



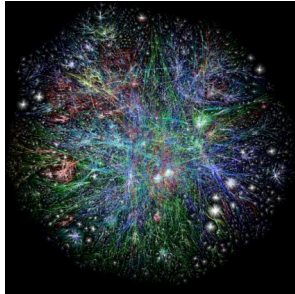
CS 225

Data Structures

*October 24 – Graph Implementations
(Adjacency Matrix)*

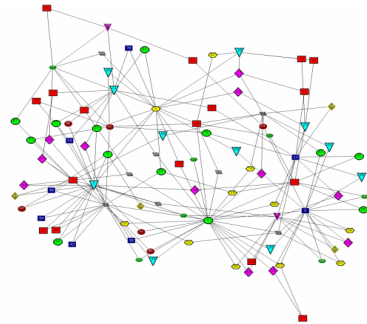
G Carl Evans

Graphs



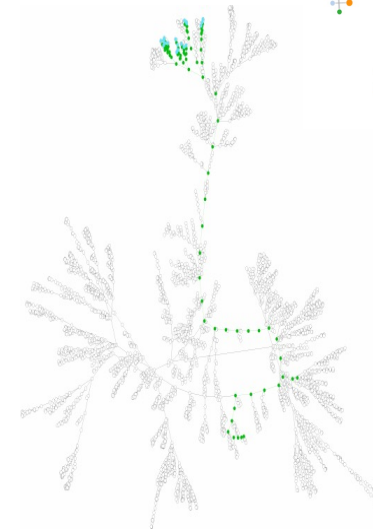
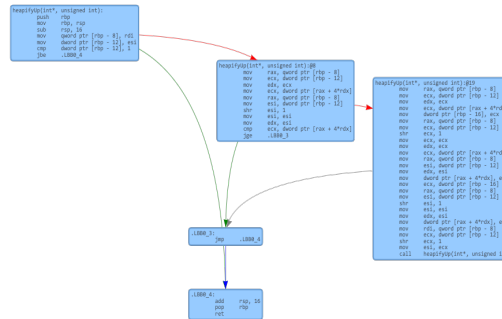
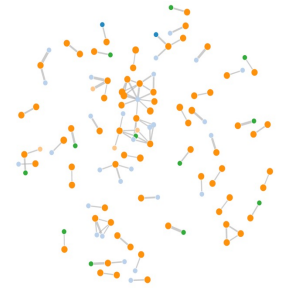
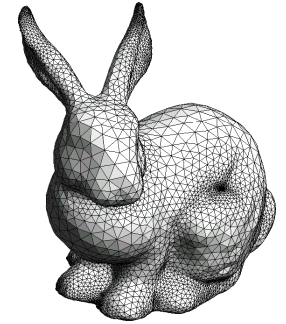
HAMLET

TROILUS AND CRESSIDA

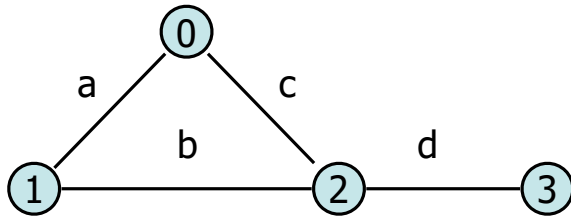


To study all of these structures:

1. A common vocabulary
2. Graph implementations
3. Graph traversals
4. Graph algorithms



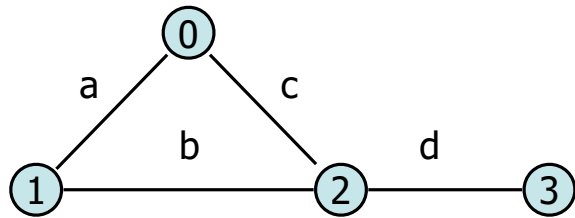
Graph Implementation: Edge List



0	0	1	a
1	1	2	b
2	0	2	c
3	2	3	d

- insertVertex(K key):** $O(1)$
- insertEdge(Vertex v1, Vertex v2, K key):** $O(1)$
- removeVertex(Vertex v):** $O(m)$
- incidentEdges(Vertex v):** $O(m)$
- areAdjacent(Vertex v1, Vertex v2):** $O(m)$

