identically distributed (IID) random variable. The central limit theorem can be applied to IID data. It states that "the sum of n numbers of IID variables, drawn from a population that has a mean of μ and a variance of σ^2, is approximately distributed as a normal variable with a mean of $n\mu$ and a variance of $n\sigma^2$." The normal distribution can be transformed into a standard normal distribution, that has a mean of 0 and a variance of 1.

Example:

Let us consider x(i) where i = 1, 2, 3, ..., n be the n number of random variables drawn from a sample of population with mean μ and variance σ^2.

Using central limit theorem, and transforming to standard normal distribution, we get:

$Z = (\sum x(i) - n\mu) / \{n^{(1/2)} * \sigma\}$

Dividing top and bottom by n, we get:

$Z = (x - \mu) / (\sigma / n^{(1/2)})$

Where $x = \sum x(i) / n$ = sample mean

10. Explain the importance of elimination of initial bias during simulation.

Ans: The many experimental results show the need to remove the initial bias, or reduce its effects. Two general approaches can be taken to remove the bias: the system can be started in a more representative state than the empty state, or the first part of the simulation run can be ignored.

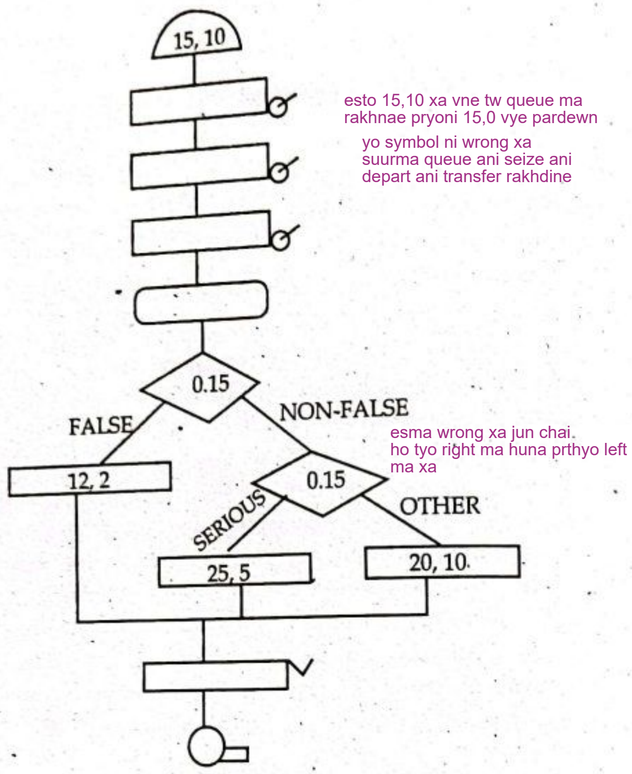
In some simulation studies, particularly of existing systems, there may be information available on the expected conditions that makes it feasible to select better initial conditions. The ideal situation is to know the steady state distribution. In the study previously discussed, Law repeated the experiments on the M/M/1 system, supplying an initial waiting line for each run, selecte4d at random from the known steady state distribution of the waiting line. The case of 40 repetitions of 320 samples, which previously resulted in a coverage of only %, was improved to a coverage of 88%. Of course, the theoretical knowledge on which this technique is based is not usually available. However, experience with an existing system, or similar type of system, could provide a reasonable approximation.

The more common approach to removing initial bias is to eliminate an initial section of the run. The run is started from an idle state and stopped after a certain period of time. The entities existing in the system at that time are left as they are. The run is then restarted with statistics being gathered from the point of restart. As a practical matter, it is usual to program the simulation so that statistics are gathered from the beginning, and simply wipe out the statistics gathered up to the point of restart. No simple rules can be given to decide how long an interval should be eliminated. It is advisable to use some pilot runs starting from the idle state to judge how long the initial bias remains.

Another disadvantage of eliminating the first part of a simulation run is that the estimate of the variance, needed to establish a confidence limit, must be based on less information. The reduction in bias, therefore, is obtained at the price of increasing the confidence interval size.

11. Ambulances are dispatched at a rate of one every 15 (+ or -) 10 mins. Fifteen percent of the calls are false alarms, which require 12 (+ or -) 2 mins to complete. All other calls can be one of two kinds. The first kind are classified as serious. They constitute 15% of non false alarms and take 25 (+ or -) 5 mins to complete. The other calls take 20 (+ or -) 10 mins. Simulate the model using GPSS.

Ans:



esto 15,10 xa vne tw queue ma rakhnae pryoni 15,0 vye pardewn

yo symbol ni wrong xa suurma queue ani seize ani depart ani transfer rakhdine

esma wrong xa jun chai. ho tyo right ma huna prthyo left ma xa

Fig: GPSS Model of Ambulance Dispatcher

12. Write short notes on (any two) (2x2.5=5)

a. Digital analog simulator

Ans: Digital Analog Simulation (DAS) is a programming technique which makes a digital computer operate much like an analog computer. Although it is not intended primarily as an analog computer simulator, the similarity of DAS programming to analog programming is readily apparent. DAS combines remarkable ease and speed of programming with reasonably high computing speed.

The DAS input language is designed to permit a simple and concise description of an analog-style block diagram of the problem to be solved. The blocks in the diagram are restricted to summers, integrators, multipliers, limiters, relays and other "components" for which macro-instructions appear in the DAS compiler.

The construction of a DAS block diagram is more straight-forward than an analog computer block diagram because there is virtually no restriction on the number of available components, and because no amplitude scaling is required.

b. Simulation tools

Ans: Learning tools in which a real-life situation is simulated using models or interactive computer programs are called simulation tools.

There are a number of powerful simulation tools available. All of them have advantages and disadvantages. They can be grouped into three parts; according to the way the system to be simulated is 'entered into the computer':

Mathematical input

In this case an exact mathematical description of the circuit to be simulated is entered into the machine. This can be done with various programming languages such as Basic, Fortran, etc., or the very practical Matlab. (Matlab with the toolbox 'Simulink' has an additional feature: the possibility to enter the mathematical description in graphic form.)

Netlist input

This form of input was used, for example, by former versions of Spice. The physical elements of a circuit, i.e., resistors, capacitors, active elements were entered on the keyboard as a description list. This form of entering a circuit can still be used in Spice. But, it is no longer necessary. Nowadays, the circuit can be entered into the computer graphically.

Graphical input

This group contains the tools with the most comfortable form of entering the system into the machine. Most of today's simulation packages for power electronic provide graphical input.

Example: Matlab / Simulink / SimPowerSystems / PLECS

Matlab is a mathematical tool that has been established for a long time. Toolboxes for various applications exist. One of them is Simulink, a graphical tool for the entering of functions. Simulink itself can be expanded with another toolbox: SimPowerSystem. This toolbox is designed for the simulation of electrical power systems including power electronics. The elements of the various toolboxes can be combined.

MODEL QUESTIONS SETS FOR PRACTICE

MODEL SET 1

Course Title: Simulation and Modeling **Full Marks: 60**
Course No: CSC317 **Pass Marks: 24**
Semester: V **Time: 3 hrs**

Candidates are required to give their answers in their own words as far as practicable.

Group A

Attempt any TWO questions: **(2x10=20)**

1. What is model? What are the different types of models? Give example for each.
2. Define the queuing system. Explain the elements of queuing system with example.
3. What are the properties of random number? The sequence of numbers 0.54, 0.73, 0.98, 0.11 and 0.68 has been generated. UsetheKolmogorov-Smirnovtestα=0.05todetermineifthehypothesisthatthenumbers are uniformly distributed on the interval 0 to 1 can be rejected. (Note that the critical value of D for α = 0.05 and N = 5 is 0.565).

Group B

Attempt any EIGHT questions: **(8x5=40)**

4. When is simulation appropriate and when it is not?
5. Define congestion in a queuing system, and describe its major characteristics.
6. What do you mean by pseudo random numbers?
7. Describe the rejection method of generating the random numbers.
8. Draw and describe the different types of GPSS blocks that are used to gather statistics?
9. Explain Hybrid simulation with example.
10. Explain the distributed lag model.
11. What do you mean by simulation tool?
12. How do you eliminate the effect of transient and initial bias in simulation output?
13. Write short notes on:
    a. Markov Chain
    b. Feedback systems

MODEL SET 2

Group A

Attempt any TWO questions: **(2x10=20)**

1. What do you mean by Queuing system? Explain the characteristics of Queuing system with example.
2. What is the main objective of gap test? Explain gap test algorithm with example.
3. Consider that a machine tool in a manufacturing shop is turning out parts at the rate of one every 5 minutes. As they are finished, the parts go to an inspector, who takes 4±3 minutes to examine each one and rejects about 10% of the parts. Now, develop a block diagram and write the code for simulating the above problem using GPSS, and also explain the function of each block used in the block diagram in detail.

Group B

Attempt any EIGHT questions: **(8x5=40)**

4. Describe the process of model building, verification, and validation in brief.
5. Explain the process of testing for auto-correlation test.
6. Explain any four program control statements that are used in GPSS.
7. Define activity, event and state variables. List out the activities and events for the following systems; (A) Super market (B) Inventory control (C) Hospital.
8. Describe the basic nature of simulation in brief.
9. Differentiate between fixed time step and event to event model with the help of suitable examples.
10. Verificationisconcernedwithbuildingthe"modelright"andvalidationis concerned with building the "right model". Justify it with suitable reasons.
11. Explain with example of calibration and validation of model.
12. Describe the process of model building, verification, and validation in brief.
13. Describe the distributed lag model with the help of any practical example.

MODEL SET 3

Group A

Attempt any two questions: (2x10=20)

1. Explain the independence test. A sequence of 1000 four digit numbers has been generated and an analysis indicates the following combinations and frequencies.

Combination (i)	Observed frequency (Oi)
Four different digits	560
One pair	394
Two pairs	32
Three digits of a kind	13
Four digits of a kind	1
	1000

Based on poker test, test whether these numbers are independent. Use α = 0.05 and $N = 4$ is 9.49.

2. Differentiate between dynamic physical models and static physical models with example.
3. Why do we perform the analysis of simulation output? Explain how do you use simulation run statistics in the output analysis?

Group B

Attempt any eight: (8x5=40)

4. Describe different types of statements, used in CSMP, with suitable examples.
5. Name the entities, attributes, activities, events, and state variables for the following systems:
   a. Cafeteria
   b. Inventory
   c. Banking
   d. A hospital emergency room
   e. Communication.
6. What are the types of simulation models?
7. Differentiate between clock time and simulation time used in system simulation.
8. Describe the importance of differential/Partial differential equations in simulation.
9. Write a computer program in C that will generate four digit random numbers using the multiplicative congruential method. Allow the user to input values of X_0, a, c and m.
10. Describe the basic nature of simulation in brief.
11. When is simulation appropriate and when it is not?
12. What do you mean by Multi Server Queues?
13. Write short note on:
    a. Replication of Runs
    b. Simulation tools

MODEL SET 4

Group A

Attempt any two questions: (2x10=20)

1. A small shop has one checkout counter. Customers arrive at this checkout counter at random from 1 to 10 minutes apart. Each possible value of inter-arrival time has the same probability of occurrence equal to 0.1. Service times vary from 1 to 6 mins with the probability shown below:

Service time	1	2	3	4	5	6
Probability	0.05	0.1	0.2	0.03	0.25	0.10

Develop simulation table for 10 customers. Find average waiting time, average service time, average time customer spends in system.

Take the random digits for arrival as 91, 72, 15, 94, 30, 92, 75, 23, 30 and for service times are 84, 10, 74, 53, 17, 79, 91, 67, 89, 38 sequentially.

2. Explain the three step approach for validation process as formulated by Nayler and Finger.
3. With a neat flow diagram, explain the steps in simulation study.

Group B

Attempt any eight: (8x5=40)

4. Explain the types of simulation with respect to the output analysis. Give examples.
5. Explain the confidence interval estimation method in brief.
6. With neat diagram explain model building verification and validation.
7. Explain linear congruential method. Write three ways of achieving maximal period.
8. Explain the characteristics of queuing system. Explain about Kendall's notation.
9. Explain inverse transform technique of producing random variates for exponential distribution.
10. Explain discrete random variables and continuous random variables, with examples.
11. Explain simulation in GPSS, with block diagram for the single server queue simulation.
12. List any five circumstances when the simulation is the appropriate tool and when it is not.
13. Write short notes on:
    a. Hybrid Simulation
    b. Use of partial differential equation in simulation model.

MODEL SET 5

Group A

Attempt any two questions: (2x10=20)

1. Define and develop a Poker test for four-digit random numbers. A sequence of 10,000 random numbers, each of four digits has been generated. The analysis of the numbers reveals that in 5120 numbers all four digits are different, 4230 contain exactly one pair of like digits, 560 contain two pairs, 75 have three digits of a kind and 15 contain all like digits. Use Poker test to determine whether these numbers are independent> (Critical value of chi-square test for $\alpha = 0.05$ and $N = 4$ is 9.49).
2. Explain the steps in simulation study. What are the limitations of simulation?
3. Describe the linear congruential method for random number generation. Explain about combined linear congruential method. Use the Multiplicative congruential method to generate a sequence of four-three digit random integers, with seed = 117, constant multiplier = 43 and modulus = 1000.

Group B

Attempt any eight: (8x5=40)

4. Explain, how do you update the clock time in system simulation?
5. Explain about acceptance rejection method for random variate.
6. Briefly explain about the characteristics of queuing system.
7. Explain about advantages and disadvantages of simulation.
8. With an example explain markov's chain.
9. Explain about the components of verification and validation of model. Explain about iterative process about calibration of model.
10. Explain about structural and data assumptions in verification and validation of simulation model.
11. A machine tool in manufacturing shop is turning out parts at a rate of one every 5 minutes. As they are finished, the parts go to an inspection, who takes 4 + - 3 minutes to examine each one and reject about 10% of the parts. Represent this scenario in GPSS.
12. Why the output analysis of simulation methods is different from traditional statistical methods of analysis.
13. Discuss about elimination of internal bias.

MODEL SET 6

Group A

Attempt any two questions: (2x10=20)

1. A computer technical support center is staffed by two people, Able and Baker, who take calls and try to answer questions and solve computer problems. The time between calls ranges from 1 to 4 minutes apart with the distribution as shown in the below table 1. Able is more experienced and can provide service faster than baker, which means that when both are idle, Able takes call. The distribution of their service times are shown in table 2 and table 3 respectively.

Table 1: Service time distribution of Able

IAT	1	2	3	4
Probability	0.25	0.4	0.2	0.15

Table 2: Inter arrival time distribution

Service time	2	3	4	5
Probability	0.3	0.28	0.25	0.17

Table 3: Service time distribution of Baker

Service time	3	4	5	6
Probability	0.35	0.25	0.20	0.2

Random digits for inter-arrival times are:
26, 98, 90, 26, 42, 74, 80, 68, 22, 48, 34, 45 ,24 ,34.
Random digits for service time are:
95, 21, 51, 92, 89, 38, 13, 61, 50, 49, 39, 53, 88, 01, 31.
Simulate this for 10 customers, by finding
1. Average waiting time for a customer.
2. Average inter arrival time
3. Average service time for able.
4. Average service time of baker.
5. Average waiting time of those who wait.

2. Define the queuing system. Explain the elements of queuing system with example.
3. Define physical model. Explain the dynamic physical model with the help of suitable diagrams and expressions.

Group B

Attempt any eight questions: (8x5=40)

4. Explain any four program control statements that are used in GPSS.
5. Differentiate between clock time and simulation time used in system simulation.
6. Describe the process of calibration and validation in detail with example.
7. How do you use estimation method in the analysis of simulation output? Explain in brief.
8. How do you eliminate the effect of transient and initial bias in simulation output?

9. "To simulate is to experiment". Justify it.
10. What do you mean by non-uniform random number?
11. What are pseudo random numbers? What are the problems that occur while generating random number?
12. Define the following terms:
    a. Entities
    b. Attributes
    c. Activities
    d. Events
    e. State
13. Explain about distributed lag model and real time simulation.

MODEL SET 7

Group A

Attempt any two questions: (2x10=20)

1. What do you understand by analog method of system simulation? Explain it with suitable example.
2. Six dump trucks are used to haul coal from entrance of a small mine to the rail road. Each truck is loaded by one of the two loaders. After loading truck immediately moves to the scale to be weighed as soon as possible. Both loaders and scale have FCFS for trucks. Travel time from loader to scale is negligible. After being weighed, a truck begins a travel time and then returns to loader queue. Simulate for clock = 20. Find average loader utilization and average scale utilization. The activity times are given in the following table.

Loading time	10	5	5	10	15	10	10
Weighing time	12	12	12	16	12	16	
Traveling time	60	100	40	40	80		

3. Define congestion. Describe different types of components, characteristics and queueing disciplines of a queueing system.

Group B

Attempt any eight questions: (8x5=40)

4. Nametheentities,attributes,activities,events,andstatevariablesforthefollowing systems:
    a. Cafeteria
    b. Inventory
    c. Banking

    d. A hospital emergency room
    e. Communication.
5. Explain non-uniform random number generation.
6. Explain the data and control statement in CSMP.
7. What do you mean by Multi Server Queues?
8. Explain Hybrid simulation with example.
9. Differentiate between analytical models and numerical models.
10. Explain the replication of runs.
11. Write short note on:
    a. Discrete systems modeling
    b. Feedback systems
12. Differentiate between clock time and simulation time used in system simulation.
13. Describe the process of calibration and validation in detail with example.

MODEL SET 8

Group A

Attempt any two questions: (2x10=20)

1. A grocery store has one checkout counter. Customers arrive at this checkout counter at random from 1 to 8 minutes apart and each interval time has the same probability of occurrence. The service times vary from 1 to 6 minutes with probability given below:

Services(minutes)	1	2	3	4	5	6
probability	0.10	0.2	0.3	0.25	0.1	0.05

Simulate the arrival of 6 customers and calculate:

a. Average waiting time for a customer
b. Probability that a customer has to wait
c. Probability of a server being idle
d. Average service time

Use the following sequence of random numbers:

Random digit for arrival: 913, 727, 015, 948, 309, 922

Random digit for service time: 84, 10, 74, 53, 17, 79

Assume that the first customer arrives at time 0.Depict the simulation in a tabular form.

