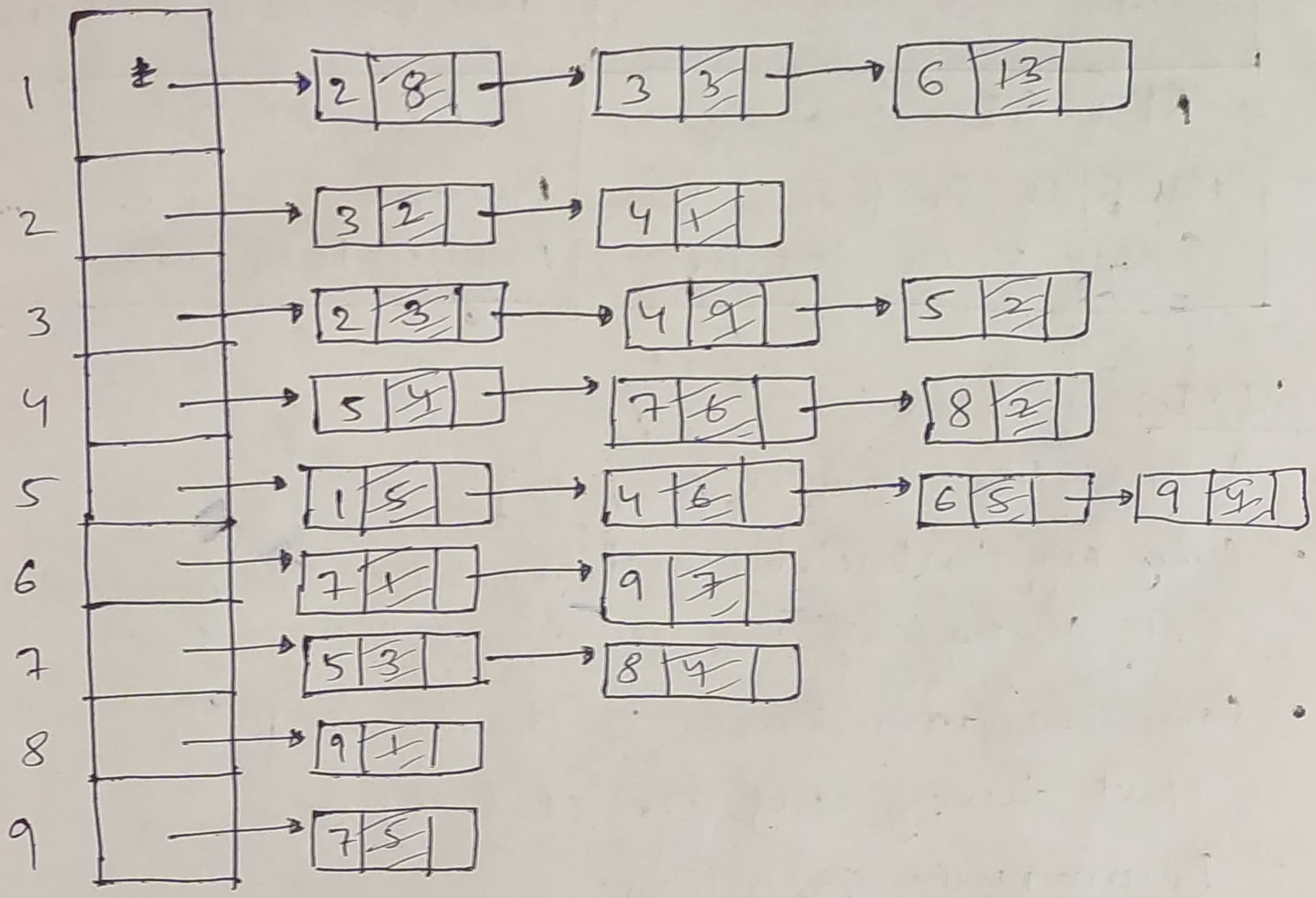
Ashutosh Jha

183129 11011 Advanced Algo

9 March, 2021

Q.1 (a) Adjacency List Representation



(b) Adjacency Matrix

$$\begin{bmatrix} * & 8 & 3 & * & * & 13 & * & * & * \\ * & * & 2 & 1 & * & * & * & * & * \\ * & 3 & * & 9 & 2 & * & * & * & * \\ * & * & * & * & 4 & * & 6 & 2 & * \\ 5 & * & * & 6 & * & 5 & * & * & 4 \\ * & * & * & * & * & * & 1 & * & 7 \\ * & * & * & * & 3 & * & * & 4 & * \\ * & * & * & * & * & * & * & * & 1 \\ * & * & * & * & * & * & 5 & * & * \end{bmatrix}$$

(c) In & Out Degree

Vertex	In	Out
1	1	3
2	2	2
3	2	3
4	3	3
5	3	4
6	2	2
7	3	2
8	2	1
9	3	1

Q.2

Kosaraju algorithm is used to find no. of strongly connected components in a graph.