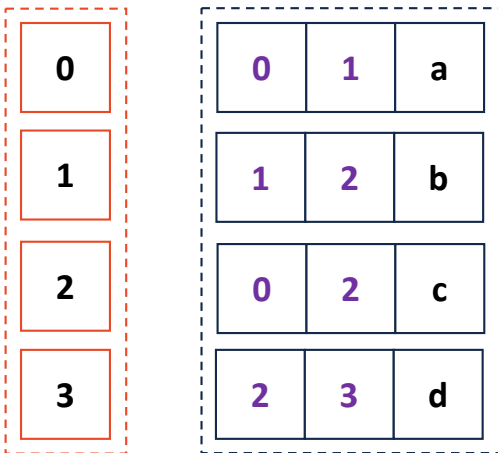
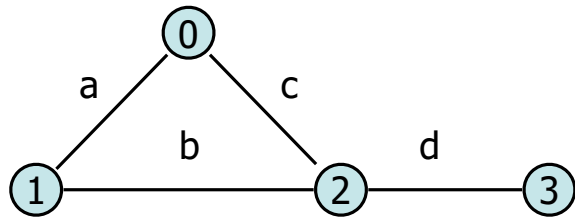
Graph Implementation: Adjacency Matrix



0	0	1	a
1	1	2	b
2	0	2	c
3	2	3	d

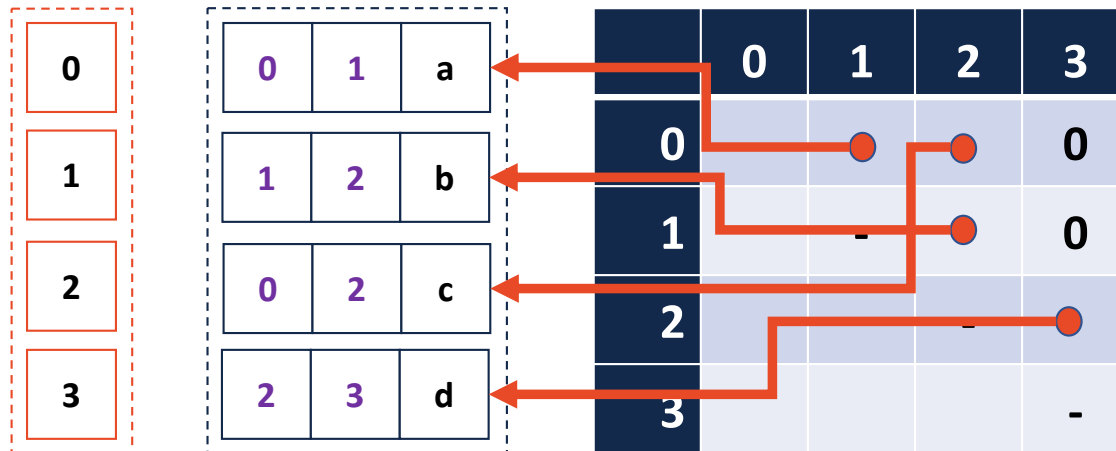
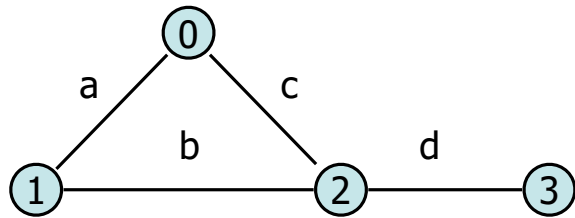
	0	1	2	3
0				
1				
2				
3				