2. Use the chi-square test with a = 0.05 to test whether the data shown below are uniformly distributed. 0.34, 0.83, 0.96, 0.47, 0.79, 0.99, 0.37, 0.72, 0.06, 0.18, 0.90, 0.76, 0.99, 0.30, 0.71, 0.17, 0.51, 0.43, 0.39,0.26 0.25 0.79 0.77,

0.17 0.23 0.99 0.54 0.56 0.84 0.97 0.89 0.64, 0.67 0.82 0.19 0.46 0.01 0.97 0.24
0.88 0.87 0.70 0.56 0.56 0.82 0.05 0.81 0.30 0.40 0.64 0.44 0.81 0.41 0.05 0.93
0.66 0.28 0.94 0.64 0.47 0.12 0.94 0.52 0.45 0.65 0.10 0.69 0.96 0.40 0.60 0.21
0.74 0.73 0.31 0.37 0.42 0.34 0.58 0.19 0.11 0.46 0.22 0.99 0.78 0.39 0.18 0.75
0.73 0.79 0.29 0.67 0.74 0.02 0.05 0.42 0.49, 0.49 0.05 0.62 0,78 .

3. Naylor and Finger have proposed three step approach as an aid in validation process which has been widely followed. Explain in detail.

Group B

Attempt any eight questions: **(8x5=40)**

4. Use the linear congruential method to generate a sequence of random numbers with $X_0 = 27$, $a = 17$, $c = 43$, and $m = 100$. Here, the integer values generated will all be between zero and 99 because of the value of the modulus. These random integers should appear to be uniformly distributed the integers zero to 99.
5. Discuss about the types of errors that can arise while generating random numbers. Discuss about the properties of good random number generator.
6. What is the difference between verification and validation? Explain about structural and data assumptions.
7. Discuss about replication of runs.
8. A machine tool in manufacturing shop is turning out parts at a rate of one every 5 minutes. As they are finished, the parts go to an inspection, who takes 4 + - 3 minutes to examine each one and reject about 10% of the parts. Represent this scenario in GPSS.
9. Discuss about the different types of language one can use for simulation.
10. Discuss about elimination of internal bias.
11. What is meant by model? Discuss its types with example of each.
12. Explain with example of calibration and validation of model.
13. Explain, how do you update the clock time in system simulation?

❑❑❑

System Analysis and Design

Course Title: System Analysis and Design
Course No: CSC315 — Full Marks: 60 + 20 + 20
Nature of the Course: Theory + Lab — Pass Marks: 24 + 8 + 8
Semester: V — Credit Hrs: 3

Course Description: This course familiarizes students with the concepts of information systems development including systems development life cycle, different approaches to systems development, project management, planning, analysis, design, implementation and maintenance. This course also covers some fundamental concepts of object oriented systems analysis and design

Course Objectives: The main objective of this course is to provide knowledge of different concepts of system analysis and design so that students will be able to develop information systems using different methodologies, tools, techniques, and approaches.

Course Contents:

Unit 1: Foundations for Systems Development **(10 Hrs.)**

1.1. The Systems Development Environment: Introduction; A Modern Approach to Systems Analysis and Design; Developing Information Systems and the Systems Development Life Cycle; The Heart of the Systems Development Process and Traditional Waterfall SDLC; CASE Tools

1.2. Other Approaches: Prototyping; Spiral; Rapid Application Development; Introduction to Agile Development

1.3. Managing the Information Systems Project: Introduction; Managing the Information Systems Project; Representing and Scheduling Project Plans; Using Project Management Software.

Unit 2: Planning **(5 Hrs.)**

2.1. Identifying and Selecting Systems Development Projects: Introduction; Identifying and Selecting Systems Development Projects; Corporate and Information Systems Planning

2.2. Initiating and Planning Systems Development Projects: Introduction; Initiating and
Planning Systems Development Projects; Process of Initiating and Planning IS Development Projects, Assessing Project Feasibility; Building and Reviewing the Baseline Project Plan.

Unit 3: Analysis **(13 Hrs.)**

3.1. Determining System Requirements: Introduction; Performing Requirements Determination; Traditional Methods for Determining Requirements; Contemporary Methods for Determining System Requirements; Radical Methods for Determining System Requirements.

3.2. Structuring System Process Requirements: Introduction; Process Modeling; Data Flow Diagrams; Modeling Logic with Decision Tables, Decision Trees, and Pseudo codes.

3.3. Structuring System Data Requirements: Introduction; Conceptual Data Modeling; Gathering Information for Conceptual Data Modeling; Introduction to E-R Modeling.

Unit 4: Design (7 Hrs.)

4.1. Designing Databases: Introduction; Database Design; Relational Database Model; Normalization; Transforming E-R Diagrams into Relations; Merging Relations; Physical File and Database Design; Designing Fields; Designing Physical Tables.

4.2. Designing Forms and Reports: Introduction; Designing Forms and Reports; Formatting Forms and Reports; Assessing Usability.

4.3. Designing Interfaces and Dialogues: Introduction; Designing Interfaces and Dialogues; Interaction Methods and Devices; Designing Interfaces; Designing Dialogues; Designing Interfaces and Dialogues in Graphical Environments

Unit 5: Implementation and Maintenance (4 Hrs.)

5.1. System Implementation: Introduction, System Implementation, Software Application Testing, Installation, Documenting the System, Training and Supporting Users, Organizational Issues in Systems Implementation.

5.2. Maintaining Information Systems: Introduction, Maintaining Information Systems, Conducting Systems Maintenance.

Unit 6: Introduction to Object-Oriented Development (6 Hrs.)

Basic Characteristics of Object-Oriented Systems; Object-Oriented System Analysis and Design (OOSAD); Introduction to Unified Modeling Language, Structural and Behavioral Diagrams

Laboratory / Project Work: In the practical session, students will learn to use project management, CASE, and modeling tools. They also prepare a project report that includes at least analysis, design, and implementation phases of system analysis and design. The project can be done in groups with at most four members in each group using any suitable database, programming, and interfacing technologies.

Text Books:

1. Joseph S. Valacich and Joey F. George, Modern Systems Analysis and Design, 8th Edition, Pearson
2. Alan Dennis, Barbara Haley Wixom, and David Tegarden, Systems Analysis and Design – An Object-Oriented Approach with UML, 5th Edition, Wiley

References Books:

1. Kenneth E. Kendall and Julie E. Kendall, System Analysis and Design, 9th Edition, Pearson
2. Jeffrey Whitten and Lonnie Bently, System Analysis and Design Methods, 7th Edition
3. Scott Tilley and Harry J. Rosenblatt, System Analysis and Design, 11th Edition

TRIBHUVAN UNIVERSITY
Institution of Science and Technology

TU QUESTIONS-ANSWERS 2076

Bachelor Level/Third Year/Fifth Semester/Science
Course Title: System Analysis and Design — Full Marks: 60
Course Code: CSC 315 — Pass Marks: 24
Time: 3 hrs.

Candidates are required to give their answers in their own words as far as practicable.
The figures in the margin indicate full marks.

Section A

Attempt any two questions. (2 × 10 = 20)

1. What is waterfall model? Explain prototyping model for developing information systems along with merits and demerits. (2+8)

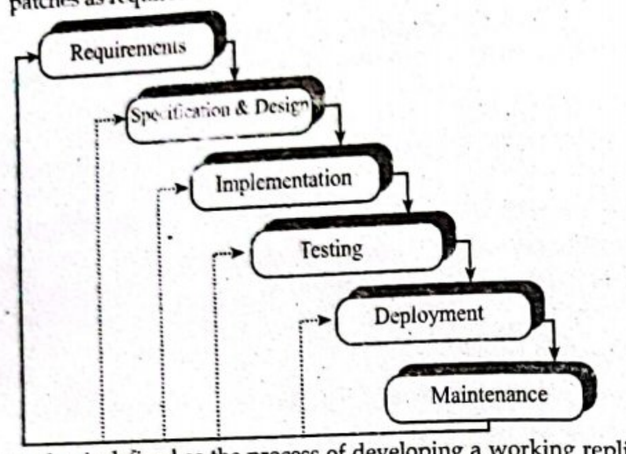
Ans: The Waterfall Model was the first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

The waterfall model is a classical model used in system development life cycle to create a system with a linear and sequential approach. It is termed as waterfall because the model develops systematically from one phase to another in a downward fashion. This model is divided into different phases and the output of one phase is used as the input of the next phase. Every phase has to be completed before the next phase starts and there is no overlapping of the phases.

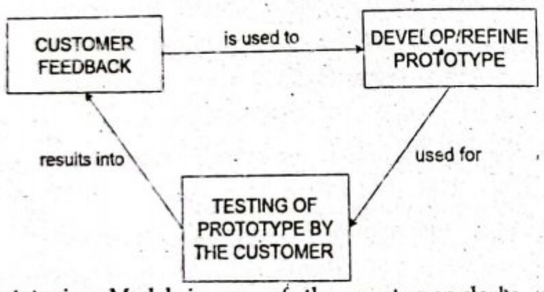
The sequential phases described in the Waterfall model are:

1. **Requirement Gathering-** All possible requirements are captured in product requirement documents.
2. **Analysis -** Read the requirement and based on analysis define the schemas, models and business rules.
3. **System Design –** Based on analysis design the software architecture.
4. **Implementation-** Development of the software in the small units with functional testing.
5. **Integration and Testing-** Integrating of each unit developed in previous phase and post integration test the entire system for any faults.
6. **Deployment of system -** Make the product live on production environment after all functional and nonfunctional testing completed.

7. **Maintenance-** Fixing issues and release new version with the issue patches as required.



Prototyping is defined as the process of developing a working replication of a product or system that has to be engineered. It offers a small scale facsimile of the end product and is used for obtaining customer feedback as described below:



The Prototyping Model is one of the most popularly used Software Development Life Cycle Models (SDLC models). This model is used when the customers do not know the exact project requirements beforehand. In this model, a prototype of the end product is first developed, tested and refined as per customer feedback repeatedly till a final acceptable prototype is achieved which forms the basis for developing the final product.

In this process model, the system is partially implemented before or during the analysis phase thereby giving the customers an opportunity to see the product early in the life cycle. The process starts by interviewing the customers and developing the incomplete high-level paper model. This document is used to build the initial prototype supporting only the basic functionality as desired by the customer. Once the customer figures out the problems, the prototype is further refined to eliminate them. The process continues until the user approves the prototype and finds the working model to be satisfactory.

There are four types of models available

A) Rapid Throwaway Prototyping

This technique offers a useful method of exploring ideas and getting customer feedback for each of them. In this method, a developed prototype

need not necessarily be a part of the ultimately accepted prototype. Customer feedback helps in preventing unnecessary design faults and hence, the final prototype developed is of better quality.

B) Evolutionary Prototyping –

In this method, the prototype developed initially is incrementally refined on the basis of customer feedback till it finally gets accepted. In comparison to Rapid Throwaway Prototyping, it offers a better approach which saves time as well as effort. This is because developing a prototype from scratch for every iteration of the process can sometimes be very frustrating for the developers.

C) Incremental Prototyping

In this type of incremental Prototyping, the final expected product is broken into different small pieces of prototypes and being developed individually. In the end, when all individual pieces are properly developed, then the different prototypes are collectively merged into a single final product in their predefined order. It's a very efficient approach that reduces the complexity of the development process, where the goal is divided into sub-parts and each sub-part is developed individually. The time interval between the project's beginning and final delivery is substantially reduced because all parts of the system are prototyped and tested simultaneously. Of course there might be the possibility that the pieces just do not fit together due to some lack of ness in the development phase – this can only be fixed by careful and complete plotting of the entire system before prototyping starts.

D) Extreme Prototyping

This method is mainly used for web development. It is consists of three sequential independent phases:

- In this phase a basic prototype with all the existing static pages are presented in the HTML format.
- In the 2nd phase, Functional screens are made with a simulated data process using a prototype services layer.
- This is the final step where all the services are implemented and associated with the final prototype.

This Extreme Prototyping method makes the project cycling and delivery robust and fast, and keeps the entire developer team focus centralized on products deliveries rather than discovering all possible needs and specifications and adding un-necessitated features.

Advantages of Prototype model

- Users are actively involved in the development
- Since in this methodology a working model of the system is provided, the users get a better understanding of the system being developed
- Errors can be detected much earlier.
- Quicker user feedback is available leading to better solutions.
- Missing functionality can be identified easily
- Confusing or difficult functions can be identified
- Requirements validation, Quick implementation of, incomplete, but
- functional, application.

Disadvantages of Prototype model

- Leads to implementing and then repairing way of building systems.
- Practically, this methodology may increase the complexity of the system as scope of the system may expand beyond original plans.
- Incomplete application may cause application not to be used as the full system was designed
- Incomplete or inadequate problem analysis.

2. Define feasibility. Explain different categories of feasibility. How do you measure economic feasibility? (2+8)

Ans: Feasibility Study can be considered as preliminary investigation that helps the management to take decision about whether study of system should be feasible for development or not.

- It identifies the possibility of improving an existing system, developing a new system, and produce refined estimates for further development of system.
- It is used to obtain the outline of the problem and decide whether feasible or appropriate solution exists or not.
- The main objective of a feasibility study is to acquire problem scope instead of solving the problem.
- The output of a feasibility study is a formal system proposal act as decision document which includes the complete nature and scope of the proposed system.

Types of Feasibilities

Economic Feasibility

It is evaluating the effectiveness of candidate system by using cost/benefit analysis method. It demonstrates the net benefit from the candidate system in terms of benefits and costs to the organization. The main aim of Economic Feasibility Analysis (EFS) is to estimate the economic requirements of candidate system before investments funds are committed to proposal.

It prefers the alternative which will maximize the net worth of organization by earliest and highest return of funds along with lowest level of risk involved in developing the candidate system.

Technical Feasibility

It investigates the technical feasibility of each implementation alternative. It analyzes and determines whether the solution can be supported by existing technology or not. The analyst determines whether current technical resources be upgraded or added it that fulfill the new requirements. It ensures that the candidate system provides appropriate responses to what extent it can support the technical enhancement.

Operational Feasibility

It determines whether the system is operating effectively once it is developed and implemented. It ensures that the management should support the proposed system and its working feasible in the current organizational environment. It analyzes whether the users will be affected and they accept the modified or new business methods that affect the possible system benefits.

It also ensures that the computer resources and network architecture of candidate system are workable.

Behavioral Feasibility

It evaluates and estimates the user attitude or behavior towards the development of new system. It helps in determining if the system requires special effort to educate, retrain, transfer, and changes in employee's job status on new ways of conducting business.

Schedule Feasibility

It ensures that the project should be completed within given time constraint or schedule. It also verifies and validates whether the deadlines of project are reasonable or not.

3. Assuming a retail clothing store in a mall, draw a context diagram and a level-0 diagram that represent the selling system at the store. (4+6)

Ans: A suggested context diagram and level-0 diagram are provided below.

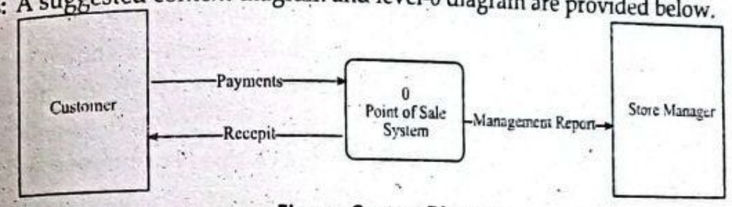


Figure: Context Diagram

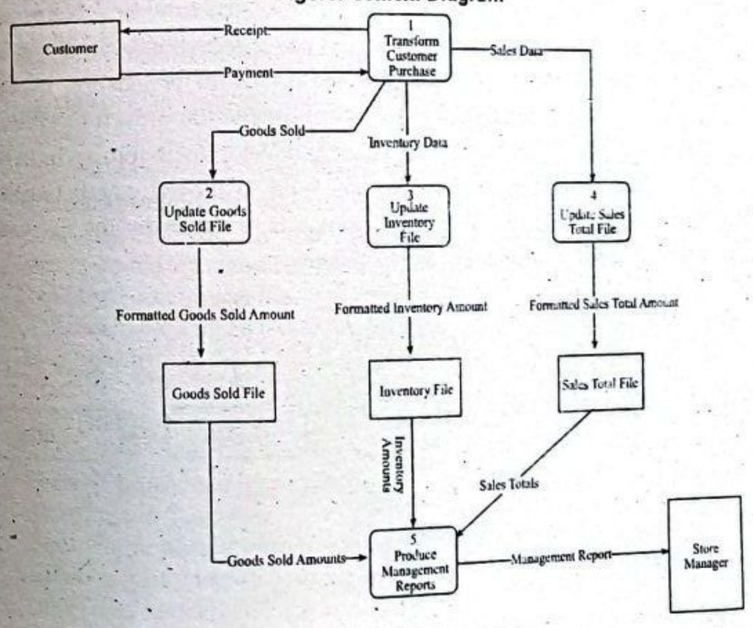


Figure: DFD level 0 Diagram

Section B

Attempt any eight questions. (8 × 5 = 40)

4. Explain modern approaches to system analysis and design. (5)

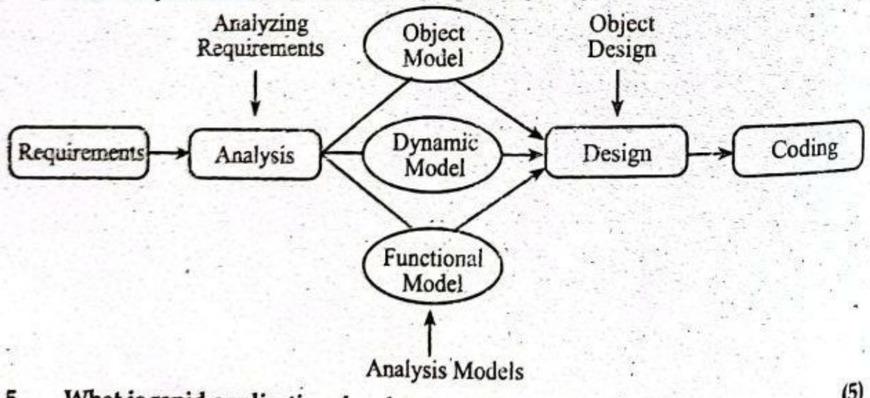
Ans: Agile methodologies

The Agile methodology is a way to manage a project by breaking it up into several phases. It involves constant collaboration with stakeholders and continuous improvement at every stage. Once the work begins, teams cycle

through a process of planning, executing, and evaluating. Continuous collaboration is vital, both with team members and project stakeholders.

