BFS for Single Source Shortest Path

Dr. Mattox Beckman

University of Illinois at Urbana-Champaign
Department of Computer Science

Objectives

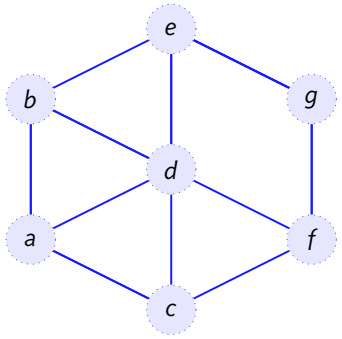
Your Objectives:

- Implement SSSP using BFS

The Algorithm

- Use this if your graph is unweighted
- Create a distance array and a parent array

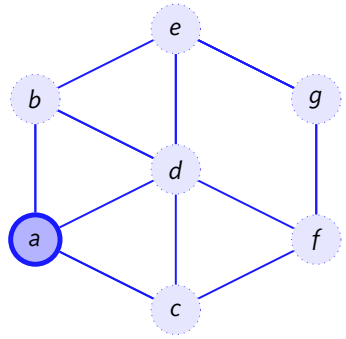
	a	b	c	d	e	f	g
Dist							
Parent							



The Algorithm

- Use this if your graph is unweighted
- Create a distance array and a parent array

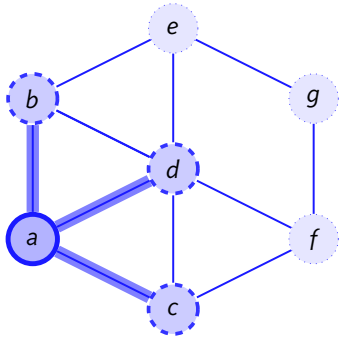
	a	b	c	d	e	f	g
Dist	0	∞	∞	∞	∞	∞	∞
Parent	-	-	-	-	-	-	-



The Algorithm

- Use this if your graph is unweighted
- Create a distance array and a parent array

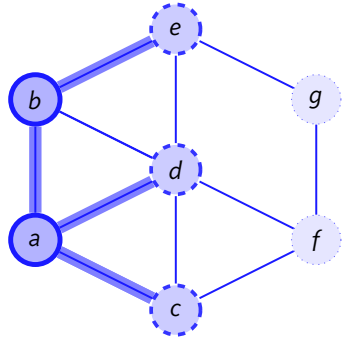
	a	b	c	d	e	f	g
Dist	0	1	1	1	∞	∞	∞
Parent	-	a	a	a	-	-	-



The Algorithm

- Use this if your graph is unweighted
- Create a distance array and a parent array

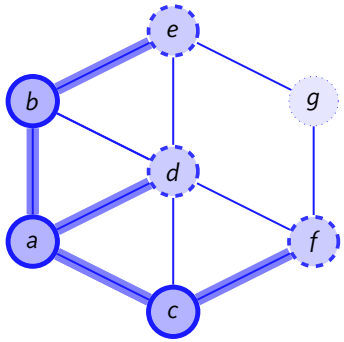
	a	b	c	d	e	f	g
Dist	0	1	1	1	2	∞	∞
Parent	-	a	a	a	b	-	-



The Algorithm

- Use this if your graph is unweighted
- Create a distance array and a parent array

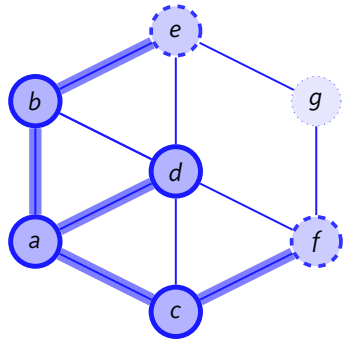
	a	b	c	d	e	f	g
Dist	0	1	1	1	2	2	∞
Parent	-	a	a	a	b	c	-



The Algorithm

- Use this if your graph is unweighted
- Create a distance array and a parent array

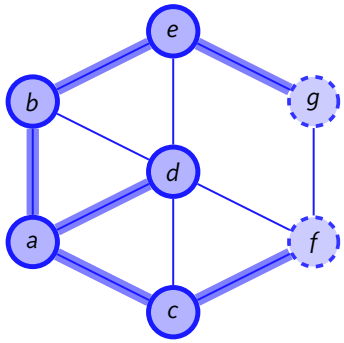
	a	b	c	d	e	f	g
Dist	0	1	1	1	2	2	∞
Parent	-	a	a	a	b	c	-



The Algorithm

- Use this if your graph is unweighted
- Create a distance array and a parent array

	a	b	c	d	e	f	g
Dist	0	1	1	1	2	2	3
Parent	-	a	a	a	b	c	e



Implementation

```
0 // Credit: Competitive Programming 3
1 vi dist(V, INF); dist[s] = 0;
2 queue<int> q; q.push(s);
3 vi parent;
4 while (!q.empty()) {
5   int u = q.front(); q.pop();
6   for (int j = 0; j < (int)AdjList[u].size(); j++) {
7     ii v = AdjList[u][j];
8     if (dist[v.first] == INF) {
9       dist[v.first] = dist[u] + 1;
10      parent[v.first] = u;
11      q.push(v.first);
12 } } }
```