Graph Implementation: Adjacency Matrix

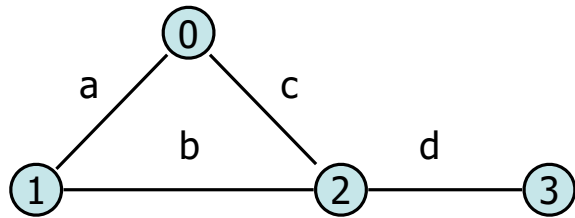


	0	1	2	3
0	-	1	1	0
1		-	1	0
2			-	1
3				-

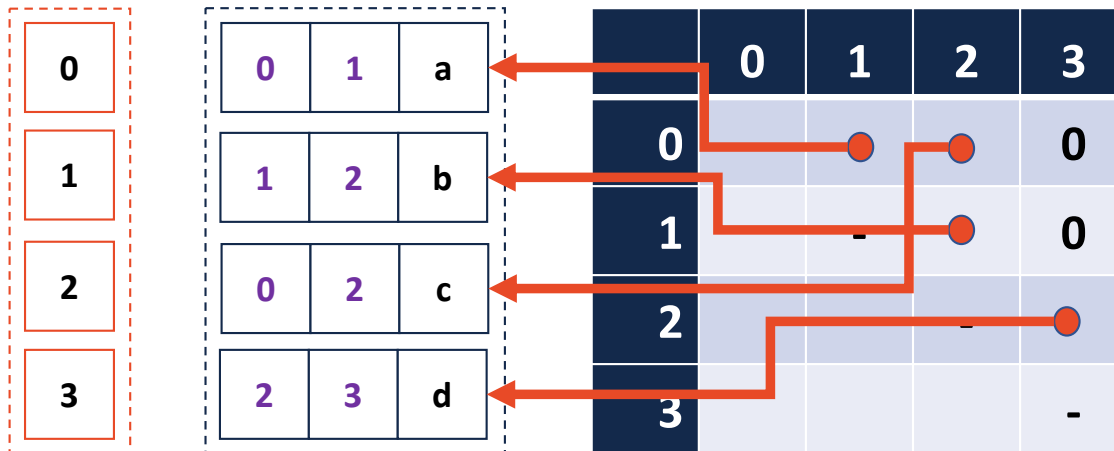
Graph Implementation: Adjacency Matrix



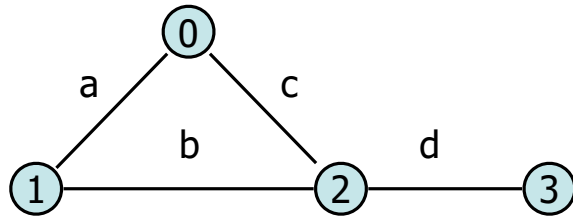
Adjacency Matrix



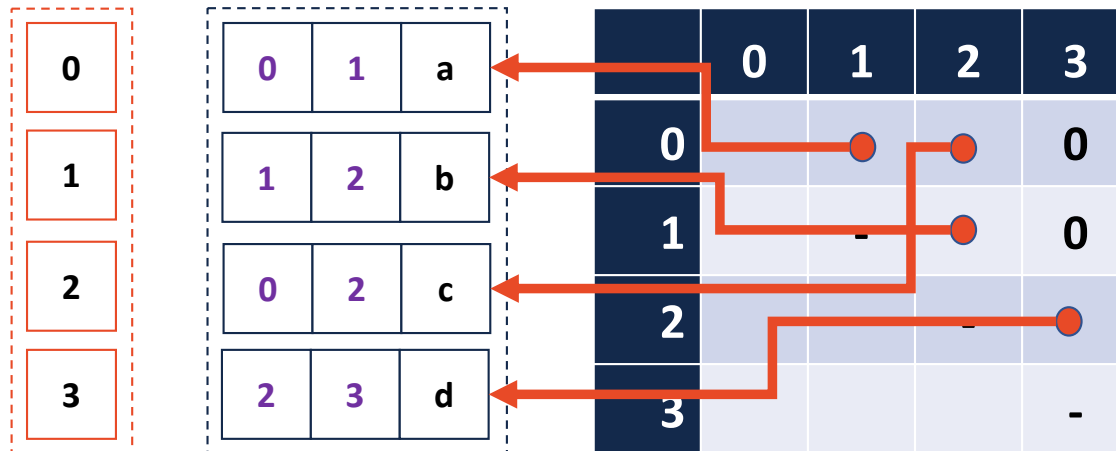
areAdjacent(Vertex v1, Vertex v2):