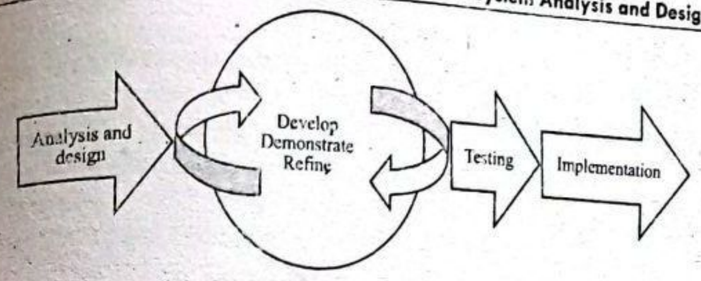
Object oriented analysis and design (OOAD)

The object oriented approach looks at a system from a bottom-up view. It combines data and processes (methods) into objects. Within an information system, objects could be customers, suppliers, contracts, and rental agreements. A set of diagrams or models is used to represent various views and functionality of the system and is commonly known as Unified Modeling Language (UML). The OO approach later becomes known as the unified process when these models are used along with a particular method of systems development. Unified process is an iterative and incremental approach to systems development.4 The goal of OOAD is to improve system quality and productivity of systems analysis and design by making it more usable. Objects are grouped into classes to share structural and behavioral characteristics. OOAD also incorporates the use of inheritance; it allows the creation of new classes that share the characteristics of existing classes. Similar to the agile methodologies, the object-oriented approach to systems development is similar in the way of iterative development approach. In the analysis phase, object-oriented models are used to fill the gap between a problem and the solution. The aim, in essence, is to transform the use cases into analysis model to realize the associated goals.



5. What is rapid application development? Explain. (5)

Ans: RAD Model or Rapid Application Development model is a software development process based on prototyping without any specific planning. In RAD model, there is less attention paid to the planning and more priority is given to the development tasks. It targets at developing software in a short span of time.



SDLC RAD modeling has following phases

- Business Modeling
- Data Modeling
- Process Modeling
- Application Generation
- Testing and Turnover

Business Modelling

The business model for the product under development is designed in terms of flow of information and the distribution of information between various business channels. A complete business analysis is performed to find the vital information for business, how it can be obtained, how and when is the information processed and what are the factors driving successful flow of information.

Data Modelling

The information gathered in the Business Modeliing phase is reviewed and analyzed to form sets of data objects vital for the business. The attributes of all data sets is identified and defined. The relation between these data objects are established and defined in detail in relevance to the business model.

Process Modelling

The data object sets defined in the Data Modelling phase are converted to establish the business information flow needed to achieve specific business objectives as per the business model. The process model for any changes or enhancements to the data object sets is defined in this phase. Process descriptions for adding, deleting, retrieving or modifying a data object are given.

Application Generation

The actual system is built and coding is done by using automation tools to convert process and data models into actual prototypes.

Testing and Turnover

The overall testing time is reduced in the RAD model as the prototypes are independently tested during every iteration. However, the data flow and the interfaces between all the components need to be thoroughly tested with complete test coverage. Since most of the programming components have already been tested, it reduces the risk of any major issues.

6. What is project initiation? Explain different activities you will perform during project initiation phase. (1+4)

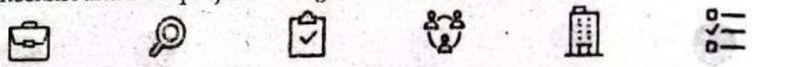
Ans: Project initiation is the first phase of the project management life cycle and in this stage, companies decide if the project is needed and how beneficial it will be for them. The two metrics that are used to judge a proposed project and determine the expectations from it are the business case and feasibility study.

Why is it important?

- Take major decisions that establish the direction and resource requirements, like the project charter and selecting the project stakeholders, are made during this phase. The stakeholders arrive at a clear objective to ensure everyone stays on the same page in terms of how the project should proceed.
- There will be multiple checks during and after project execution to prevent miscommunication and to ensure the project stays on track throughout its course. However, precious time and resources might get wasted which is undesirable.
- Effective project management requires you to maximize benefits and minimize costs while delivering 'value' to the customer. Having a clear project objective helps you achieve all this.

Project initiation process – 6 key steps to follow

Now that we have established what project initiation is and why it is so important, it's time to see what are the key steps in the project initiation checklist and how project managers initiate their projects.



Business Case	Feasibility Study	Project Charter	Team	Project Office	Review
Explain why the project is necessary and how it will succeed.	Research the reason for the project and determine if it will succeed.	How will the project be structured and executed?	Find the people with the right skills and experience to execute the project.	Where the project manager and support staff are located to assist with projects.	Review the initiation phase and keep treviewing progress throughout the project.

Figure: Project Initiation process

1. **Creating a business case**

   The business case is an important document that explains how the project's goals align with the company's long-term plans. This document explains why should the company spend its technical, financial, and human resources on the specific project.

   An ideal business case does not talk about any technical details of the project and focuses solely on the business aspects. It is made to convince the upper management to approve the project and answer their concerns related to possible financial and business-related risks.

2. **Conducting a feasibility study**

   After the approval of the business case, the next step is to determine the likelihood of the project's success after considering all the factors. This study identifies the high-level project constraints and assumptions of the project and decides whether the project is worth it or not.

3. **Establishing a project charter**

   The project charter is perhaps the most comprehensive and important part of the project initiation process. It answers to identify the scope/objective, team members, and the possible timeframe of the project.

   The charter is, in some ways, the first document of the project that identifies the necessary details like the goals and the constraints of the project. It also identifies the project scope and lists the required resources for the completion of the project.

4. **Identifying stakeholders and making a stakeholder register**

   Communication and negotiations are a huge part of effective project management and a large part of a project manager's time is usually spent dealing with project stakeholders. Project stakeholders can either be internal or external and each type has its own communication requirement.

   It's the responsibility of the project manager to ensure the means and frequency of communication in project management with project stakeholders according to their influence and interest in the project. A common practice is to maintain a stakeholder register or a stakeholder map to decide the frequency and means of communication for each stakeholder according to their influence and interest in the project.

5. **Assembling the team and establishing a project office**

   No project can be started without a project team. Assembling a working project team and assigning them roles and responsibilities is a vital part of the project initiation phase. Assigning roles and responsibilities early on also increases the overall accountability of the entire team and can help you as a manager in the later phases of the project life cycle.

6. **Final review**

   After performing everything, it's a good practice to review the entire project initiation stage to ensure you missed nothing. In later stages, you'll continue reviewing your work as monitoring and controlling is one of the five phases of the project management life cycle.

7. **Explain the process of identifying and selecting information system development project in brief.** (5)

Ans: Project identification and selection consists of three primary activities: identifying potential development projects, classifying and ranking projects, and selecting projects for development.

1. **Identifying potential development projects.** Organizations vary as to how they identify projects. This process can be performed by:
   - A key member of top management, either the CEO of a small or medium-size organization or a senior executive in a larger organization
   - A steering committee, composed of a cross section of managers with an interest in systems
   - User departments, in which either the head of the requesting unit or a committee from the requesting department decides which projects to submit (as a systems analyst, you will help users prepare such requests)
   - The development group or a senior IS manager
2. **Classifying and ranking IS development projects**

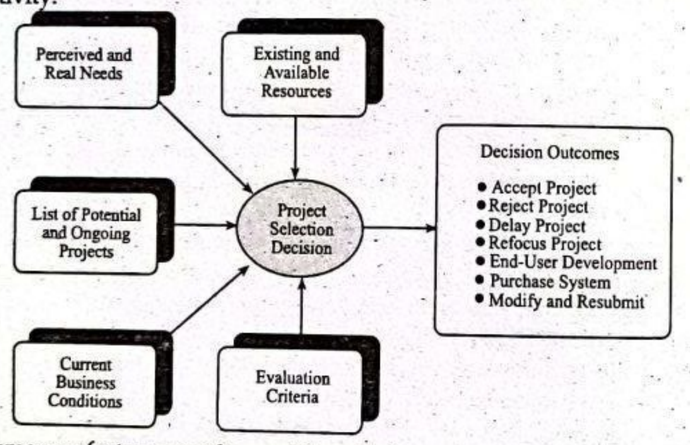
   Assessing the merit of potential projects is the second major activity in the project identification and selection phase. As with project identification,

classifying and ranking projects can be performed by top managers, a steering committee, business units, or the IS development group. The criteria used to assign the merit of a given project can vary based on the size of the organization. In any given organization, one or several criteria might be used during the classifying and ranking process

As with project identification, the criteria used to evaluate projects will vary by organization. If, for example, an organization uses a steering committee, it may choose to meet monthly or quarterly to review projects and use a wide variety of evaluation criteria. At these meetings, new project requests are reviewed relative to projects already identified, and ongoing projects are monitored. The relative ratings of projects are used to guide the final activity of this identification process—project selection.

3. **Selecting IS development projects**

The selection of projects is the final activity in the project identification and selection phase. The short- and long-term projects most likely to achieve business objectives are considered. As business conditions change over time, the relative importance of any single project may substantially change. Thus, the identification and selection of projects is an important and ongoing activity.



Numerous factors must be considered when selecting a project, as illustrated in Figure above. These factors include:

- Perceived needs of the organization
- Existing systems and ongoing projects
- Resource availability
- Evaluation criteria
- Current business conditions
- Perspectives of the decision makers

8. **What is group interview? What are the benefits and drawbacks of group interview?** (2+3)

Ans: Group interview can be defined as a interview process where in more than two people or multiple employees are interviewed at the same time simultaneously. There are two types of group interviews--group and panel. A group interview consists of a single interviewer interviewing multiple candidates at the same time. Group interviews are most common in industries like food service, hospitality and retail.

Panel interviews, on the other hand, consists of a panel of multiple members of a hiring team interviewing a single job candidate. The interview group usually includes the hiring managers, relevant team members, and an HR representative.

Advantages of Group Interviews:

There are several advantages and benefits of group interviews and in that bucket list of benefits of group interviews, some can be named very much essential for the growth of the overall activities of the company.

1. **A quick selection:** Most of the time while attending a group interview, employers get an idea of the perfect candidate for the position they are offering. And with the help of this group interview, employers can eliminate unwanted candidates who doesn't suit the job profile.
   Moreover, this group interview method reduces time of unwanted conversation with an irrelevant candidate who have applied for the job position.
   Therefore, group interviews are quite a quick and possible solution for quick and easy selection of the perfect candidate.
2. **Maximum of candidates can be interviewed:** As this process is called as a group interview it is quite obvious that in this stage of the interview the employer can interview a group of candidates for the position.
   And at the end of this interview session the employer can find the suitable candidate for the position offered.
   Moreover, it creates some sort of time constraints for the candidates, but for instance, it is beneficial for the employer because the employer doesn't need to waste his or her energy by interviewing each and every candidate separately.
3. **Creates a background for discussion:** It is completely explained to everyone that group interview is based on a background of discussion where in which discussion would be about necessary elements which affects the growth of the company.
   Most of the time in the middle of such interviews the employer expects such candidates who can prove themselves as an asset to the company.
   Therefore, at the end of group interview discussion the candidate with creative ideas will be finalized for the position offered.
4. **Similar pattern of question for discussion:** As it is quite familiar with everyone that during a group interview a group of candidates will be gathered in a small conference room and the employer present all sorts of similar questions to the candidates.
   And at the end of the group interview the employer will come to know the suitable candidate for the position.
   Therefore, it concludes that the group interviews are beneficial for the employer and the candidates as well.
5. **It delivers candidates strong potential:** During group interviews all the candidates present their best personality in front of the interviewing panel.
   Bring in the candidate's best personality during group interviews, so that it can help the employer to make a correct decision.
   It is very much advisable for the candidates attending any sort of interview that they need to be strong during their interviews so that they can deliver their best personality in front of the employer's panel.

6. **It creates better communication level:** As it states that during group interviews the candidates are required to present their opinions on the table of discussion and this table discussion helps all the candidates so that they share their views and opinions with the employer's panel.
   Therefore, it is very much important for the employers that because of the group interviews the candidates build their communication level where in which he or she can deliver their creativity in front of the interviewing panel.
7. **It is helpful for quality control purpose:** Most of the time during group interviews the employers inform all the candidates about the topic on which the candidates are going to present their views on.
   Therefore, in that matter of time the candidate presents their views and this format of discussion will be stored for the purpose of quality control. The quality control provides an example of the company's future activities.
   Moreover, all the candidates who go through such group interview will be exposed to a knowledgeable level of information.
8. **The candidates can show better interest:** It is possible that during group interviews most of the candidates experience some sort of difference of opinion with other candidates and to overcome that situation the candidate need to deliver his or her interest in the interview for better growth.
   Therefore, at the end of the interview process the employer will select a candidate from that group who actually showed greater interest in the discussion.

Disadvantages of Group Interviews:

There are some of the disadvantages when it comes to the group interview and this disadvantages provide better realization of the features and benefits of group interviews. And those disadvantages are as follows.

1. **It creates competition:** As it is quite clear to everyone that during group interviews most of the candidates goes through tough competition in their way of selection. And sometimes these competition even can reduce their chances of getting that position in that particular company.
   Therefore, it is necessary to understand the competition will be healthy and positive if the candidate's participation is healthy and positive enough to handle the situation.
2. **Control issues:** Most of the time during group interviews the situation goes out of hand in the middle of the group discussion and that can create some sort of miscommunication between the candidate and the employer and eventually the candidate can lose his or her chances of being selected for the position offered.
   Therefore, it is very much essential for the employer that before conducting such high level of discussion, he or she need to be ready with controlling measures to face all sorts of issues.
3. **Overpowering qualified candidates:** During group interviews the company will invite most qualified candidates for the position for which they require potential candidates. And because of such group interviews some of the qualified and potential candidates can even lose their chances on the position.
   Therefore, overpowering qualified candidates during group interviews holds a strong evidence on losing potential candidates for the position.

4. **Limited questionnaire:** It is possible that the group interviews can be held on the similar pattern of discussion and that pattern of discussion bares the similar and limited questionnaires.
   These limited questionnaires deliver a situation where in which all the focus points on the subject will be clear for the sake of selection.
   But unfortunately these limited patterns of questionnaire don't help in any sorts of interviews and which eventually leads the employer in finding a perfect candidate for the post.
5. **Conflicts between candidates:** During group interviews it is quite possible that most of the candidates may not agree with the opinion of another candidate who presented his or her point of view.
   And in that process, the candidates might get into some conflict between the candidates and end up being in trouble for themselves.
   Therefore, it is possible that the conflict between candidates can get its serious terms and even can damage their future chances of getting a job.
6. **It is more of a group process skill:** It is not necessary that group process skill is beneficial at all times of the group interviews. The group process skill is a process wherein which all the relevant activities will be measured fot the sake of growth of the company.
   Therefore, this group process skill can be provided with a skill which is suitable only for the growth and benefit of the interviewing company.
   Moreover, all the skills are needed to clear this group interview.
7. **Lack of creative approach:** As it is familiar to everyone that the group interviews are conducted using similar pattern of questionnaires and these questionnaires can be prepared in a similar format with answers.
   Therefore, it lacks the creative approach due to this, moreover this limited approach can lack creativity in it.
   This can eliminate the chances of approaching for the best possible solution in terms of the group interviews.
8. **It eliminates potential candidate's growth:** Most of the candidates who attend a group interviews comes with potential talent and skill for the help of the perfect candidate selection.
   Moreover, it eliminates potential candidate's growth and eventually it grows as a disadvantage for the purpose of the group interviews.

9. **How do you format forms and reports? Explain general guidelines for formatting forms and reports?** (2+3)

Ans: Formatting Forms and Reports:

1. Types of Information.
   a) Internal Information (within an organization).
   b) External Information.
   c) Turnarround document. (delivered to an external customer as an output that can be returned to provide new information as an input to an information system.
2. General Formatting Guidelines.
3. Highlighting Information.
4. Color vs. No-Color
5. Displaying Text
6. Designing Tables and List

Main Guidelines for Report design

1. **Keeping the Reports Simple and Attractive:** Printed reports and screen output should be attractive and easy to read and understand. Good reports design such as should consider the report header and footers, columns heading alignment, column spacing and field order.
   Every report should have a report header; which appears at the beginning of the report, identifies the report and contains report title, date and other necessary information. Report footer appears at the end of the report can include grand total of the numeric fields or any other end-report information.
   Other than that, every report should have page header. Page header appears at the top of the pages and includes column headings that identify the data. Column heading is a caption used referring to a column, and we should space the columns carefully. If the information listed in a report contains more than values, it's good if the detail lines is grouping based on a control field. The report should be easy and simple. The presentation of the reports should be consistent with uniform formats.
2. **Understanding the Reports Usage:** The important principle in designing reports is to understand the report usage. Reports can be used for many purposes. In most cases, reports have been used to identify specific items as references in finding information; so it's needed to classify the items based on categories depending on users' need. This should be applied when designing a web-based or electronic report for information system. These types of reports planned to be read from beginning to the end and should be represented in one long scrollable page.
   If the report is used to find any specific information, it should be broken into multiple pages with a separate link. The frequency of report prepared also affected when designing a good reports such as real time report and batch reports. Real time report is a report which provide data that are accurate and changes in a second or minute such as stock market data meanwhile batch reports contains historical information that may be in month, days and often provide additional information such as total, summaries and others.
3. **Managing Information Load:** Different levels of users need different categories of data with a different amount of data. Most managers need information but they prefer in graphical format, so it's easy for them to understand, summarize and make decision from the graphic. The goal of well-designed report is to provide all the information needed to support the task for which it was designed. This report shouldn't have all the information but only information required by certain users. Some reports display the most important information generally should be presented first in the top of the report. Other way we can use is by highlighting the most important information needed in the report.
4. **Minimizing Bias:** Whatever forms it takes, output is not only just a neutral product that is subsequently analyzed and acted upon by decision makers. Output will affects users in many different ways of how it's presented. Bias is present in everything that humans create. There are three main ways in which presentation of output are unintentionally biased :
   a. How information is sorted
   b. Setting of acceptable limits
   c. Choice of graphics

10. **List major activities of maintenance. Explain different types of maintenance activities.** (1+4)

Ans: System maintenance is an ongoing activity, which covers a wide variety of activities, including removing program and design errors, updating documentation and test data and updating user support. For the purpose of convenience, maintenance may be categorized into three classes, namely:

i) **Corrective Maintenance**: This type of maintenance implies removing errors in a program, which might have crept in the system due to faulty design or wrong assumptions. Thus, in corrective maintenance, processing or performance failures are repaired.

ii) **Adaptive Maintenance**: In adaptive maintenance, program functions are changed to enable the information system to satisfy the information needs of the user. This type of maintenance may become necessary because of organizational changes which may include:
   a) Change in the organizational procedures,
   b) Change in organizational objectives, goals, policies, etc.
   c) Change in forms,
   d) Change in information needs of managers.
   e) Change in system controls and security needs, etc.

iii) **Perfective Maintenance**: Perfective maintenance means adding new programs or modifying the existing programs to enhance the performance of the information system. This type of maintenance undertaken to respond to user's additional needs which may be due to the changes within or outside of the organization. Outside changes are primarily environmental changes, which may in the absence of system maintenance, render the information system ineffective and inefficient. These environmental changes include:
   a) Changes in governmental policies, laws, etc.,
   b) Economic and competitive conditions, and
   c) New technology.

11. **What is object oriented development? How is it different from structured development?** (2+3)

Ans: **Object-oriented Development** (OOD) a group of methodologies that sees real world entities as objects and classes. For example, hospital is a real world entity, becomes hospital class and later multiple hospital objects are created, each with unique property values.

