# SNS COLLEGE OF TECHNOLOGY

**Coimbatore-35**
**An Autonomous Institution**

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

## DEPARTMENT OF INFORMATION TECHNOLOGY

## 16IT302 – DESIGN AND ANALYSIS OF ALGORITHMS

II YEAR IV SEM

UNIT-II-BRUTE FORCE AND DIVIDE AND CONQUER

TOPIC: Traveling Salesman Problem

Prepared by
T.Shanmugapriya,AP/IT

# BRUTE FORCE TRAVELING SALESMAN PROBLEM

**Subject :Design and Analysis of Algorithm**
**Unit :II**

# EXHAUSTIVE SEARCH

- Traveling Salesman Problem
- Knapsack Problem
- Assignment problem

.

# *Exhaustive search*

***Exhaustive search*** is simply a brute-force approach to combinatorial problems such as permutation, combination and sum of subsets

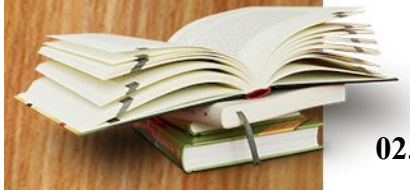
**The method**:

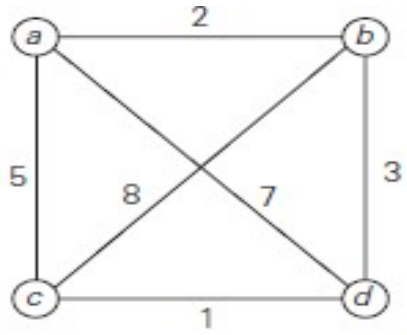
- Generate a list of all the potential solutions to the problem
- selectthe solution that satisfy all the constraints, and then
- Find the desiredsolution that optimizes some objective function.

# Traveling Salesman Problem

The problem asks to find the shortest tour (Hamiltonian circuit) through a given set of *n* ci ties that visits each city exactly once before returning to the city where it started. The pro blem can be conveniently modeled by a weighted graph, with the graph's vertices repre senting the cities and the edge weights specifying the distances.

It is easy to see that a Hamiltonian circuit can also be defined as a sequence of n + 1adjac ent vertices vi0, vi1, . . . , vin−1, vi0, where the first vertex of the sequence is the same as the last one and all the other n − 1 vertices are distinct.






| Tour | Length | |
|---|---|---|
| $a \rightarrow b \rightarrow c \rightarrow d \rightarrow a$ | $l = 2 + 8 + 1 + 7 = 18$ | |
| $a \rightarrow b \rightarrow d \rightarrow c \rightarrow a$ | $l = 2 + 3 + 1 + 5 = 11$ | optimal |
| $a \rightarrow c \rightarrow b \rightarrow d \rightarrow a$ | $l = 5 + 8 + 3 + 7 = 23$ | |
| $a \rightarrow c \rightarrow d \rightarrow b \rightarrow a$ | $l = 5 + 1 + 3 + 2 = 11$ | optimal |
| $a \rightarrow d \rightarrow b \rightarrow c \rightarrow a$ | $l = 7 + 3 + 8 + 5 = 23$ | |
| $a \rightarrow d \rightarrow c \rightarrow b \rightarrow a$ | $l = 7 + 1 + 8 + 2 = 18$ | |

**Analysis:**

To reduce the number of computation, choose any two intermediate vertices, say, b and c, and then consider only permutations in which b precedes c.

T(n)= $\frac{1}{2}(n-1)!$

# Thank you!