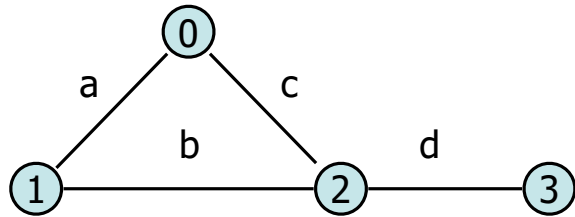
Adjacency Matrix



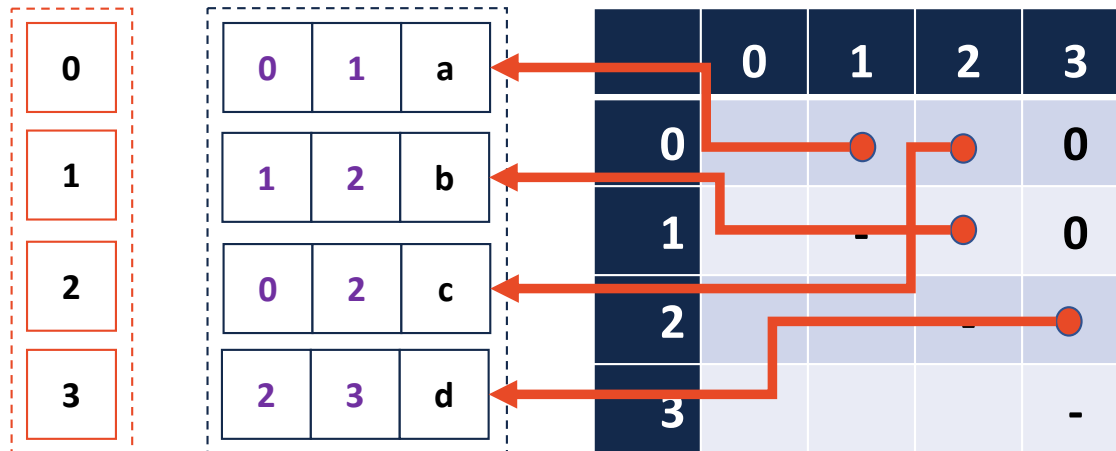
incidentEdges(Vertex v):



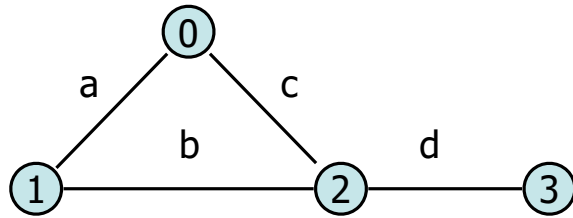
Adjacency Matrix



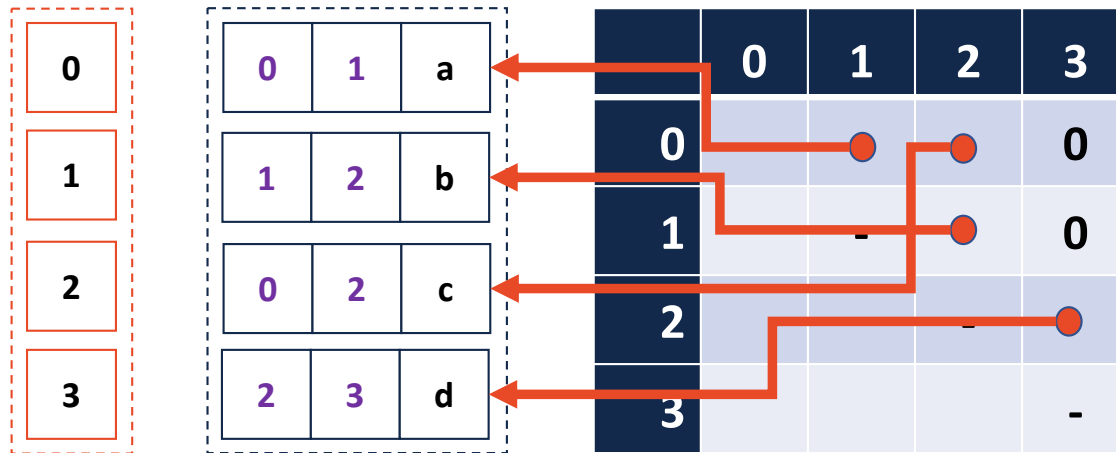
insertEdge(Vertex v1, Vertex v2, K key):