Many benefits are cited for OOD, often to an unrealistic degree. Some of these potential benefits are:-

- **Faster Development:** OOD has long been touted as leading to faster development. Many of the claims of potentially reduced development time are correct in principle, if a bit overstated.
- **Reuse of Previous work:** This is the benefit cited most commonly in literature, particularly in business periodicals. OOD produces software modules that can be plugged into one another, which allows creation of new programs. However, such reuse does not come easily. It takes planning and investment.
- **Increased Quality:** Increases in quality are largely a by-product of this program reuse. If 90% of a new application consists of proven, existing components, then only the remaining 10% of the code has to be tested from scratch. That observation implies an order-of-magnitude reduction in defects.

- **Modular Architecture:** Object-oriented systems have a natural structure for modular design: objects, subsystems, framework, and so on. Thus, OOD systems are easier to modify. OOD systems can be altered in fundamental ways without ever breaking up since changes are neatly encapsulated. However, nothing in OOD guarantees or requires that the code produced will be modular. The same level of care in design and implementation is required to produce a modular structure in OOD, as it is for any form of software development.
- **Client/Server Applications:** By their very nature, client/server applications involve transmission of messages back and forth over a network, and the object-message paradigm of OOD meshes well with the physical and conceptual architecture of client/server applications.
- **Better Mapping to the Problem Domain:** This is a clear winner for OOD, particularly when the project maps to the real world. Whether objects represent customers, machinery, banks, sensors or pieces of paper, they can provide a clean, self-contained implication which fits naturally into human thought processes.

Difference between Structured Programming and Object-Oriented Programming :

Structured Programming	Object-Oriented Programming
1. It is a subset of procedural programming.	1. It relies on concept of objects that contain data and code.
2. Programs are divided into small programs or functions.	2. Programs are divided into objects or entities.
3. It is all about facilitating creation of programs with readable code and reusable components.	3. It is all about creating objects that usually contain both functions and data.
4. Its main aim is to improve and increase quality, clarity, and development time of computer program.	4. Its main aim is to improve and increase both quality and productivity of system analysis and design.
5. It simply focuses on functions and processes that usually work on data.	5. It simply focuses on representing both structure and behavior of information system into tiny or small modules that generally combines data and process both.
6. It is a method of organizing, managing and coding programs that can give or provide much easier modification and understanding.	6. It is a method in which set of objects can vary dynamically and can execute just by acting and reading to each other.
7. In this, methods are written globally and code lines are processed one by one i.e., Run sequentially.	7. In this, method works dynamically, make calls as per need of code for certain time.
8. It generally follows "Top-Down Approach".	8. It generally follows "Bottom-Up Approach".
9. It provides less flexibility and abstraction as compared to object-oriented programming.	9. It provides more flexibility and abstraction as compared to structured programming.

10. It is more difficult to modify structured program and reuse code as compared to object-oriented programs.	10. It is less difficult to modify object-oriented programs and reuse code as compared to structured programs.
11. It gives more importance of code.	11. It gives more importance to data.

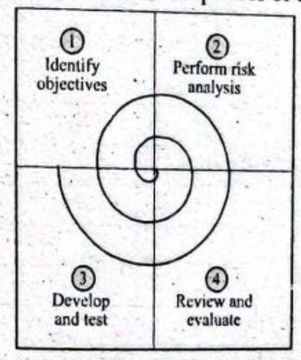
12. Write short notes on:

a) Spiral model **(2.5+2.5)**

Ans: Spiral model is one of the most important Software Development Life Cycle models, which provides support for Risk Handling. In its diagrammatic representation, it looks like a spiral with many loops. The exact number of loops of the spiral is unknown and can vary from project to project. Each loop of the spiral is called a Phase of the software development process. The exact number of phases needed to develop the product can be varied by the project manager depending upon the project risks. As the project manager dynamically determines the number of phases, so the project manager has an important role to develop a product using the spiral model.

The Radius of the spiral at any point represents the expenses(cost) of the project so far, and the angular dimension represents the progress made so far in the current phase.

The below diagram shows the different phases of the Spiral Model:



Each phase of the Spiral Model is divided into four quadrants as shown in the above figure. The functions of these four quadrants are discussed below-

- Objectives determination and identify alternative solutions: Requirements are gathered from the customers and the objectives are identified, elaborated, and analyzed at the start of every phase. Then alternative solutions possible for the phase are proposed in this quadrant.
- Identify and resolve Risks: During the second quadrant, all the possible solutions are evaluated to select the best possible solution. Then the risks associated with that solution are identified and the risks are resolved using the best possible strategy. At the end of this quadrant, the Prototype is built for the best possible solution.

- Develop next version of the Product: During the third quadrant, the identified features are developed and verified through testing. At the end of the third quadrant, the next version of the software is available.
- Review and plan for the next Phase: In the fourth quadrant, the Customers evaluate the so far developed version of the software. In the end, planning for the next phase is started.

b) Decision table

Ans: Decision Table is just a tabular representation of all conditions and actions. Decision Trees are always used whenever the processing logic is very complicated and involves multiple conditions. The main components used for the formation of the Data Table are Conditions Stubs, Action Stubs, and rules.

Decision Table Example:

Let's take an example scenario for an ATM where a decision table would be of use.

A customer requests a cash withdrawal. One of the business rules for the ATM is that the ATM machine pays out the amount if the customer has sufficient funds in their account or if the customer has the credit granted. Already, this simple example of a business rule is quite complicated to describe in text. A decision table makes the same requirements clearer to understand:

Conditions	R_1	R_2	R_3
Withdrawal Amount <= Balance	T	F	F
Credit granted	-	T	F
Actions			
Withdrawal granted	T	T	F

In a decision table, conditions are usually expressed as true (T) or false (F). Each column in the table corresponds to a rule in the business logic that describes the unique combination of circumstances that will result in the actions. The table above contains three different business rules, and one of them is the "withdrawal is granted if the requested amount is covered by the balance." It is normal to create at least one test case per column, which results in full coverage of all business rules.

TU QUESTIONS-ANSWERS 2078

Bachelor Level/Third Year/Fifth Semester/Science
Course Title: System Analysis and Design — Full Marks: 60
Course Code: CSC 315 — Pass Marks: 24
Time: 3 hrs.

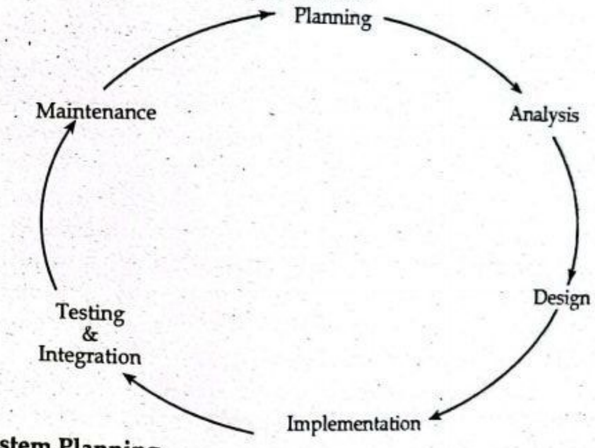
Candidates are required to give their answers in their own words as far as practicable.
The figures in the margin indicate full marks.

Section A

Attempt any two questions. (2 × 10 = 20)

1. What is system development life cycle (SDLC)? Explain each phase of SDLC in detail.

Ans: System Development Life Cycle (SDLC) is a series of six main phases to create a hardware system only, a software system only or a combination of both to meet or exceed customer's expectations. It's a term that can be used in different industries, therefore Software Development Life Cycle is a limited term that explains the phases of creating a software component that integrates with other software components to create the whole system.
System Development Life Cycle Phases:



1- System Planning

The Planning phase is the most crucial step in creating a successful system, during this phase you decide exactly what you want to do and the problems you're trying to solve, by:

- Defining the problems, the objectives and the resources such as personnel and costs.
- Studying the ability of proposing alternative solutions after meeting with clients, suppliers, consultants and employees.
- Studying how to make your product better than your competitors'.

After analyzing this data you will have three choices: develop a new system, improve the current system or leave the system as it is.

2- System Analysis

The end-user's requirements should be determined and documented, what their expectations are for the system, and how it will perform. A feasibility study will be made for the project as well, involving determining whether

it's organizationally, economically, socially, technologically feasible. It's very important to maintain strong communication level with the clients to make sure you have a clear vision of the finished product and its function.

3- System Design

The design phase comes after a good understanding of customer's requirements, this phase defines the elements of a system, the components, the security level, modules, architecture and the different interfaces and type of data that goes through the system. A general system design can be done with a pen and a piece of paper to determine how the system will look like and how it will function, and then a detailed and expanded system design is produced, and it will meet all functional and technical requirements, logically and physically.

4- Implementation and Deployment

This phase comes after a complete understanding of system requirements and specifications, it's the actual construction process after having a complete and illustrated design for the requested system. In the Software Development Life Cycle, the actual code is written here, and if the system contains hardware, then the implementation phase will contain configuration and fine-tuning for the hardware to meet certain requirements and functions.

In this phase, the system is ready to be deployed and installed in customer's premises, ready to become running, live and productive, training may be required for end users to make sure they know how to use the system and to get familiar with it, the implementation phase may take a long time and that depends on the complexity of the system and the solution it presents.

5- System Testing and Integration

Bringing different components and subsystems together to create the whole integrated system, and then introducing the system to different inputs to obtain and analyze its outputs and behavior and the way it functions. Testing is becoming more and more important to ensure customer's satisfaction, and it requires no knowledge in coding, hardware configuration or design. Testing can be performed by real users, or by a team of specialized personnel, it can also be systematic and automated to ensure that the actual outcomes are compared and equal to the predicted and desired outcomes.

6- System Maintenance

In this phase, periodic maintenance for the system will be carried out to make sure that the system won't become obsolete, this will include replacing the old hardware and continuously evaluating system's performance, it also includes providing latest updates for certain components to make sure it meets the right standards and the latest technologies to face current security threats.

2. **Assuming monetary benefits of an information system at $85,000 per year, one-time costs of $75,000, recurring costs of $35,000 per year, a discount rate of 12 percent, and a five year time horizon, calculate the net present value of these costs and benefits of an information system. Also calculate the overall return on investment of the project and then present a break-even analysis. At what point does breakeven occur?** (10)

Ans: **Net Present Value of Benefits:** The net value of benefit will be the sum of overall benefits and will be calculated using following formulae:

$$NPV = PV_1 + PV_2 + PV_3 + PV_4 + PV_5$$
$$= 75,897 + 67,762 + 60,503 + 54,018 + 48,229$$
$$= 306408 \text{ (Cell H5)}$$

Present value (PV) Calculations for Costs: Here, the one-time cost ($ 75,000) is treated as cost occurring in year 0 (now).

$$PV_0 = 75,000 \times \frac{1}{(1+12)^0}$$

Recurring cost ($35,000) happens every year starting at year 1

$$PV_1 = = 35,000 \times \frac{1}{(1+12)^1} = 35,000 \times 0.8929 = 31,252$$

$$PV_2 = 35,000 \times \frac{1}{(1+12)^2} = 35,000 \times 0.7972 = 27,902$$

$$PV_3 = 35,000 \times \frac{1}{(1+12)^3} = 35,000 \times 0.7118 = 24,913$$

$$PV_4 = 35,000 \times \frac{1}{(1+12)^4} = 35,000 \times 0.6355 = 22,243$$

$$PV_5 = 35,000 \times \frac{1}{(1+12)^5} = 35,000 \times 0.5674 = 19,859$$

Net present value of costs,

$$NPV = PV_0 + PV_1 + PV_2 + PV_3 + PV_4 + PV_5$$
$$= 75,000 + 31,252, +27,902 + 24,913 + 22,243 + 19,859$$
$$= 2,01,168 \text{ (Cell H16)}$$

2. Overall Return on Investment ROI)

Overall ROI = (Overall NPV/NPV of All Costs)

Overall NPV = (NPV of All Benefits of All COSTS)
= 3,06,4089 – 2,01,168
= 1,05,240

From the spreadsheet, Overall NPV is 105240 (Cell H18) and NPV of ALL COSTS is 201168 (Cell H16).

ROI = 105,240/201,168
= 0.5231448 (Cell H20)

Consider the given data:

Monetary Benefits of IS = $85000 per year
One-time costs = $75000
Recurring Costs = $ 35,000 (Per year)
Discount rate = 12%
Time period = 5 years

The worksheet given below (figure) shows the present value calculations of all costs and benefits, Break-Even Analysis Overall Return on Investment for this problems using the above data.

The spreadsheet displays the following items.

1. Net Present values (NPV) of all Benefits and costs
2. Overall Return on investment (ROI)
3. Break-Even Analysis (BEA)

1. **Net Present Value of Benefits and Costs:**

Present Value of Benefits of Costs can be calculated using the below formula:

$$PV_n = Y \times \frac{1}{(1+i)^n}$$

Here, PV_n is the present value of Y dollars n years from now, and I is the discount rate.

Present Value (PV) calculations for Benefits:

Benefit start from year 1. So the calculation of PV from year 1 onwards.

$$PV_1 = 85{,}000 \times \frac{1}{(1+12)^1} = 85{,}000 \times 0.8929 = 75{,}897$$

$$PV_2 = 85{,}000 \times \frac{1}{(1+12)^2} = 85{,}000 \times 0.7972 = 67{,}762$$

$$PV_3 = .85{,}000 \times \frac{1}{(1+12)^3} = 85{,}000 \times 0.7118 = 60{,}503$$

$$PV_4 = 85{,}000 \times \frac{1}{(1+12)^4} = 85{,}000 \times 0.6355 = 54{,}018$$

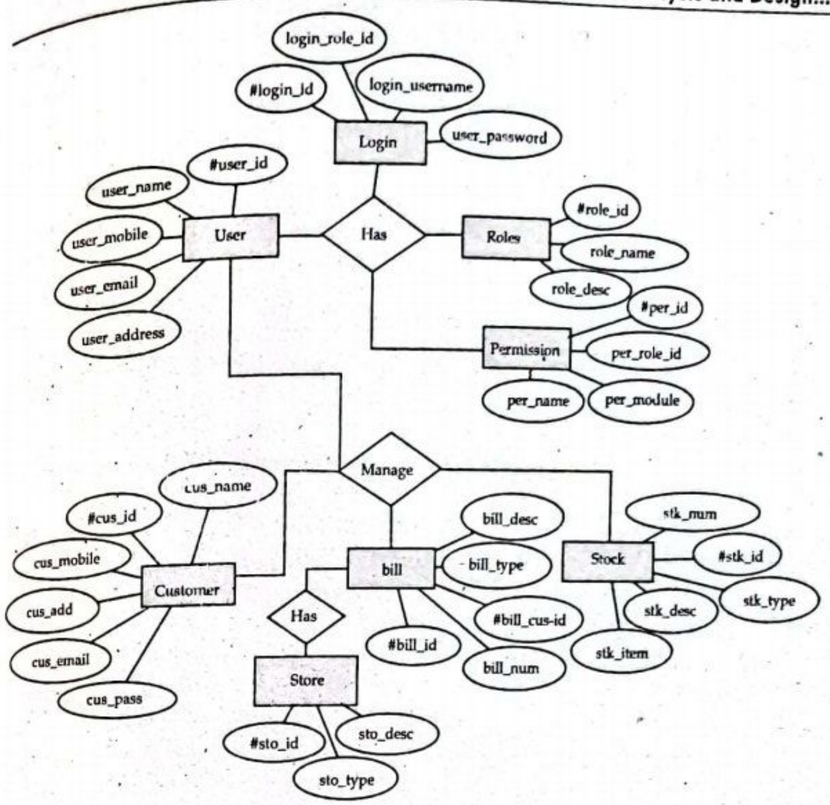
$$PV_5 = 85{,}000 \times \frac{1}{(1+12)^5} = 85{,}000 \times 0.5674 = 48{,}229$$

3. **What is conceptual data model? How do you gather information for conceptual data modeling? Draw an ER diagram for a retail store in a mall which sells different items to its customers.** (2+3+5)

Ans: The conceptual data model is a structured business view of the data required to support business processes, record business events, and track related performance measures. This model focuses on identifying the data used in the business but not its processing flow or physical characteristics. This model's perspective is independent of any underlying business applications. For example, it allows business people to view sales data, expense data, customers, and products—business subjects that are in the integrated model and outside of the applications themselves.

The conceptual data model represents the overall structure of data required to support the business requirements independent of any software or data storage structure. The characteristics of the conceptual data model include:

- An overall view of the structure of the data in a business context.
- Features that are independent of any database or physical storage structure.
- Objects that may not ever be implemented in physical databases. There are some concepts and processes that will not find their way into models, but they are needed for the business to understand and explain what is needed in the enterprise.
- Data needed to perform business processes or enterprise operations.



Section B

Attempt any eight questions. (8 × 5 = 40)

4. **What is system analysis and design? Why is it important for developing information system?**

Ans: System analysis and design deal with planning the development of information systems through understanding and specifying in detail what a system should do and how the components of the system should be implemented and work together. System analysts solve business problems through analyzing the requirements of information systems and designing such systems by applying analysis and design techniques.

Benefits of system analysis and design

The most common benefit of system analysis and design is improving upon a previous system and enjoying increased operational efficiency. Here's a list of other benefits you and your employing organization may enjoy from this practice:

- Enabling comprehension of complicated structures
- Allowing for better management of any business changes
- Aligning the organization with its environment and strategic priorities
- Minimizing IT issues and reducing the workload of IT employees
- Reducing costs in certain areas, saving the organization money and resources for use in other departments
- Identifying potential risks and threats to the processes before they arise
- Improving the overall quality of the system
- Improving the usability of the system by employees
- Increasing productivity and customer satisfaction

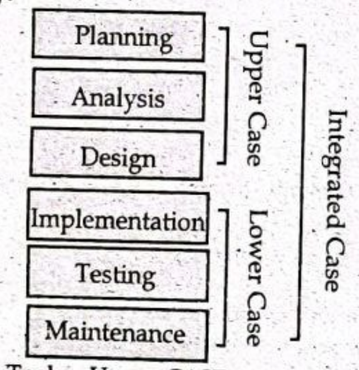
5. What is CASE tool? Explain different components of CASE tool. (1+4)

Ans: CASE stands for Computer Aided Software Engineering. It means, development and maintenance of software projects with help of various automated software tools. CASE tools are set of software application programs, which are used to automate SDLC activities. CASE tools are used by software project managers, analysts and engineers to develop software system. There are number of CASE tools available to simplify various stages of Software Development Life Cycle such as Analysis tools, Design tools, Project management tools, Database Management tools, Documentation tools are to name a few. Use of CASE tools accelerates the development of project to produce desired result and helps to uncover flaws before moving ahead with next stage in software development.

