SNS COLLEGE OF ENGINEERING

Coimbatore-107

COURSE NAME: ANALYSIS OF ALGORITHM

II YEAR/ IV SEMESTER

UNIT – V

BACKTRACK ALGORITHM

Topic

Hamiltonian Circuit Problem

Back Tracking

Hamiltonian Circuit Problem

It is a path in graph that

*. Visits every vertex exactly once
*. Returns to starting vertex.

Vertex A :

	0	1	2	3
0	0	1	1	1
1	1	0	1	0
2	1	1	0	1
3	1	0	1	0

Meaning:

0 Connected to 1, 2, 3
1 Connected to 0, 2
2 Connected to 0, 1, 3
3 connected to 0, 2

1) Start from vertex 0

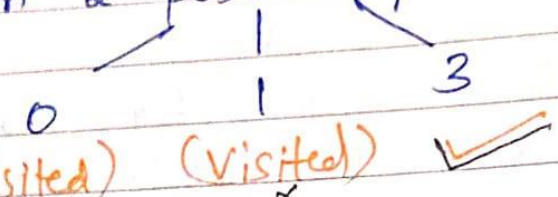
(2) Step 1: 0 → 1
Visited: 0 → 1 ✓

(3) Step 2: 1 → 2
From 1 → 0 (Already visited) ✗
→ 2 ✓

Visited: 1 → 2

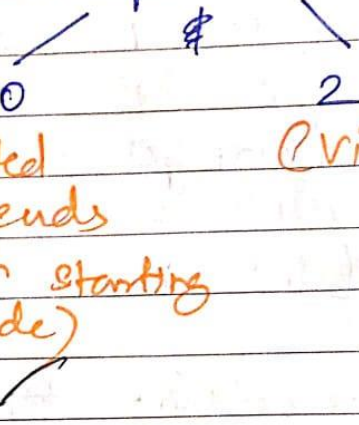
(3). Step 3: 2 → 3

From 2 possible options



(4) Step 4: 3 → 0

From 3 possible options



Rule:

⇒ We must visit every vertex exactly once.

⇒ We are allowed to return to the starting vertex at end to form circuit.

Logic:

(i) Start from 0.

(ii) Keep path list: Stores visited vertices

(iii) Keep visited list: Marks where you have

(iv) Try reaching neighbours: If they are not visited so there & continue

(v) check if path length == number of vertices
check:
* If there a connection back to 0
(to form circuit)
* If yes → Hamiltonian circuit is formed.
(vi) If stuck
↓
Do Back Tracking.

Algorithm: hamiltonian(graph, path, visited, pos)
if pos == len(graph);
// check if there is return edge to start
return path[0] in graph[path[-1]]
for vertex in graph[path[pos-1]];
if not visited[vertex];
visited[vertex] = True
if hamiltonian(graph, path, visited, pos)
return True;
// Back track
visited[vertex] = false
return false

Time Complexity: (NP complete problem)
It checks (n-1)! paths for graph of 'n' vertices.
∴ Time complexity = O(n!)

Space Complexity:

Uses Extra Space :

visited $O(n)$

path $O(n)$

$\therefore$ Space Complexity $= O(n)$