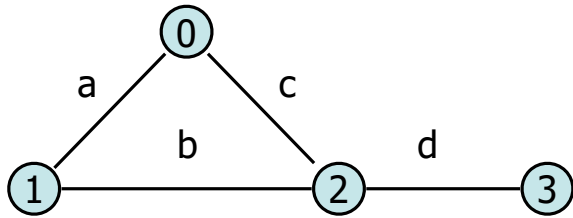
Adjacency Matrix



removeVertex(Vertex v1):

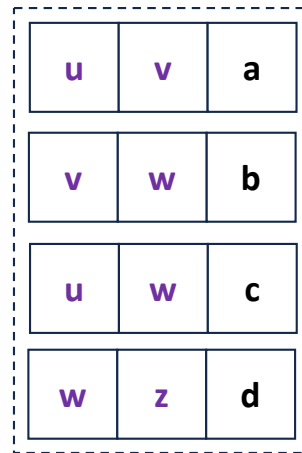
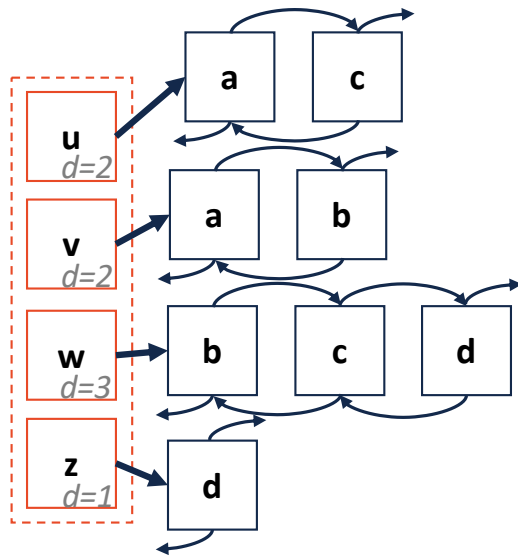
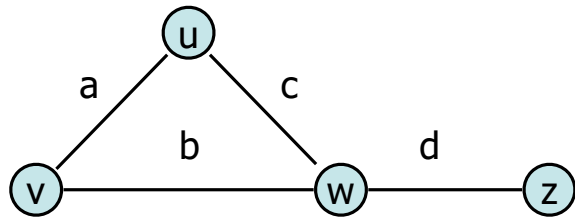