Components of CASE Tools

CASE tools can be broadly divided into the following parts based on their use at a particular SDLC stage:

- **Central Repository** - CASE tools require a central repository, which can serve as a source of common, integrated and consistent information. Central repository is a central place of storage where product specifications, requirement documents, related reports and diagrams, other useful information regarding management is stored. Central repository also serves as data dictionary.



- Upper Case Tools - Upper CASE tools are used in planning, analysis and design stages of SDLC.
- Lower Case Tools - Lower CASE tools are used in implementation, testing and maintenance.
- Integrated Case Tools - Integrated CASE tools are helpful in all the stages of SDLC, from Requirement gathering to Testing and documentation.

CASE tools can be grouped together if they have similar functionality, process activities and capability of getting integrated with other tools.

6. What is project management? Explain some activities and skills of a project manager. (1+4)

Ans: To understand project management, we must look deeper into what constitutes a project. Essentially, projects are temporary efforts to create value through unique products, services, and processes. Some projects are engineered to quickly resolve problems.

Project management is the use of specific knowledge, skills, tools and techniques to deliver something of value to people. The development of

software for an improved business process, the construction of a building, the relief effort after a natural disaster, the expansion of sales into a new geographic market—these are all examples of projects.

Project manager technical skills

Technical skills (you might also hear these referred to as hard skills) are the more tangible and measurable abilities required to be an effective project manager.

1. **Planning and forecasting:** It goes without saying, but proper project management requires skilled planning. This can be challenging, especially since many project managers need to make educated guesses about timelines and required resources.
2. **Risk management:** Every project has risks. Perhaps a resource won't be available when you need it, or delayed approval from a client will set your timeline back a few days.
   Project managers are responsible for not only navigating around risks but anticipating them so that they can try their best to avoid them altogether.
3. **Budgeting:** Only 2.5% of companies successfully complete all of the projects they take on. The rest go over schedule, over budget, or both.
   Project managers know that there are financial constraints they need to work within, and they use their budgeting and financial management skills to deliver winning projects within those limitations.
4. **Tracking and monitoring:** Project management isn't just about completing a project — it's about completing a successful project. That won't happen if project managers fail to keep their fingers on the pulse. They need to use their performance tracking and monitoring skills to ensure projects are running according to plan and still supporting the broader business goals. If not? They'll course-correct when necessary.
5. **Project management methodologies:** From Agile to Waterfall, there are numerous project management methodologies and approaches. These outline specific principles for overseeing and completing projects. Experienced project managers are familiar with those methodologies and can determine which ones are the best fit for their specific teams and projects.
6. **Meeting facilitation:** Kickoff meetings, status updates, retrospectives — the typical project process has many meetings, most of which are led by the project manager.
   For that reason, a project manager needs to be skilled at facilitating meetings, including creating an agenda, documenting notes, and following up on action items.
7. **Subject matter expertise:** Project managers work in a variety of industries, from construction to IT and everything in between. While it's not an absolute necessity, it's helpful for the project manager to have a basic familiarity with the industry and the types of projects they're managing.
8. **Project management software:** The best project managers know better than to try to coordinate all of the elements of a project with jumbled spreadsheets and task lists.

Project manager soft skills

9. **Leadership:** Project managers are the project leaders and often, the team leaders too. They're responsible for setting the team's vision and ensuring everyone is on board and motivated to bring the project through each phase.

   This requires getting buy-in from executives and project team members. These leaders should also equip people with the time, tools, and other resources they need to handle their to-do lists.

10. **Communication:** Any project management skills list is sure to include communication near the top. This includes written and verbal communication.

    Project managers need to ensure that team members and stakeholders are informed about the project plan, timeline, and budget and updated on the project's latest happenings.

11. **Collaboration:** It typically takes a village to complete a project. The project manager is tasked with rallying team members around the project vision, coordinating tasks, and ensuring that everybody works together effectively.

    To make that happen, a project manager needs to be a skilled collaborator. This also involves conflict resolution, as occasional project-related disagreements are unavoidable.

12. **Time management:** Every project manager will have numerous demands placed on their time – especially since they're acting as the point of contact for so many departments and team members.

    They must be able to manage their own time and the time and capacity of all of the project's key players.

13. **Organization:** Deadlines, resources, task dependencies – it's enough to make anybody's head spin, but a project manager views it as a fulfilling challenge.

    The best project managers are exceptionally organized and able to keep track of all of the moving pieces.

14. **Problem solving:** As much as you'd like to think that your project will go off without a hitch, unexpected issues are bound to crop up.

    Project managers can't be discouraged by a problem. Instead, they need to develop solutions to keep the project moving forward – even when the best-laid plans fall apart.

15. **Adaptability:** Project managers need to be adaptable. While planning is a core skill, they can't be so rigid with their strategies that everything falls apart the moment something unanticipated happens.

7. Explain the process of initialing and planning information system development project in brief. (5)

Ans: As its name implies, two major activities occur during project initiation and project planning. Project initiation focuses on activities that will help organize a team to conduct project planning. During initiation, one or more analysts are assigned to work with a customer to establish work standards and communication procedures.

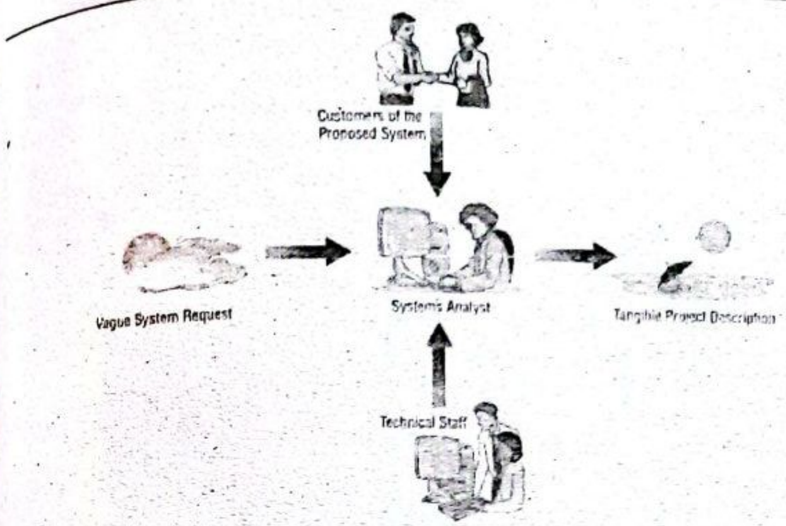


Figure: The systems analyst transforms a vague systems request into a tangible project description during project initiation and planning.

The second activity, project planning, focuses on defining clear, discrete tasks and the work needed to complete each task. The objective of the project planning process is to produce two documents: a baseline project plan (BPP) and the project scope statement (PSS). The BPP becomes the foundation for the remainder of the development project. It is an internal document used by the development team but not shared with customers. The PSS, produced by the project team, clearly outlines the objectives of the project for the customer. As with the project initiation process, the size, scope, and complexity of a project dictate the comprehensiveness of the project planning process and the resulting documents. Further, numerous assumptions about resource availability and potential problems will have to be made. Analysis of these assumptions and system costs and benefits forms a business case.

Types of Activities Performed during Project Initiation

- Establishing the project initiation team
- Establishing a relationship with the customer
- Establishing the project initiation plan
- Establishing management procedures
- Establishing the project management environment and project workbook.
- Developing the project charter

Activities Performed during Project Planning

- Describing the project scope, alternatives and feasibility
- Dividing the project into manageable tasks
- Estimating resources and creating a resource plan
- Developing a preliminary schedule
- Developing a communication plan
- Determining project standards and procedures
- Identifying and assessing risk
- Creating a preliminary budget
- Developing a project scope statement
- Setting a baseline project plan

8. **How can you use prototyping for determining system requirements? Compare throwaway prototyping with evolutionary prototyping.** (3+2)

Ans: Prototyping is a means of exploring ideas before you invest in them. Most system developers believe that the benefits from early usability data are at least ten times greater than those from late usability data. Prototyping allow system analysts quickly show users the basic requirement into a working version of the desired information system. After viewing and testing the prototype, the users usually adjust existing requirements to new ones. The goal with using prototyping to support requirement determination is to develop concrete specification for the ultimate system, not to build the ultimate system from prototyping. Prototyping is most useful for requirements determination when user requirements are not clear or well understood, one or a few users and other stakeholders are involved with the system, possible designs are complex and require concrete form to fully evaluate, communication problems have existed in the past between users and analysts, and Tools and data are readily available to rapidly build working systems, etc.

Prototyping is most useful for requirements determination when:

- User requirements are not clear or well understood, which is often the case for totally new systems or systems that support decision making.
- One or a few users and other stakeholders are involved with the system.
- Possible designs are complex and require concrete form to evaluate fully.
- Communication problems have existed in the past between users and analysts, and both parties want to be sure that system requirements are as specific as possible.
- Tools (such as form and report generators) and data are readily available to rapidly build working systems.

Throwaway Prototyping

Throwaway prototypes are developed from the initial requirements but they are not used for the final product and not an alternative for written specification of the requirements. It enables quick prototyping and commit to throwing the prototype away. If the users can get quick feedback on their requirements, they may be able to refine the requirements early in the development of the software. Then changes can be done early in the development life cycle.

Throwaway prototype has a short project timeline and easier and faster to develop the interface. This type of prototyping can be used at any time in a project by any of the project's personnel. Throwaway prototypes actually do nothing, it's just presentation only for a limited purpose. Soon it will be starting to become a thing of the past. This type of prototyping is not getting used as much now.

Evolutionary Prototyping

Evolutionary Prototyping is considered to be the most fundamental form of prototyping and this prototyping type is also known as breadboard prototyping. The main concept of this prototyping type is to build a robust prototype and constantly improve it. These prototypes are built only with well understood requirements instead of acknowledging all the requirements. It allows developers to add features or make changes that couldn't be devised during the requirements analyzing and designing.

Developers are helped to develop part by part of the system considering the usability aspects and this type of prototypes are delivered a working system to the end user.

9. **What is the purpose of database design? Compare logical design with physical design.** (2+3)

Ans: Certain principles guide the database design process. The first principle is that duplicate information (also called redundant data) is bad, because it wastes space and increases the likelihood of errors and inconsistencies. The second principle is that the correctness and completeness of information is important. If your database contains incorrect information, any reports that pull information from the database will also contain incorrect information. As a result, any decisions you make that are based on those reports will then be misinformed.

A good database design is, therefore, one that:

- Divides your information into subject-based tables to reduce redundant data.
- Provides Access with the information it requires to join the information in the tables together as needed.
- Helps support and ensure the accuracy and integrity of your information.
- Accommodates your data processing and reporting needs.

Database Design is a collection of processes that facilitate the designing, development, implementation and maintenance of enterprise data management systems. Properly designed database are easy to maintain, improves data consistency and are cost effective in terms of disk storage space. The database designer decides how the data elements correlate and what data must be stored.

The main objectives of database design in DBMS are to produce logical and physical designs models of the proposed database system.

- The logical model concentrates on the data requirements and the data to be stored independent of physical considerations. It does not concern itself with how the data will be stored or where it will be stored physically.
- The physical data design model involves translating the logical DB design of the database onto physical media using hardware resources and software systems such as database management systems (DBMS).

10. **What is testing? Explain any four different testing techniques.** (2+3)

Ans: Testing is the process or activity that checks the functionality and correctness of software according to specified user requirements in order to improve the quality and reliability of system. It is an expensive, time consuming, and critical approach in system development which requires proper planning of overall testing process.

A successful test is one that finds the errors. It executes the program with explicit intention of finding error, i.e., making the program fail. It is a process of evaluating system with an intention of creating a strong system and mainly focuses on the weak areas of the system or software.

Types of Testing

Testing can be of various types and different types of tests are conducted depending on the kind of bugs one seeks to discover:

Unit Testing

During this first round of testing, the program is submitted to assessments that focus on specific units or components of the software to determine whether each one is fully functional. The main aim of this endeavor is to determine whether the application functions as designed. In this phase, a unit can refer to a function, individual program or even a procedure, and a White-box Testing method is usually used to get the job done. One of the biggest benefits of this testing phase is that it can be run every time a piece of code is changed, allowing issues to be resolved as quickly as possible. It's quite common for software developers to perform unit tests before delivering software to testers for formal testing.

Integration Testing

Integration testing allows individuals the opportunity to combine all of the units within a program and test them as a group. This testing level is designed to find interface defects between the modules/functions. This is particularly beneficial because it determines how efficiently the units are running together. Keep in mind that no matter how efficiently each unit is running, if they aren't properly integrated, it will affect the functionality of the software program. In order to run these types of tests, individuals can make use of various testing methods, but the specific method that will be used to get the job done will depend greatly on the way in which the units are defined.

System Testing

System testing is the first level in which the complete application is tested as a whole. The goal at this level is to evaluate whether the system has complied with all of the outlined requirements and to see that it meets Quality Standards. System testing is undertaken by independent testers who haven't played a role in developing the program. This testing is performed in an environment that closely mirrors production. System Testing is very important because it verifies that the application meets the technical, functional, and business requirements that were set by the customer.

Acceptance Testing

The final level, Acceptance testing (or User Acceptance Testing), is conducted to determine whether the system is ready for release. During the Software development life cycle, requirements changes can sometimes be misinterpreted in a fashion that does not meet the intended needs of the users. During this final phase, the user will test the system to find out whether the application meets their business' needs. Once this process has been completed and the software has passed, the program will then be delivered to production.

11. What is class diagram ? Explain class diagram with suitable example. (2+3)

Ans: The UML Class diagram is a graphical notation used to construct and visualize object oriented systems. A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the systems:

- classes,
- their attributes,
- operations (or methods),
- And the relationships among objects.

Purpose of Class Diagrams

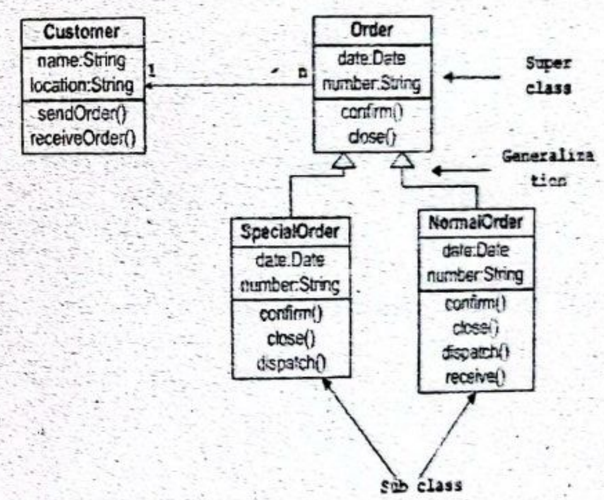
- Shows static structure of classifiers in a system
- Diagram provides a basic notation for other structure diagrams prescribed by UML
- Helpful for developers and other team members too
- Business Analysts can use class diagrams to model systems from a business perspective

The following diagram is an example of an Order System of an application. It describes a particular aspect of the entire application.

- First of all, Order and Customer are identified as the two elements of the system. They have a one-to-many relationship because a customer can have multiple orders.
- Order class is an abstract class and it has two concrete classes (inheritance relationship) Special Order and Normal Order.
- The two inherited classes have all the properties as the Order class. In addition, they have additional functions like dispatch () and receive ().

The following class diagram has been drawn considering all the points mentioned above.

Sample Class Diagram



12. Write short notes on:

a. Agile development

Ans: In earlier days Iterative Waterfall model was very popular to complete a project. But nowadays developers face various problems while using it to develop a software. The main difficulties included handling change requests from customers during project development and the high cost and time required to incorporate these changes. To overcome these drawbacks of Waterfall model, in the mid-1990s the Agile Software Development model was proposed.

The Agile model was primarily designed to help a project to adapt to change requests quickly. So, the main aim of the agile model is to facilitate quick project completion. To accomplish this task agility is required. Agility is

achieved by fitting the process to the project, removing activities that may not be essential for a specific project. Also, anything that is wastage of time and effort is avoided.

In the Agile model, the requirements are decomposed into many small parts that can be incrementally developed. The Agile model adopts Iterative development. Each incremental part is developed over an iteration. Each iteration is intended to be small and easily manageable and that can be completed within a couple of weeks only. At a time one iteration is planned, developed and deployed to the customers. Long-term plans are not made. Agile model is the combination of iterative and incremental process models. Steps involve in agile SDLC models are:

- Requirement gathering
- Requirement Analysis
- Design
- Coding
- Unit testing
- Acceptance testing

b. Decision Tree

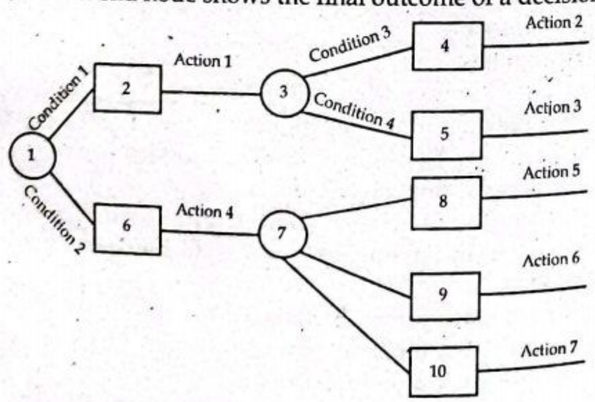
Ans: A decision tree is a diagram that shows alternative actions and conditions within horizontal tree framework. Thus, it depicts which conditions to consider first, second, and so on.

Decision trees depict the relationship of each condition and their permissible actions. A square node indicates an action and a circle indicates a condition. It forces analysts to consider the sequence of decisions and identifies the actual decision that must be made.

A decision tree is a map of the possible outcomes of a series of related choices. It allows an individual or organization to weigh possible actions against one another based on their costs, probabilities, and benefits. They can can be used either to drive informal discussion or to map out an algorithm that predicts the best choice mathematically.

A decision tree typically starts with a single node, which branches into possible outcomes. Each of those outcomes leads to additional nodes, which branch off into other possibilities. This gives it a treelike shape.

There are three different types of nodes: chance nodes, decision nodes, and end nodes. A chance node, represented by a circle, shows the probabilities of certain results. A decision node, represented by a square, shows a decision to be made, and an end node shows the final outcome of a decision path.



MODEL QUESTIONS SETS FOR PRACTICE

MODEL SET 1

Course Title: System Analysis and Design — **Full Marks: 60**
Course No: CSC314 — **Pass Marks: 24**
Semester: V — **Time: 3 hrs**

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Explain the RAD model. State the advantages and disadvantages of RAD approach.
2. List and describe the advantages of top-down planning over other planning approaches.
3. How are E-R diagrams similar to and different from decision trees? In what ways are data and logic modeling techniques complementary? What problems might be encountered if either data or logic modeling techniques were not performed well or not performed at all as part of the systems development process?