SCC → is a maximal subset of vertices of C such that any two vertices of this subset are reachable from each other

i.e. for any $u, v \in C$

$u \mapsto v$, $v \mapsto u$

where

$\mapsto$ means reachibity i.e. existence of path from first vertex to the second.

Steps to find SCC Using Kosarajus's Algo.

1. Perform DFS. Push node to stack before returning
2. Find the transpose graph by reversing the edges.
3. Pop nodes one by one from the stack and again do DFS on modified graph.
   ↳ Each successful DFS will give us 1 SCC.

Time Complexity

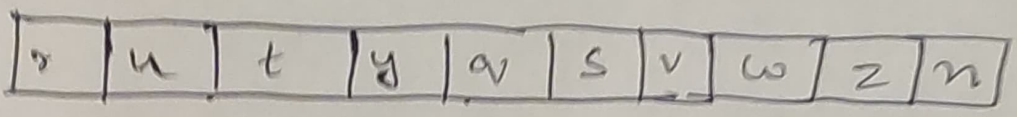
$3 \times O(V+E) \to O(V+E)$

↳ Traversing graph 3 times

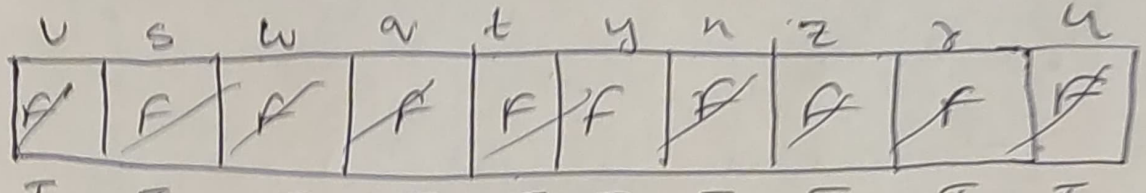
for the given graph

1) Starting Dfs from vertex v

Stack

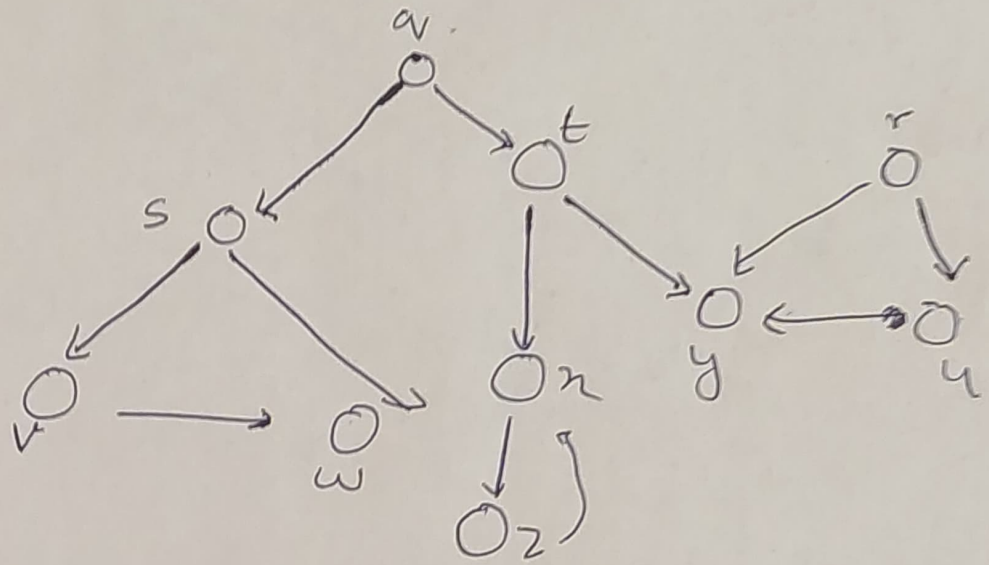


Visited



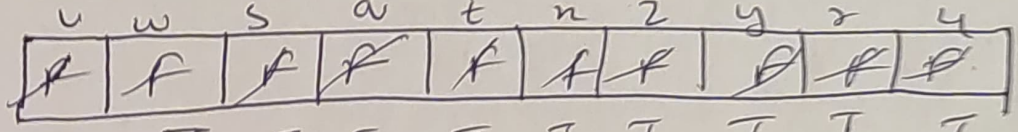
Order of DFS

Step-2 Transpose of Graph



Popping from stack & doing DFS

Visited



5 3 4 6 7 1 3 8 10 9 → Order

Doing DFS from vertex (n)

SCC → n, z (1st SCC)

2) Doing DFS on w

2nd SCC → v, w, s

3) Doing DFS on q

3rd SCC → q, t, y

4) Doing DFS on u

4th SCC → u

5) Doing DFS of r (topping last element)

5th SCC → r

So, the SCC are :-

1) n, z
2) v, w, s
3) q, t, y
4) u
5) r

Q3

Cost Adjacency Matrix of the graph is:

Nodes → / Start ↓	1	2	3	4	5	6	7
(1)$_1$	0	12	(10)$_1$	-	-	-	12
2	12	0	8	-	-	-	-
(3)$_2$	10	8	0	11	(3)$_2$	-	9
(4)$_5$	-	(12)$_5$	11	0	11	5	-
(5)$_3$	-	-	3	11	0	(6)$_3$	7
(6)$_4$	-	-	-	(5)$_4$	6	0	(5)$_4$
7	12	-	9	-	7	5	0

Step-1 · Starting from node 1, node 3 is selected as next node with least distance i.e. 10

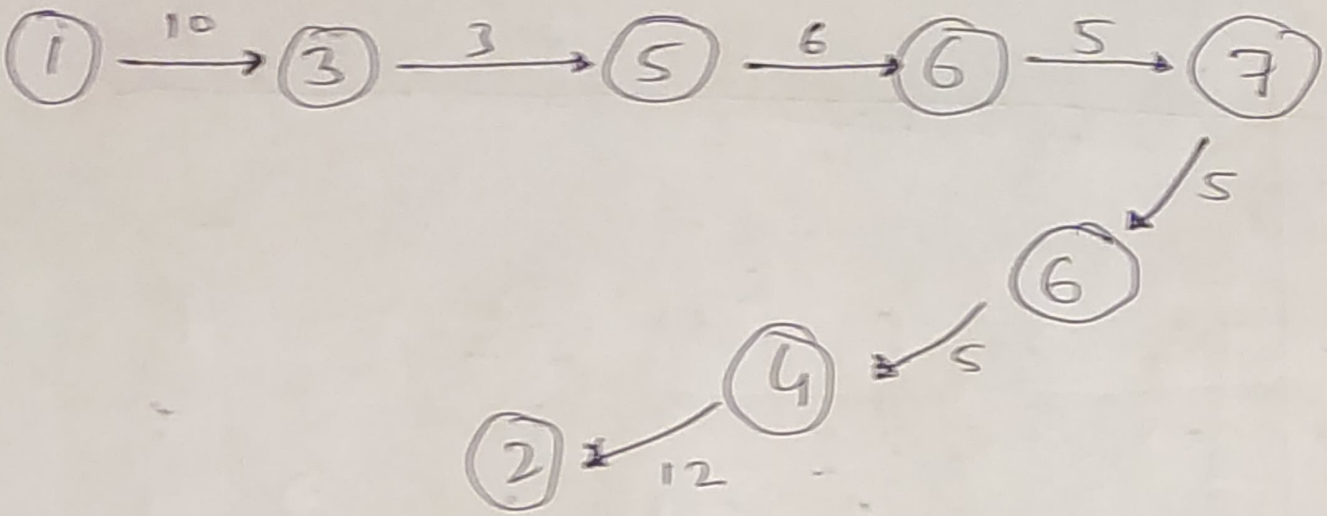
Step-2 Starting from node 3, node 5 is selected as next node with least distance i.e. 3

Step-3 Starting from node 5, node 6 is selected as next node with least distance i.e. 6

Step-4 Starting from node 6, node 7 is selected as next node and returned (5+5 = 10) and then node 4 is selected

Step-5 Starting from node 4, node 2 (which is also the not yet visited node) is selected which is 12 units away from node 4

So, route with minimum cost by travelling salesperson algorithm is obtained as -



So, Minimum Cost = 10 13 + 6 15 + 5 15 1 12

= 46 units

(b) Flynn's Classical Taxanomy

There are different ways to classify parallel computers among which Flynn's Taxanomy is one of the widely used classifications.

Flynn's Taxanomy distinguishes multi-processor computer architectures according to how they can be classified along the two independent dimensions of Instruction Stream and Data Stream. Each of these dimensions can have only one of two possible states : Single or Multiple

The matrix below defines the 4 possible classifications according to flynn

SISD Single Instruction Stream Single Data Stream	SIMD Single Instruction Stream Multiple Data Stream
MISD Multiple Instruction Stream Single Data Stream	MIMD Multiple Instruction Stream Multiple Data Stream

SISD:

- A serial (non-parallel computer)
- Only one instruction is being acted on by the CPU during any one clock cycle.
- Only one data stream is being used as input during any one clock cycle
- Deterministic execution

eg- Older Generation Mainframes

SIMD

- A type of parallel computer
- All processing units execute the same instruction
- Each processing unit can operate on a diff. data element.
- Synchronous & Deterministic execution

eg: Vector Pipelines.

MISD :

- A type of parallel computer
- Each processing unit operates on the data independently via seperate instructions stream
- A single data stream is fed into multiple processing units.

ex: Multiple Frequency filters

MIMD

- A type of parallel computer
- Every processor may be executing a diffrent instruction stream
- Every processor may be working with a diffrent data stream

ex: Most current supercomputers