Edge List

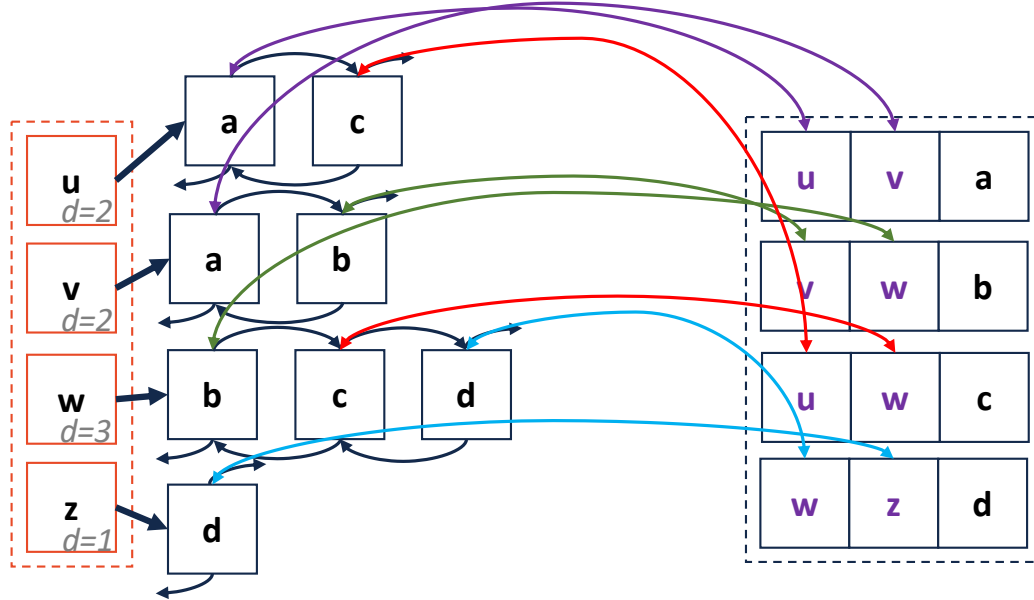
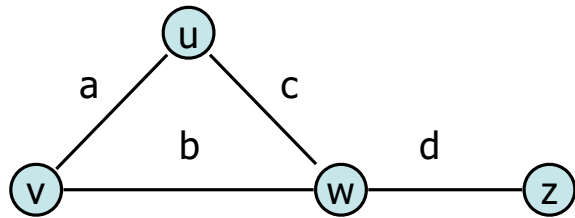


0	0	1	a
1	1	2	b
2	0	2	c
3	2	3	d

Graph Implementation: Adjacency List

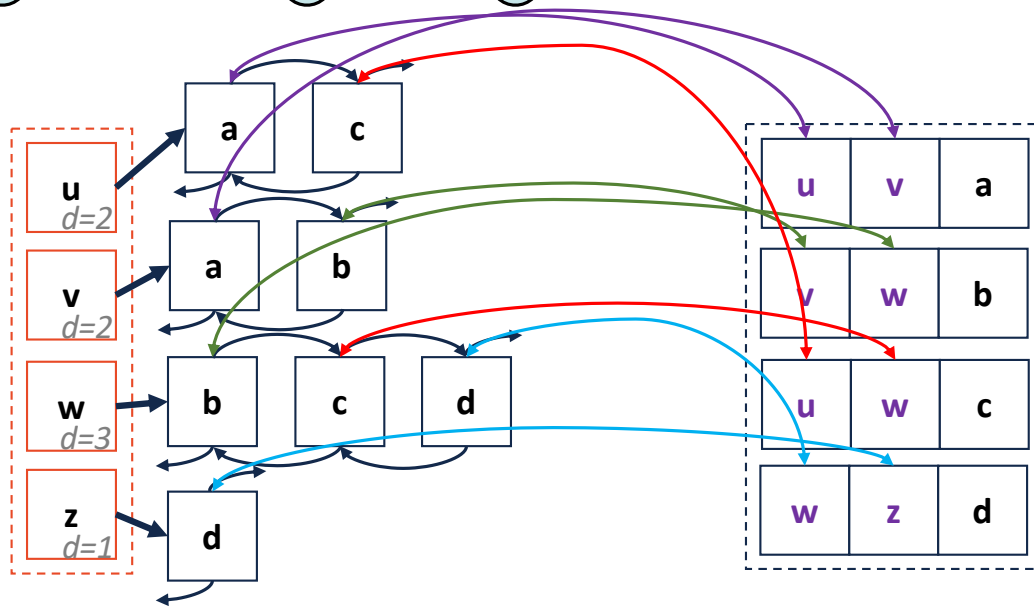
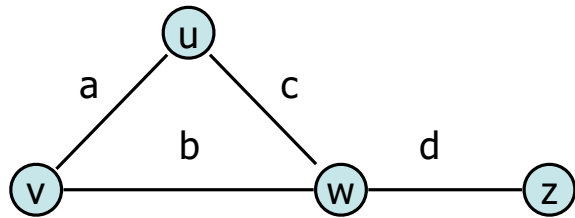