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. What is a business process? Why is business process diagramming important?
5. What do you mean by system? Explain the characteristics of system.
6. Explain object oriented system with reference to class, object, encapsulation, abstraction, message passing, inheritance, interface and polymorphism with suitable example.
7. What types of measurements must be taken to gain an understanding of the effectiveness of maintenance? Why is tracking mean time between failures an important measurement?
8. What column, row, and text formatting issues are important when designing tables and lists?
9. Explain the purpose of data compression techniques.
10. What is DFD? Explain in details about DFD levels with suitable example.
11. Explain development methodology used in developing information system in detail with example.
12. Explain with example of the linkage between data flow, decision tables and entity relationship diagram.

MODEL SET 2

Section A

Attempt any two questions. (2 × 10 = 20)

1. What do you mean by Normalization? How can you transforming E-R diagram into relation? Explain with suitable example.
2. Differentiate between Use Case diagram and Sequence diagram in object oriented analysis and design.
3. Draw a DFD diagram of Mess Management System up to level 2.

Section B

Attempt any eight questions. (8 × 5 = 40)

4. Choose a transaction that you are likely to encounter, perhaps ordering a cap and gown for graduation, and develop a high-level DFD or a context diagram. Decompose this to a level-0 diagram.
5. Define Information System. What are the typical components of Information Systems?
6. What is the meaning of encapsulation from the viewpoint of structured system analysis and design? Explain how does encapsulation and abstraction concepts work together in object orientation?
7. What managerial issues can be better understood by measuring maintenance effectiveness?
8. Describe how numeric, textual, and alphanumeric data should be formatted in a table or list.
9. What problems can arise when merging relations (view integration)? What do you mean by relation? List out any five properties of relation.
10. Differentiate between logical and physical DFD?
11. What are the process for designing forms and reports?
12. Explain the unified modeling language (UML) with example.

MODEL SET 3

Section A

Attempt any two questions. (2 × 10 = 20)

1. What are different types of file organizations used in creating file design during SDLC? Discuss.
2. What are various characteristics and elements of a system? Discuss.
3. Write a short note on fundamental of coding. What should be the criteria for selection of programming language?

Section B

Attempt any eight questions. (8 × 5 = 40)

4. Describe systems analysis and the major activities that occur during this phase of the systems development life cycle.

5. Explain the types of Information System. What is information systems analysis and design?
6. Differentiate Persistent & Non-persistent Objects? What is the use of sequence diagram? Explain with suitable example.
7. Describe the process for controlling maintenance requests. Should all requests be handled in the same way or are there situations when you should be able to circumvent the process? If so, when and why?
8. What is meant by usability and what characteristics of an interface are used to assess a system's usability?
9. What are the goals of designing physical tables?
10. What are various types of testing? Explain.
11. Briefly explain the steps in feasibility analysis.
12. What are the main principles used in designing Forms and Reports.

MODEL SET 4

Section A

Attempt any two questions. (2 × 10 = 20)

1. What is file? Explain any technique for implementing records in file.
2. What is software testing? Explain different types of testing.
3. State the activities involved in Object-Oriented Development Life Cycle.

Section B

Attempt any eight questions. (8 × 5 = 40)

4. Describe four traditional techniques for collecting information during analysis. When might one be better than another?
5. Describe the steps involved in corporate strategic planning. What are the three generic competitive strategies?
6. Who is system analyst? Discuss the roles of system analyst.
7. How has systems analysis and design changed over the past four decades?
8. What do you mean by analysis and design? What are the main underlying concepts of object orientation?
9. What is meant by configuration management? Why do you think organizations have adopted the approach of using a systems librarian?
10. Discuss the benefits, problems, and general design process for the use of color when designing system output.
11. What is the purpose of de-normalization? Why might you not want to create one physical table or file for each relation in a logical data model?
12. What is Class diagram? Draw a class diagram for Payment and Purchase of Customer.

MODEL SET 5

Section A

Attempt any two questions. (2 × 10 = 20)

1. Explain the use of structure chart in structured design with a suitable example.

2. How to construct a checklist for design review? Explain.
3. Define the concept of Integrated CASE tools. How are they useful?

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. What is JAD? How is it better than traditional information gathering techniques? What are its weaknesses?
5. List and describe the advantages of top-down planning over other planning approaches.
6. List and explain the different phases in the SDLC.
7. List and explain some of the problems with the traditional waterfall SDLC.
8. How does the requirement elicitation process happen in object oriented analysis? Explain with reference to system behavior analysis of any exemplary system case.
9. How are automated tools used in the maintenance of information systems? What is the difference between reverse engineering and reengineering tools?
10. What type of labeling can you use in a table or list to improve its usability?
11. Explain the process of system implementation and Maintenance.
12. What is Information system analysis and Design? Explain the stages of SDLC.

MODEL SET 6

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Discuss the role of a system analyst in system analysis and design. Explain in detail the structure of SRS.
2. Discuss software specifications and classifications in system analysis.
3. Define System maintenance. Give the example of System maintenance by taking a case study.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. Describe how prototyping can be used during requirements determination. How is it better or worse than traditional methods?
5. List and describe the steps in the project initiation and planning process.
6. Define CASE tools. Explain the types of CASE tools.
7. In OOAD, there are various types of models, like conceptual, structural, behavioral etc. what is the significance of these many different types of model? Explain with illustrative example.
8. What are the deliverables from coding, testing, and installation? Explain the code-testing process.
9. How can differences in user, task, system, or the environment influence the design of a form or report? Provide an example that contrasts characteristics for each difference.
10. What is the purpose of information gathering tools? Discuss various methods of information gathering.
11. Explain the design principles. What is the importance of user manual in design?

12. Write qualities of good team leader.

MODEL SET 7

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. What is the importance of Testing? Discuss the use of System Testing.
2. What is software implementation process? What are the phases of the system implementation process?
3. What do you mean by system implementation? Discuss different methods used for system implementation.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. What are disruptive technologies and how do they enable organizations to radically change their business processes?
5. What intangible benefits might an organization obtain from the development of an information system?
6. What is framework? How design pattern is useful? Explain any one design pattern in detail with suitable example.
7. What is the conventional wisdom about implementation success?
8. List and define the factors that are important to successful implementation efforts.
9. Describe five methods of interacting with a system. Is one method better than all others? Why or why not?
10. List and describe the common interface and dialogue design errors found on websites.
11. What do you mean by system? Discuss different types of system.
12. What is the purpose of documentation? Discuss the use of different types of documents prepared during documentation.

MODEL SET 8

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. A system costs Rs. 2, 00, 000 to install and Rs. 10,000 per month as recurring expenses. The benefit per year is 1, 50,000. Assuming as interest rate is 15%, what is the payback period of the investment?
2. Who is system analyst? List and explain the skills of system analyst.
3. Draw a DFD diagram of college library system up to level 2.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. Why is continual user involvement a useful way to discover system requirements? Under what conditions it might be used? Under what conditions might it not be used?

5. Describe three commonly used methods for performing economic cost-benefit analysis.
6. Explain Agile Methodologies. When would you use agile methodologies?
7. What is framework? How design pattern is useful? Explain any one design pattern in detail with suitable example.
8. In OOAD, there are various types of models, like conceptual, structural, behavioral etc. what is the significance of these many different types of model? Explain with illustrative example.
9. How does the requirement elicitation process happen in object oriented analysis? Explain with reference to system behavior analysis of any exemplary system case.
10. What are the four approaches to installation? Which is the most expensive? Which is the most risky? How does an organization decide which approach to use?
11. Describe the properties of windows and forms in a GUI environment. Which property do you feel is most important? Why?
12. List four contributing factors that have acted to impede the design of high-quality interfaces and dialogues on Internet-based applications.

MODEL SET 9

Section A

Attempt any two questions. (2 × 10 = 20)

1. What is software requirement specification (SRS)? What are the characteristics of good SRS document? Explain.
2. Differentiate between transaction analysis and transform analysis.
3. Draw the sequence diagram to renew a book from library.

Section B

Attempt any eight questions. (8 × 5 = 40)

4. Which types of CASE tools are appropriate for use during requirements determination?
5. Is any feasibility factor most important? Why or why not?
6. Explain the RAD model. State the advantages and disadvantages of RAD approach.
7. What is the use of class diagram? Explain use of class diagram with suitable example.
8. What is the conventional wisdom about implementation success?
9. What are structured walk-throughs for code? What is their purpose? How are they conducted? How are they different from code inspections?
10. What is the purpose of normalization?
11. What is significance of software testing? Discuss in detail module level and system level testing methods.
12. Describe software specification documents and attributes.

□□□

Web Technology

Full Marks: 60 + 20 + 20
Pass Marks: 24 + 8 + 8
Credit Hrs: 3

Course Title: Web Technology
Course No: CSC318
Nature of the Course: Theory + Lab
Semester: V

Course Description: This course covers the fundamental concepts of HTML, CSS, JavaScript, XML, and PHP.

Course Objectives: The main objective of this course is to provide basic knowledge of web design using HTML and CSS, client side scripting using JavaScript, handling web data using XML and server side scripting using PHP.

Course Contents:

Unit 1: Introduction **(3 Hrs.)**

Web Basics: Internet, Intranet, WWW, Static and Dynamic Web Page; Web Clients; Web Servers; Client Server Architecture: Single Tier, Two-Tier, Multi-Tier; HTTP: HTTP Request and Response; URL, Client Side Scripting, Server Side, Scripting, Web 1.0, Web 2.0

Unit 2: Hyper Text Markup Language **(10 Hrs.)**

Introduction to HTML; Elements of HTML Document; HTML Elements and HTML Attributes,

Headings, Paragraph, Division, Formatting: b, i, small, sup, sub; Spacing: Pre, Br; Formatting Text Phrases: span, strong, tt; Image element; Anchors; Lists: Ordered and Unordered and Definition; Tables; Frames; Forms: Form Elements, ID attributes, Class Attributes of HTML Elements; Meta Tag, Audio, Video, Canvas, Main, Section, Article, Header, Footer, Aside, Nav, Figure Tags; HTML Events: Window Events, Form Element Events, Keyboard Events, Mouse Events

Unit 3: Cascading Style Sheets **(8 Hrs.)**

Introduction; Cascading Style Sheets (CSS); CSS Syntax; Inserting CSS: Inline, Internal, External,

ID and Class Selectors; Colors; Backgrounds; Borders; Text; Font; List; Table; CSS Box Model; Normal Flow Box Layout: Basic Box Layout, Display Property, Padding, Margin; Positioning: Relative, Float, Absolute; CSS3 Borders, Box Shadows, Text Effects and shadow; Basics of Responsive Web Designs; Media Queries, Introduction to Bootstrap

Unit 4: Client Side Scripting with JavaScript **(9 Hrs.)**

Structure of JavaScript Program; Variables and Data Types; Statements: Expression, Keyword,

Block; Operators; Flow Controls, Looping, Functions; Popup Boxes: Alert, Confirm, Prompt; Objects and properties; Constructors; Arrays; Built-in Objects: Window, String, Number, Boolean, Date, Math, RegExp, Form, DOM; User Defined Objects; Event Handling and Form Validation, Error Handling, Handling Cookies, jQuery Syntax; jQuery Selectors, Events and Effects; Introduction to JSON

Unit 5: AJAX and XML (7 Hrs.)

Basics of AJAX; Introduction to XML and its Application; Syntax Rules for creating XML document; XML Elements; XML Attributes; XML Tree; XML Namespace; XML schema languages: Document Type Definition(DTD), XML Schema Definition (XSD); XSD Simple Types, XSD Attributes; XSD Complex Types; XML Style Sheets (XSLT), XQuery

Unit 6: Server Side Scripting using PHP (8 Hrs.)

PHP Syntax, Variables, Data Types, Strings, Constants, Operators, Control structure, Functions,

Array, Creating Class and Objects, PHP Forms, Accessing Form Elements, Form Validation,

Events, Cookies and Sessions, Working with PHP and MySQL, Connecting to Database, Creating, Selecting, Deleting, Updating Records in a table, Inserting Multiple Data, Introduction to CodeIgniter, Laravel, Wordpress etc.

Laboratory Works:

The laboratory work includes creating web pages and applications with using HTML, CSS,

JavaScript, XML, and PHP. Students have to prepare a web based application, using above mentioned technologies, as a project work.

Text Books:

1. Web Design with HTML, CSS, JavaScript and jQuery Set, Jon Duckett, *John Wiley & Sons*
2. Web Technologies: A Computer Science Perspective, Jeffrey C. Jackson, *Pearson Prentice Hall*
3. Learning PHP, MySQL & JavaScript: with jQuery, CSS & HTML5, Robin Nixon, *O'Reilly*
4. PHP & MySQL: Server-side Web Development, Jon Ducket, *Wiley*

Reference Books:

1. HTML5 and CSS3 for the Real World", Estelle Weyl, Louis Lazaris, Alexis Goldstein, *Sitepoint*
2. HTML & CSS: Design and Build Websites, Jon Duckett, *John Wiley & Sons*
3. Dynamic Web Programming and HTML5, Paul S. Wang, *CRC Press*
4. HTML5 Programming with JavaScript for Dummies, John Paul Mueller
5. JavaScript and JQuery: Interactive Front-end Web Development, Jon Duckett, *Wiley*
6. The Complete Reference: HTML and CSS, Thomas A. Powell, *Mc Graw Hill*
7. JavaScript: The Web Technologies Series, Don Gosseli, *Course Technology Cengage Learning*
8. Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, ASP.NET, XML and AJAX, Black Book, *Dreamtech Press*
9. An Introduction to XML and Web Technologies, Anders Møller and Michael I. Schwartzbach, *Addison-Wesley*
10. PHP and MySQL Web Development, Luke Welling, *Addison Wesley*
11. www.w3schools.com

TRIBHUVAN UNIVERSITY
Institution of Science and Technology
TU QUESTIONS-ANSWERS 2076

Bachelor Level/Third Year/Fifth Semester/Science — Full Marks: 60
Course Title: Web Technology — Pass Marks: 24
Course Code: CSC 315 — Time: 3 hrs.

Candidates are required to give their answers in their own words as far as practicable.

The figures in the margin indicate full marks.

Section A **(2 × 10 = 20)**

Attempt any two questions.

1. **Create web form for book search catalog. The form should contain a dropdown defining search type, a text box for search keyword, a radio button for download type true or false, now write PHP script to store data from the form into database table and also retrieve the results from stored table in a new page.** **[10]**

Ans:

```
//index.html
<!DOCTYPE html>
<html>
<head>
 <meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Form Registration</title>
</head>
<body>
<h2>Book Search Catalog</h2>
<form method="POST" action="formSubmit.php">
<div>
 <label for="country">Choose Search Type: </label>
 <select name="searchtype">
   <option value="history">History</option>
   <option value="politics">Politics</option>
   <option value="childrenbook">Children's Book</option>
  <option value="drama">Drama</option>
  <option value="science">Science</option>
  </select>
 </div>
 <div>
 <label for="name">Please enter name of book: </label>
 <input type="text" name="name" id="name">
 </div>
 <div>
 <label for="download">Download: </label>
```

```
 <input type="radio" name="download" value="true">True
 <input type="radio" name="download" value="false">False
</div>
<div>
 <input type="submit" name="submit" value="Submit">
</div>
</form>
</body>
</html>

//formSubmit.php
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "webtech";
$tablename = "books";
if($_SERVER["REQUEST_METHOD"] == "POST")
{
$searchtype = $_POST['searchtype'];
$name = $_POST['name'];
$download = $_POST['download'];
//connect to database
$conn = mysqli_connect($servername, $username, $password) or
die(mysql_error()); //Connect to server
mysqli_select_db($conn, $dbname) or die("Cannot connect to database");
//Connect to database

// Insert the values into database
mysqli_query($conn, "INSERT INTO ".$tablename."(searchtype, name,
download) VALUES ('$searchtype', '$name', '$download')");
Print '<script> alert("Congrats! Your Submission is Successfull!"); </script>';
Print '<script>window.location.assign("display.php");
</script>';
}
?>
//display.php
<?php
$db = "webtech";
$conn = mysqli_connect("localhost", "root","") or die(mysql_error());
mysqli_select_db($conn, $db) or die("Cannot connect to database");
$query = mysqli_query($conn, "Select * from books");
?>
<html>
<head>
```



```
<title>Displaying Data</title>
</head>
<body>
<table border="1">
<tr>
<th>Search Type</th>
<th>Book Name</th>
<th>Download Type</th>
</tr>
<?php
while($row = mysqli_fetch_array($query))
{
?>
<tr>
<th><?php echo $row['searchtype']; ?> </th>
<th><?php echo $row['name']; ?> </th>
<th><?php echo $row['download']; ?> </th>
</tr>
<?php
}
?>
</table>
<a href="index.html">Go back to Search New Book</a>
</body>
</html>
```

2. **How can you create objects in JavaScript? Create a HTML page containing a division with image and its description in paragraph. Write a JavaScript function to change the value of description in the paragraph during on mouse cover and on mouse out event on the image.** **[4+6]**

Ans: A javaScript object is an entity having state and behavior (properties and method). For example: car, pen, bike, chair, glass, keyboard, monitor etc. There are different ways to create new objects:

1. By using object literal: The syntax of creating object using object literal is given below:
   Object = {property1:value1, property2:value2.....propertyN:valueN}
   Example: const person = {name: Aadesh, age: 03, hobbies: ['reading', 'singing', 'coding']};
2. By creating instance of Object directly (using new keyword): The syntax of creating object directly is given below:
   objectname = new Object();
   Example: const person = new Object ({name: Aadesh, age: 03, hobbies: ['reading', 'singing', 'coding']});
3. By using an object constructor (using new keyword): Here, we need to create function with arguments. Each argument value can be assigned in the current object by using this keyword.

Example:

```
function Person(name, age)
{
this.name = name;
this.age = age;
}
const person = new Person("Aadesh", 03);
```

Second part:

index.html

```
<!DOCTYPE html>
<html>
<head>
<title>TU</title>
<script type="text/javascript" src="web.js"></script>
</head>
<body>
<div>
<img src="images/tu.jpg" onmouseover="changeText()" onmouseout=
"defaultText()">
<div id="description">
<p>Department of CSIT, Tribhuvan University</p>
</div>
</div>
</body>
</html>
```

web.js

```
function changeText()
{
var display = document.getElementById('description');
display.innerHTML = "";
display.innerHTML = "Department of Computer Science and Information
Technology, TU!! (Changed Text)";
}
function defaultText()
{
var display = document.getElementById('description');
display.innerHTML = "";
display.innerHTML = "Department of CSIT, Tribhuvan University";
}
```

3. How positioning of HTML elements can be done using CSS? Create a HTML page with a div with some content and two paragraph tags with some contents having id p1 and p2. Write external CSS for the div tag having fixed position with border style solid and width 2px. The p1 should have relative position. The font type of p1 should be Arial and color should be red. The p2 have absolute position with top 0px and left 200px. (2+8)

Ans: The position property in CSS tells about the method of positioning for an element or an HTML entity. There are five different types of position property available in CSS:

- Fixed
- Static
- Relative
- Absolute
- Sticky

The positioning of an element can be done using the top, right, bottom, and left properties. These specify the distance of an HTML element from the edge of the viewport. To set the position by these four properties, we have to declare the positioning method.

Second Part:

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="style.css">
<title>Collegenote</title>
</head>
<body>
<div class="cn">
The New summit college is one of the oldest and Pioneer IT College.
</div>
<p id="p1">
New Summit College mainly focus on Practical knowledge and real life
practices to the students.
</p>
<p id="p2">
Students of B.Sc. CSIT, BCA and BBM are technical students.
</p>
</body>
</html>
```

style.css

```
.cn{
position:fixed;
border: 2px solid black;
}
#p1{
position: relative;
font-family: Arial;
color: red;
}
#p2{
position:absolute;
```

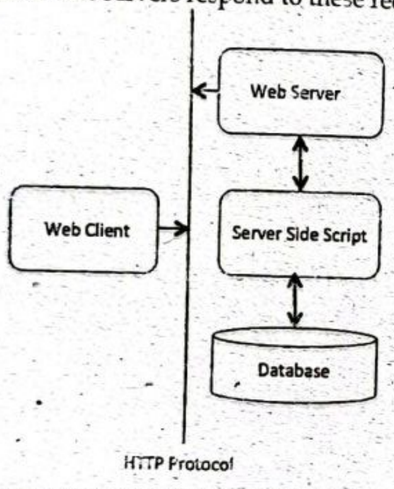
```
top: 0px;
left: 200px;
}
```

Section B

Attempt any eight questions. $(8 \times 5 = 40)$

4. What is HTTP protocol? Define HTTP Request and Response. [2+3]

Ans: HTTP is a TCP/IP based communication protocol, that is used to deliver data (HTML files, image files, query results, etc.) on the World Wide Web. The default port is TCP 80, but other ports can be used as well. It provides a standardized way for computers to communicate with each other. HTTP specification specifies how clients' request data will be constructed and sent to the server, and how the servers respond to these requests.



HTTP works as a request-response protocol between a client and server. For example: A client (browser) sends an HTTP request to the server; then the server returns a response to the client. The response contains status information about the request and may also contain the requested content.

- **HTTP Request**: HTTP client sends an HTTP Request to the HTTP Server according to the HTTP standard, specifying the information the client like to retrieve from the Hypertext Transfer Protocol (HTTP) Server.
- **HTTP Response:** Once the HTTP Request arrived at the HTTP server, it will process the request and creates an HTTP Response message. The HTTP response message may contain the resource the HTTP Client requested or information why the HTTP request failed.

5. Create a HTML page with tags header, article and footer. Insert a link containing mail to info@iost.edu.np in the footer tag. Set the keywords 'iost', 'csit' using Meta tag in the page. [5]

Ans:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="keywords" content="iost, csit">
<title>IOST</title>
```



```
</head>
<body>
<header style="background-color: darkgray; margin-bottom: 2px;">
<nav>
<a href="#">Home</a>
<a href="#">About</a>
<a href="#">Contact</a>
</nav>
</header>
<article>
Institute of Science and Technology (IoST) is one of the oldest and the largest technical institutes in TU.
</article>
<footer style="background-color: darkgray; margin-top: 2px; text-align: center;">
<a href="info@iost.edu.np">Send Email</a>
</footer>
</body>
</html>
```

6. How JQuery animate cab be used to create custom animation? Write syntax with sample script.

Ans: The animate() method performs a custom animation of a set of CSS properties. This method changes an element from one state to another with CSS styles. The CSS property value is changed gradually, to create an animated effect. Only numeric values can be animated (like "margin: 30px"). String values cannot be animated (like "background-color : red"), except for the strings "show", "hide" and "toggle".

Syntax:

$(selector).animate ({properties}, duration, callback);

- Properties parameter defines the CSS properties to be animated.
- Duration parameter is optional and specifies the duration of the effect. It can be set as "slow", "fast" or milliseconds.
- Callback parameter is also optional and it is a function which is executed after the animation completes.

Example:

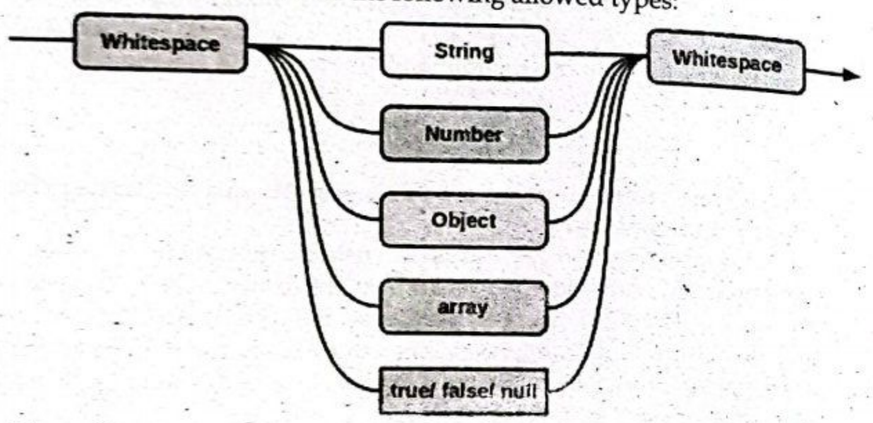
```
<!DOCTYPE html>
<html>
<head>
<script
src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></
script>
<script>
$(document).ready(function(){
$("button").click(function(){
$("div").animate({left: '450px'});
});
});
</script>
</head>
```

```
<body>
<button>Start Animation</button>
<div style="background: #98bf21; height:100px; width:100px; position: absolute;"></div>
</body>
</html>
```

7. What is the use of JSON? How can you parse a JSON, illustrate with an example. [1+4]

Ans: JavaScript Object Notation (JSON) is a standard text-based format for representing structured data based on JavaScript object syntax. It is commonly used for transmitting data in web applications (e.g., sending some data from the server to the client, so it can be displayed on a web page, or vice versa).

JSON value can take one of the following allowed types:



Example:

```
{
"name": "Aashna",
"age": 06
}
```

The JSON.parse() method parses a JSON string, constructing the JavaScript value or object described by the string.

Syntax: JSON.parse(text);

text is the string to parse as JSON.

Example:

```
const json = '{"name":"Aashna, "age":08}';
const obj = JSON.parse(json);
console.log(obj.name);      // expected output: Aashna
console.log(obj.age);       // expected output:08
```

8. What is XML? Create an XML file containing records of employee having elements of simple and complex types. [1+4]

Ans: XML stands for Extensible Markup Language. It is a text-based markup language derived from Standard Generalized Markup Language (SGML). XML tags identify the data and are used to store and organize the data, rather than specifying how to display it like HTML tags, which are used to display the data. XML is not going to replace HTML in the near future, but it introduces new possibilities by adopting many successful features of HTML.

There are three important characteristics of XML that make it useful in a variety of systems and solutions:

- **XML is extensible** – XML allows you to create your own self-descriptive tags, or language, that suits your application.
- **XML carries the data, does not present it** – XML allows you to store the data irrespective of how it will be presented.
- **XML is a public standard** – XML was developed by an organization called the World Wide Web Consortium (W3C) and is available as an open standard.

Second part,

```
<employee>
 <firstname>Kamala</firstname>
 <lastname>Karki</lastname>
</employee>

<xs:element name="employee" type="fullpersoninfo"/>
<xs:complexType name="personinfo">
 <xs:sequence>
  <xs:element name="firstname" type="xs:string"/>
  <xs:element name="lastname" type="xs:string"/>
 </xs:sequence>
</xs:complexType>

<xs:complexType name="fullpersoninfo">
 <xs:complexContent>
  <xs:extension base="personinfo">
   <xs:sequence>
    <xs:element name="address" type="xs:string"/>
    <xs:element name="city" type="xs:string"/>
    <xs:element name="country" type="xs:string"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>
```

9. How XSD of a XML file is created? Illustrate with an example. [2+3]

Ans: An XML Schema describes the structure of an XML document. The XML Schema language is also referred to as XML Schema Definition (XSD).

The purpose of an XML Schema is to define the legal building blocks of an XML document:

- the elements and attributes that can appear in a document
- the number of (and order of) child elements
- data types for elements and attributes
- default and fixed values for elements and attributes

An XML Schema is also an XML document with XSD extension.

First item: xml declaration

```
<?xml version="1.0" encoding="UTF-8"?>
```

XML comments and processing instructions are allowed.

Root element: schema with a namespace declaration.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```
</xs:schema>
```

Possible namespace prefixes: xs, xsd, or none.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<shiporder orderid="889923"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="shiporder.xsd">
 <orderperson>John Smith</orderperson>
 <shipto>
  <name>Ola Nordmann</name>
  <address>Langgt 23</address>
  <city>4000 Stavanger</city>
  <country>Norway</country>
 </shipto>
 <item>
  <title>Empire Burlesque</title>
  <note>Special Edition</note>
  <quantity>1</quantity>
  <price>10.90</price>
 </item>
 <item>
  <title>Hide your heart</title>
  <quantity>1</quantity>
  <price>9.90</price>
 </item>
</shiporder>
```

10. How can you define variables in PHP? Define any two variables of string types and display their results after concatenation. [1+4]

Ans: In PHP, a variable is declared using a $ sign followed by the variable name.

```
$variablename = value;
```

E.g.

```
$str="hello string";
$x=200;
$y=44.6;
```

Second Part:

```
<?php
    $fname = 'Jayanta';    // First String
    $lname = 'Poudel';    // Second String
    $fullname = $fname." ".$lname;    // Concatenation Of String
    echo " $fullname \n";    // print Concatenate String
?>
```

11. How web pages can be made responsive using media queries? Illustrate with an example. [2+3]

Ans: CSS Media Queries allow us to create responsive websites that look good on all screen sizes, from desktop to mobile. It uses the @media rule to include a

block of CSS properties only if a certain condition is true. A media query consists of a media type and can contain one or more expressions, which resolve to either true or false.

```
@media not|only mediatype and (expressions)
{
 CSS-Code;
}
```

The result of the query is true if the specified media type matches the type of device the document is being displayed on and all expressions in the media query are true. When a media query is true, the corresponding style sheet or style rules are applied, following the normal cascading rules.

Example:

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
.wrapper {overflow: auto;}
.menuitem {
 border: 1px solid #d4d4d4;
 list-style-type: none;
 margin: 4px;
 padding: 2px;
}
@media screen and (min-width: 480px)
{
 #leftsidebar
 {
  width: 200px;
  float: left;
 }
 #main
 {
  margin-left: 216px;
 }
}
</style>
</head>
<body>
<div class="wrapper">
 <div id="leftsidebar">
  <ul id="menulist">
   <li class="menuitem">Menu-item 1</li>
   <li class="menuitem">Menu-item 2</li>
   <li class="menuitem">Menu-item 3</li>
  </ul>
 </div>
```

```
<div id="main">
<p>This example shows a menu that will float to the left of the page if the viewport is 480 pixels wide or wider. If the viewport is less than 480 pixels, the menu will be on top of the content. </p>
</div>
</div>
</body>
</html>
```

12. Why inline frames are used? Create a HTML page containing ifframe with in a paragraph. The iframe have source to http:/www.tuiost.edu.np and its height and width are 200px and 400px respectively. [1+4]

Ans: An inline frame is used to embed another document within the current HTML document. The HTML <iframe> tag specifies an inline frame.

Syntax: <iframe src="url" title="description"></iframe>

index.html

```
<!DOCTYPE html>
<html>
<head>
  <title>TU</title>
</head>
<body>
 <p>
Institute of Science and Technology (IoST) is one of the oldest and the largest technical institutes in TU with 13 Central Departments, 1 School, 24 constituent campuses and 89 affiliated campuses. For more details visit our website:
<iframe src = "http://www.tuiost.edu.np" width = "400px" height = "200px" title="tuiost website">
Sorry your browser does not support inline frames.
  </iframe>
 </p>
</body>
</html>
```


TRIBHUVAN UNIVERSITY

Institution of Science and Technology

TU QUESTIONS-ANSWERS 2078

Bachelor Level/Third Year/Fifth Semester/Science

Course Title: Web Technology — Full Marks: 60

Course Code: CSC 315 — Pass Marks: 24

Time: 3 hrs.

Candidates are required to give their answers in their own words as far as practicable.

The figures in the margin indicate full marks.

Section A

Attempt any two questions. (2 × 10 = 20)

1. How you define array in PHP? Write a PHP script to create a multidimensional array named product that will contain pcode, pname and price. Initialize the array with at least three instances. Also write HTML script to display the initialized values of array in a HTML table. [2+8]

Ans: A multidimensional array is an array containing one or more arrays. PHP supports multidimensional arrays that are two, three, four, five, or more levels deep.

Multi-dimensional arrays are such type of arrays which stores another array at each index instead of single element. In other words, define multi-dimensional arrays as array of arrays. As the name suggests, every element in this array can be an array and they can also hold other sub-arrays within. Arrays or sub-arrays in multidimensional arrays can be accessed using multiple dimensions.

```
<!DOCTYPE html>
<html>
<body>
<?php
$Product = array (
 array("P101","Car",1800000),
 array("P102","Iron",1600),
 array("P103","Clouth",1800),
 array("P104","Laptop",160000),
);
echo $product[0][0].": of name: ".$ product [0][1].", of price: ".$ product [0][2].".<br>";
echo $ product [1][0].": of name: ".$ product [1][1].", of price: ".$ product [1][2].".<br>";
echo $ product [2][0].": of name: ".$ product [2][1].", of price: ".$ product [2][2].".<br>";
echo $ product [3][0].": of name: ".$ product [3][1].", of price: ".$ product [3][2].".<br>";
?>
</body>
</html>
```

2. Create a HTML signup form with fields Name, Email, password, Age. Validate the form using JavaScript. Write functions for validating each of elements. All of the fields should not be empty. The email address should be a valid email, the password should be of length at least 6 and should start with alphabet and end with digit. The age should be in between 8 and 60. [10]

Ans: index.html

```
<!DOCTYPE html>
<html lang="en">
 <head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0"
     />
  <link rel="stylesheet" href="./style.css" />
  <title>Document</title>
 </head>
 <body>
  <h3>Registration form:</h3>
  <div class="registration">
   <form name="f1" action="#" onsubmit="return validate()">
    <table>
     <tr>
      <td>Enter Name:</td>
      <td>
       <input type="text" name="name" />
       <span style="color: red" id="nameloc"></span>
      </td>
     </tr>
     <tr>
      <td>Enter Password:</td>
      <td>
       <input
        type="password"
        name="password"
        pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}"
       />
       <span style="color: red" id="passwordloc"></span>
      </td>
     </tr>
     <tr>
      <td>Enter Email:</td>
```



```
      <td>
       <input type="email" name="email" />
       <span style="color: red" id="emailloc"></span>
      </td>
     </tr>

     <tr>
      <td>Enter Age:</td>
      <td>
       <input type="text" name="age" />
       <span style="color: red" id="ageloc"></span>
      </td>
     </tr>
     <tr>
      <td colspan="2"><input type="submit" value="register" /></td>
     </tr>
    </table>
   </form>
  </div>
  <script src="./index.js"></script>
 </body>
</html>
```

Index.js

```
function validate()
{
 var name = document.f1.name.value;
 var password = document.f1.password.value;
 var minNumberofChars = 6;
 var maxNumberofChars = 16;
 var regularExpression =
  /^(?=.*[0-9])(?=.*[!@#$%^&*])[a-zA-Z0-9!@#$%^&*]{6,16}$/;
 var email = document.f1.email.value;
 var atposition = email.indexOf("@");
 var dotposition = email.lastIndexOf(".");
 var age = document.f1.age.value;
 var numbers = /^[0-9]+$/;
 var status = false;
 if (name.length < 1)
{
  document.getElementById("nameloc").innerHTML = " Please enter your
     name";
```

```
    status = false;
  }
 else
 {
   document.getElementById("nameloc").innerHTML = "";
   status = true;
  }
  if (password.length < 8)
  {
   document.getElementById("passwordloc").innerHTML =
    " Password must be at least 8 char long, at least 1 capital letter and one
      special character and one number";
   status = false;
  }
 else
 {
   document.getElementById("passwordloc").innerHTML = "";
  }
  if
  (
   atposition < 1 ||
   dotposition < atposition + 2 ||
   dotposition + 2 >= x.length
  )
  {
   document.getElementById("emailloc").innerHTML = "please enter valid
       email";
   status = false;
  }
  if (age.match(numbers))
  {
   document.getElementById("ageloc").innerHTML = "";
   status = true;
  }
 else
{
   document.getElementById("ageloc").innerHTML =
    "Please enter age between 8 to 60";
   status = false;
  }
  return status;
}
```


3. **How external CSS is inserted in a HTML page. Create a HTML page with two paragraph tags with id p1 and p2 and one div tag with id dv1. Write external CSS file as per the description; the margin of the paragraphs should be 100px from top and 80px from left. Both of paragraphs should have display set to block. The dv1 should have padding left and right set to 30px and 25px respectively, its background color should be blue and font color should be red with font type Verdana. Set the display of dv1 to none.** **[2+8]**

Ans: index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="style.css">
  <title>Collegenote</title>
</head>
<body>
  <div class="cn">
    The New summit college is one of the oldest and Pioneer IT College.
  </div>
  <p id="p1">
  New Summit College mainly focus on Practical knowledge and real life
     practices to the students.
  </p>
  <p id="p2">
    Students of B.Sc. CSIT, BCA and BBM are technical students.
  </p>
</body>
</html>
```

style.css

```
.cn{
    position:fixed;
    border: 2px solid black;
  }

#p1{
    position: relative;
    font-family: Arial;
    color: red;
  }

#p2{
    position:absolute;
    top: 0px;
    left: 200px;
  }
```

Section B

Attempt any eight questions. (8 × 5 = 40)

4. What are the services provided under web 2.0? [5]

Ans: Web 2.0 are websites and applications that make use of user-generated content for end-users. Web 2.0 is characterized by greater user interactivity and collaboration, more pervasive network connectivity and enhanced communication channels.

Web 2.0 does not refer to any specific technical upgrades to the internet. It simply refers to a shift in how the internet is used in the 21st century. In the new age, there is a higher level of information sharing and interconnectedness among participants. This new version allows users to actively participate in the experience rather than just acting as passive viewers who take in information.