Graph Implementation: Adjacency List



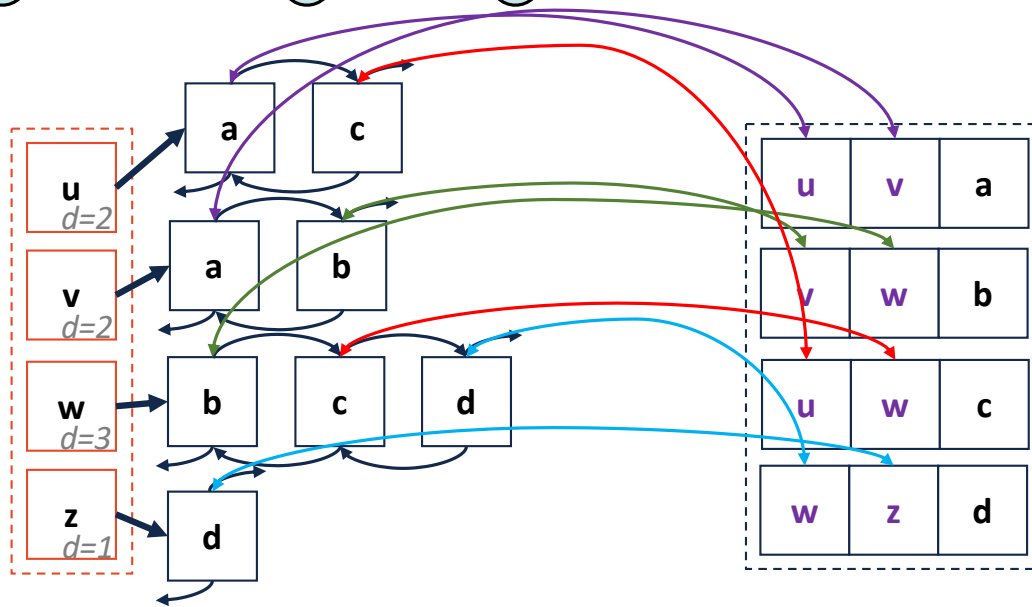
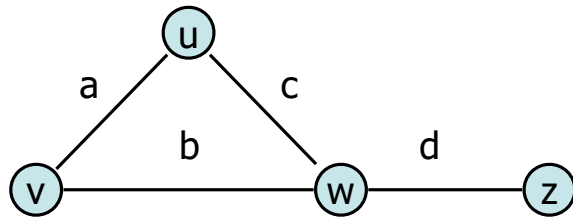
Adjacency List

insertVertex(K key):