The term **Web 2.0** is associated with web applications that facilitate participatory information sharing, interoperability, user-centered design, and collaboration on the World Wide Web. Web 2.0, a second advanced generation of WWW, is about revolutionizing the way of creating, editing, and sharing user generated content online. Web 2.0 is one of the series of improved technology rather than a specific version of web. It is characterized specifically as a transition from Static web pages to highly Dynamic web pages or user generated content. A Web 2.0 site allows users to interact and collaborate with each other in a social media dialogue as creators of user-generated content in a virtual community, in contrast to websites where users are limited to the passive viewing of content that was created for them. Examples of Web 2.0 include social networking sites, blogs, wikis, video sharing sites, hosted services, web applications.

Features of Web 2.0

- Web 2.0 uses the approach of "guide on the Side" rather than implementing "top-down" approach i.e dynamically change or edit the content rather then simply reading.
- It changed the concept of "mostly read only web" to "widely read and write" over web.
- Web 2.0 provides a perfect platform base for effective user interaction that was not available before.
- It changed the idea from passive consumption and delivery of content, to actively participating in creation, sharing, and collaboration.
- It is subjected to be a powerful lure for an Enterprise; that fetch more employees into accounts at a lower cost for greater participation in projects and idea sharing.

Advantages of Web 2.0:

- Available at any time, any place.
- Variety of media.
- Ease of usage.

- Learners can actively be involved in knowledge building.
- Can create dynamic learning communities.
- Everybody is the author and the editor, every edit that has been made can be tracked.
- User friendly.
- Updates in wiki are immediate and it offers more sources for researchers.
- Provides real-time discussion.

5. Create an HTML page with its body containing a div. The div should contain an image within it. Create a link on the image to redirect to the url http://www.tuiost.edu.np. Set the title of the page to "iost". Add Meta tag on the page. [5]

Ans:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="keywords" content="iost, csit">
  <title>IOST</title>
</head>
<body>
  <header style="background-color: darkgray; margin-bottom: 2px;">
    <nav>
      <a href="#">Home</a>
      <a href="#">About</a>
      <a href="#">Contact</a>
    </nav>
  </header>
  <article>
    Institute of Science and Technology (IoST) is one of the oldest and the
    largest technical institutes in TU.
  </article>
  <footer style="background-color: darkgray; margin-top: 2px; text-align:
    center;">
    <a href="info@iost.edu.np">Send Email</a>
  </footer>
</body>
</html>
```

6. How JQuery id selector can be used to select specific element? Write an example with JQuery that will hide a paragraph on clicking a button. [2+3]

Ans: jQuery uses CSS-style selectors to select parts, or elements, of an HTML page. It then lets you do something with the elements using jQuery methods, or functions.

The CSS ID selector applies styles to a specific html element. The CSS ID selector must match the ID attribute of an HTML element. Unlike classes, which can be applied to multiple elements throughout a site, a specific ID may only be applied to one single element on a site. CSS ID will override CSS Class properties. To select an element with a specific id, write a hash (#) character, followed by the id of the element.

Syntax

```
#specified_id {/* styles */ }
```

Program part,

```
<!DOCTYPE html>
<html>
<head>
<script
  src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js">
</script>
<script>
$(document).ready(function(){
 $("#hide").click(function(){
  $("p").hide();
 });
 $("#show").click(function(){
  $("p").show();
 });
});
</script>
</head>
<body>
<h2>jQuery show and hide example</h2>
<p> New Summit College is one of the oldest IT College </p>
<button id="hide">hide</button>
<button id="show">show</button>
</body>
</html>
```

7. Create a HTML page containing ordered and unordered list. Set the value of an ordered list type to "A". The list should start at "D". [5]

Ans: Unordered List

The list which are not ordered by numbers are known as unordered list. An unordered list starts with the <ul> tag. Each list item starts with the <li> tag. The list items will be marked with bullets by defaults however it can be changed. To change the bullets an attribute type is added. The value of type are disc,circle,square,none. Like <ul type="cirlce"> The example following shows the unordered list

```
<html>
<body>
```



```
<h2>An unordered HTML list </h2>
<ul>
    <li> Coffee </li>
    <li> Tea <li>
    <li> Milk </li>
</ul>
</body>
<lhtml>
```

Output:

An unordered HTML list

- Coffee
- Tea
- Milk

Ordered List

The list which are ordered by numbers are known as ordered list. An ordered list starts with the <ol> tag. Each list item starts with the <li> tag. The list items will be marked with numbers by default. The ordered list has the type attribute of the <ol> tag, defines the type of the list item marker. The following table shows the type attribute with the descriptions

Type	Description
type="1"	The list items will be numbered with numbers (default)
type="A"	The list items will be numbered with uppercase letters
type="a"	The list items will be numbered with lowercase letters
type="i"	The list items numbered with lowercase roman numbers
type="I"	The list items numbered with uppercase roman numbers

Example: List starting with numbers

```
<html>
<body>
  <ol start="D">
    <li>Nepal</li>
    <li>India</li>
    <li>China</li>
  </ol>
</body>
</html>
```

Output:

D. Nepal
E. India
F. China

8. Discuss about different JSON data types. [5]

Ans: JSON supports mainly 6 data types:

- string
- number
- boolean
- null
- object
- array

String: JSON strings must be written in double quotes like C-language there are various special characters(Escape Characters) in JSON that you can use in strings such as \ (backslash), / (forward slash), b (backspace), n (new line), r (carriage return), t (horizontal tab) etc.

Example:

```
{"name":"Rajan"}
{"city":"Ktm\/Nepal"}
```

Here \/ is used for Escape Character / (forward slash).

Number: Represented in base 10 and octal and hexadecimal formats are not used.

Example:

```
{"age": 20}
{"percentage": 82.44}
```

Boolean: This data type can be either true or false.

Example:

```
{"result": true}
```

Null: It is just a define nullable value.

Example:

```
{
 "result" : true,
 "grade" : null,
 "rollno" : 210
}
```

Object: It is a set of name or value pairs inserted between {} (curly braces). The keys must be strings and should be unique and multiple key and value pairs are separated by a, (comma).

Syntax:

```
{key: value, .....}
```

Example:

```
{
        "Geek":{ "name":"Peter", "age":20, "score": 50.05}
}
```

Array: It is an ordered collection of values and begins with [(left bracket) and ends with] (right bracket). The values of array are separated by , (comma).

Syntax:

```
[Value ...]
```

Example:

```
{
"geek":[ "Sahil", "Rajan", "Karki" ]
}
{
 "collection" : [
     {"id" : 101},
     {"id" : 102},
     {"id" : 103}
]
```



```
}
```

Example of JSON document:

```
{
 "Geeks" : [
  {
   "Geekname" : "Sahil kumar",
   "subject" : "Data structures",
   "Articles" : 10
  },
  {
   "Geekname" : "Pawan singh",
   "subject" : "Algorithms",
   "Articles" : 16
  },
  {
   "Geekname" : "Ayush Goel",
   "subject" : "Networking",
   "Articles" : 7
  }
 ]
}
```

9. What is DTD? Create a XML file and write its equivalent DTD. **[1+4]**

Ans: A Document Type Definition (DTD) describes the tree structure of a document and something about its data. It is a set of markup affirmations that actually define a type of document for the SGML family, like GML, SGML, HTML, and XML.

A DTD can be declared inside an XML document as inline or as an external recommendation. DTD determines how many times a node should appear, and how their child nodes are ordered.

There are 2 data types, PCDATA and CDATA

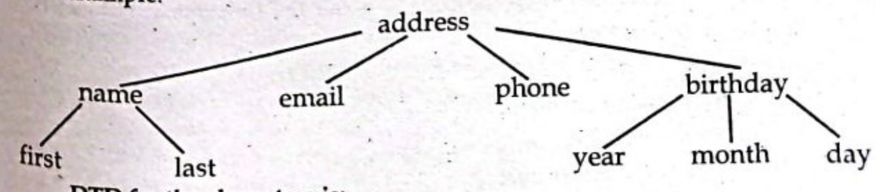
- PCDATA is parsed character data.
- CDATA is character data, not usually parsed.

Syntax:

```
<!DOCTYPE element DTD identifier
[
  first declaration
  second declaration

  .
  .
  nth declaration
]>
```

Example:



DTD for the above tree is:

XML document with an internal DTD:

```
<?xml version="1.0"?>
```

```
<!DOCTYPE address [
<!ELEMENT address (name, email, phone, birthday)>
<!ELEMENT name (first, last)>
<!ELEMENT first (#PCDATA)>
<!ELEMENT last (#PCDATA)>
<!ELEMENT email (#PCDATA)>
<!ELEMENT phone (#PCDATA)>
<!ELEMENT birthday (year, month, day)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT month (#PCDATA)>
<!ELEMENT day (#PCDATA)>
]>

<address>
  <name>
    <first>Rohit</first>
    <last>Sharma</last>
  </name>
  <email>sharmarohit@gmail.com</email>
  <phone>9876543210</phone>
  <birthday>
    <year>1987</year>
    <month>June</month>
    <day>23</day>
  </birthday>
</address>
```

10. How can you define function in PHP? Create a function that will take two integers as argument and will return product of them. **[2+3]**

Ans: In creating a large program with a huge set of programs the whole code becomes clumsy and which makes it look like a big mystery instead of having simple code. To solve this problem we use functions concept.

It is also in programs where we have to perform certain functionality repeatedly instead of writing the code for that many times we can simply create a function and store the code and call the function. It reduces both lines of code and time.

In PHP functions are of 2 types.

- Pre-Defined Functions
- User-Defined Functions

PHP Pre-Defined Functions:

These are the in-built functions which are already defined in PHP. This provides the basic functionality for PHP.

Example: any( ), bin( ).

PHP User-defined Functions:

These are the functions which user built to use them in their programs.

Syntax:

function function_name()

To pass parameters to functions we just pass them in ( ) after the function name.

Syntax:

function function_name(parameter1,parameter2...)

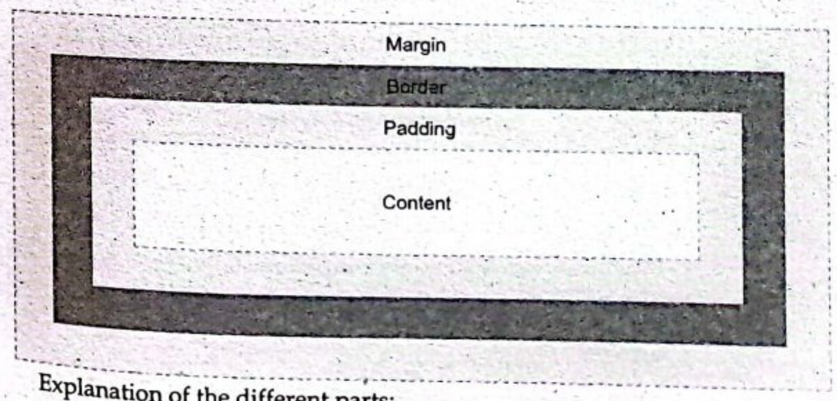
PHP function to find products of two numbers

We are going to multiply 2 numbers using a function named as mult.

```
<?php
function mult(int $x, int $y)
{
  return $x + $y;
}
//calling the function and printing the result
echo " product of 10 and 25 : " . mult (10, 25);
echo "<br>";
echo " product of 30 and -10: " . mult (30, -10);
echo "<br>";
?>
```

11. Describe CSS Box model with example. **[5]**

Ans: In CSS, the term "box model" is used when talking about design and layout. The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content. The image below illustrates the box model:



Explanation of the different parts:

- Content - The content of the box, where text and images appear
- Padding - Clears an area around the content. The padding is transparent
- Border - A border that goes around the padding and content
- Margin - Clears an area outside the border. The margin is transparent

Example: The box model allows us to add a border around elements, and to define space between elements.

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
 background-color: lightgrey;
 width: 300px;
 border: 15px solid green;
 padding: 50px;
 margin: 20px;
}
</style>
</head>
<body>
<h2>Demonstrating the Box Model</h2>
<p>The CSS box model is essentially a box that wraps around every HTML element. It consists of: borders, padding, margins, and the actual content.
</p>
<div> This text is the content of the box. We have added a 50px padding, 20px margin and a 15px green border. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
</div>
</body>
</html>
```


MODEL QUESTIONS SETS FOR PRACTICE

MODEL SET 1

Course Title: Web Technology **Full Marks: 60**
Course No: CSC318 **Pass Marks: 24**
Semester: V **Time: 3 hrs**

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Describe in details about the Tier Technology and its Architecture.
2. Write the structure of a XML file with example. Write an XML and DTD to describe "weather_report" as an element and "date, location, city, state, and temperature_range" as its attributes.
3. Make a simple Web site that takes information about the user and stores the information in a database. Use client-side script to validate the user input

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. What is Internet? Discuss some of its services.
5. Discuss different ways of inserting style sheets in HTML documents.
6. Discuss the use of Cookies with suitable example.
7. What are HTTP Protocol Methods? Explain.
8. What is HTML DOM? Explain some DOM methods used in web technology.
9. What is XSLT? Explain the XSL<xls: choose> Element.
10. What is Session? Explain its use with suitable example.
11. What are anti-overload techniques in web Server?
12. Discuss about Tag Libraries.

MODEL SET 2

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Explain different types of statements available in PHP.

2. Explain the concept of Polymorphism, Inheritance and exception handling in Java with suitable example.
3. Which java script operator calculates the remainder by dividing two integers? Explain with example.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. Discuss file handling with suitable example.
5. Describe the scoping rules for the Java script.
6. Create a DTD to display daily schedule of user.
7. Write XML Schema for library information system.
8. Discuss the various terms related to Document Type Definition.
9. How the result set of MySQL be handled in PHP?
10. Write a PHP script to sort the elements of an array.
11. Explain the classification of HTML tags with examples.
12. Explain the role of web browser.

MODEL SET 3

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Explain different types of operators available in PHP.
2. State the advantages of XML. Write a detailed note on XML syntax rules, XML elements and XML attributes.
3. Differentiate cookies from session variables with suitable example.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. Explain the term WWW.
5. Explain the term Internet Addressing.
6. What are the Advantages of HTML? Explain the purpose of HTML Frames.
7. Prepare a form containing the field User ID, Password, and Account Type and three buttons **add, view** and **delete**. Account Type may be user or admin.
8. What is JavaScript? Are Java and JavaScript the Same?
9. Explain the Conditional operator in JavaScript with an example.
10. Explain the term PHP sessions.
11. What is the purpose of the XML DTD? Explain.
12. Write a brief note on Tools for Web site creation.

MODEL SET 4

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Write a detailed note on different types of operators available in JavaScript.
2. Write a JavaScript that scrolls a text message in the status bar of the browser window.
3. Explain the frames and table tags of HTML with suitable example.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. Write a PHP program to print reverse of any number.
5. Give a note on need and applications of XML.
6. Discuss AJAX architecture and compare it with DOM.
7. Write a brief note on Web Applications.
8. Write a short note on HTML Lists.
9. Describe different popup boxes supported by javascript briefly. Write a sample program to show use these popup boxes.
10. How can a cookie be created using Java script? Explain.
11. State the advantages of XML. Write a detailed note on XML syntax rules
12. Describe the use of html **Meta** and **div** tags in detail.

MODEL SET 5

Section A

Attempt any TWO questions. **(2 × 10 = 20)**

1. What is cookie? Explain the methods of setting and reading cookies using java script.
2. Explain 2-tier, 3-tier and n-tier client server architecture.
3. Write a XML document for storing records of 5 employees with ID, Name, Salary and Age. Create a XSLT file to display the record of employees having salary more than 20000.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. What is meant by dynamic HTML?
5. When the namespace is called in XML? Why?
6. Differentiate between 'Get' and 'Post' methods in PHP.
7. What are the major advantages and disadvantages of AJAX?

8. What is JavaScript? How it is different from server side scripting languages? Discuss uses of JavaScript briefly.
9. What are the drawbacks of HTML? How are they addressed in XML?
10. Elaborate Internet addressing and types of Internet connections.
11. Explain the concept of Exception Handing. How will you create your own exception?
12. What is the use of XML namespaces? Explain with suitable example.

MODEL SET 6

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. Why XSD is used? How it is different from DTD? Write XSD and DTD for Book, where book contains author, publication, Title, price and pages. Further Author contains First Name, Last Name, and Education in order and can occur 4 times maximum and 1 times minimum. Define root element of your own interest
2. What are two methods of validating the forms? Highlight their good and bad points. Prepare a form containing the controls Name, Phone, and Email. Validate the form such that Name field contains only text, Phone field contains only digits, and Email field contains valid email id.
3. Prepare a form containing the field User ID, Password, and Account Type and two buttons add, and view. Account Type may be user or admin. write ASP.net code for adding new account to the table user account and displaying all admin users from the table.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. Describe the scoping rules for the Java script.
5. Create a HTML which uses CSS that gives all H1 and H2 elements a padding of 0.5 ems; a grooved border style and a margin of 0.5 ems.
6. Write a short note on creating tables in HTML.
7. Write a short note on JavaScript DOM.
8. What is AJAX? What are the advantages of AJAX? Write a brief note on AJAX Database.
9. Differentiate between SMTP and POP.
10. Why CSS are important in web designing? Describe the use of id and class selectors with suitable example.
11. Write the JAVA script to print "Good-Day" using IF-ELSE condition.
12. Explain that how web servers executes .aspx files

MODEL SET 7

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. What is Document object model? Discuss the various DOM methods used with java script.
2. What are the benefits of using styles compared with placing formatting directly into the text of the Web page?
3. Write a script that reads an integer and determines whether it is PRIME Number or Not.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. Create a DTD to display daily schedule of user.
5. Write XML Schema for library information system.
6. Discuss the various terms related to Document Type Definition.
7. How the result set of MySQL is handled in PHP?
8. Describe different popup boxes supported by java script briefly. Write a sample program to show use these popup boxes.
9. What types of primary data structures are supported in Perl? Discuss
10. Create a HTML form with five basic features.
11. Write about the various Objects used in Java script.
12. Define frame. Create a HTML page that displays multiple frames in a window.

MODEL SET 8

Section A

Attempt any two questions. **(2 × 10 = 20)**

1. 'JavaScript is referred to as Object based programming language'. Justify with an example.
2. Which java script operator calculates the remainder by dividing two integers? Explain with example
3. What is client/server technology? Differentiate between web client and web server.

Section B

Attempt any eight questions. **(8 × 5 = 40)**

4. How can you draw a table in HTML? Discuss with suitable example.
5. How can you insert external style sheets in your HTML document? Discuss with example.

6. Write a simple client side script to set and retrieve cookies.
7. What is FTP?
8. Discuss public external declaration of DTD with example.
9. Define session. Explain its use with example.
10. What are anti-overload techniques in web server?
11. Write a script that asks the user to enter two numbers, obtains the two numbers from the user and outputs text that displays the sum, product, difference and quotient of the two numbers.
12. Explain the classification of HTML tags with examples.

Best Wishes To All
Of You.

Lokaj.