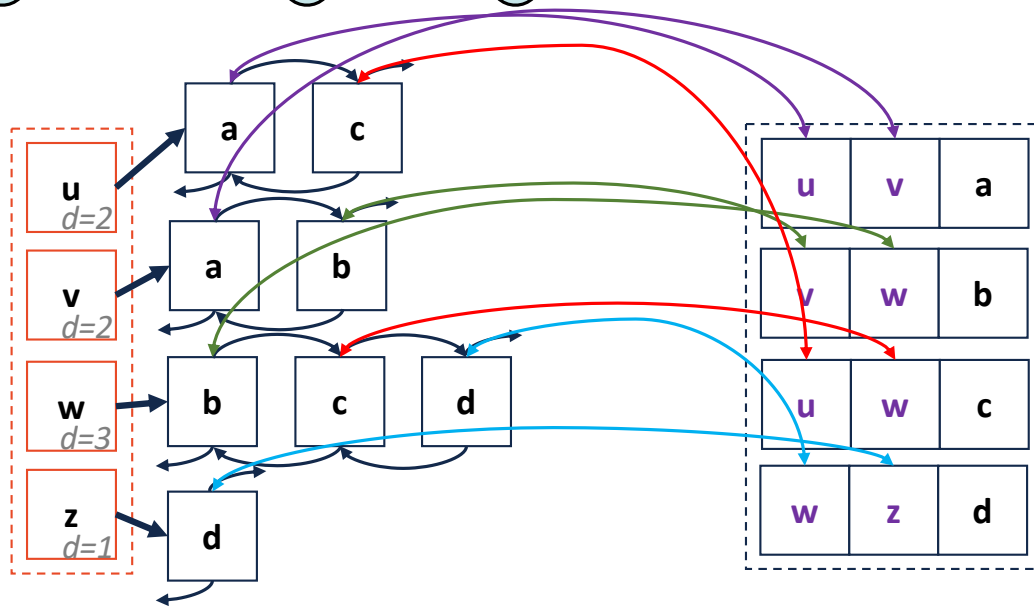
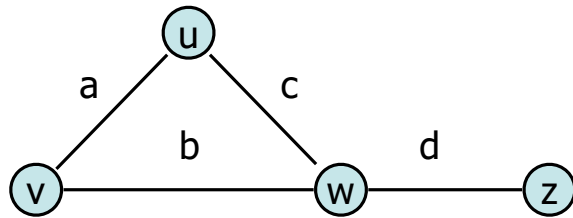
Adjacency List

removeVertex(Vertex v):



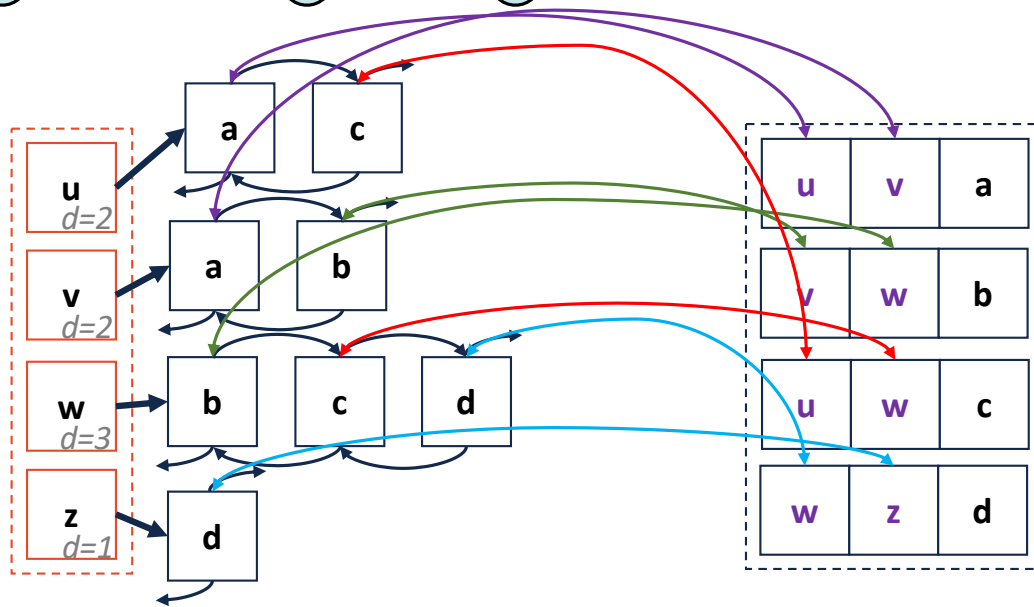
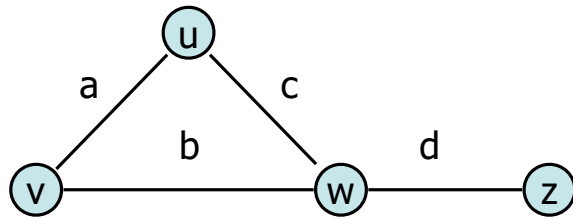
Adjacency List

incidentEdges(Vertex v):



Adjacency List

areAdjacent(Vertex v1, Vertex v2):



Adjacency List

insertEdge(Vertex v1, Vertex v2, K key):

