Dijkstra's Algorithm Continued

E.W. Dijkstra (1930-2002)

1

Dijkstra's Algorithm: Pseudocode

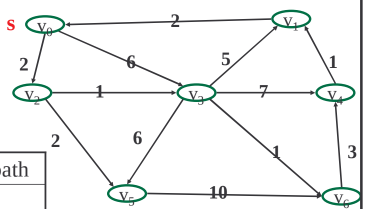
Initialize the cost of each node to ∞

Initialize the cost of the source to 0

While there are unknown nodes left in the graph
  Select an unknown node *b* with the lowest cost
  Mark *b* as known

For each node *a* adjacent to *b*
  a's cost = min(*a*'s old cost, *b*'s cost + cost of (*b*, *a*))

2



V	Known	Dist	path
v0			
v1			
v2			
v3			
v4			
v5			
v6			

3

```
void Graph::dijkstra(Vertex s){
  Vertex v,w;

   Initialize s.dist = 0 and set dist of all other
   vertices to infinity

   while (there exist unknown vertices, find the
   one b with the smallest distance)
     b.known = true;
   }
   for each a adjacent to b
       if (!a.known)
         if (b.dist + Cost_ba < a.dist){
           decrease(a.dist to= b.dist + Cost_ba);
           a.path = b;
       }
   }
}
```

4

Dijkstra's Alg: Implementation

Initialize the cost of each node to ∞
Initialize the cost of the source to 0
While there are unknown nodes left in the graph
  Select the unknown node *b* with the lowest cost
  Mark *b* as known
For each node *a* adjacent to *b*
  a's cost = min(*a*'s old cost, *b*'s cost + cost of (*b*, *a*))

What data structures should we use?
Operations to be performed:
  `deleteMin()`
  `decreaseKey()`

5

Dijkstra's vs BFS

At each step:
1) Pick closest unknown vertex
2) Add it to finished vertices
3) Update distances

Dijkstra's Algorithm

At each step:
1) Pick vertex from queue
2) Add it to visited vertices
3) Update queue with neighbors

Breadth-first Search

$O(|V|^2 + |E|)$ directly
$O(|V|\log|V|+|E|\log|V|)$ heaps

$O(|V|+|E|)$

6

Single-Source Shortest Path

- Given a graph **G** = (V, E) and a single distinguished vertex **s**, find the shortest weighted path from **s** to every other vertex in **G**.

All-Pairs Shortest Path:

- Find the shortest paths between all pairs of vertices in the graph.
- How?

7

Analysis

- Total running time for Dijkstra's:

  $O(|V|^2 + |E|)$ (linear scan)

  $O(|V| \log |V| + |E| \log |V|)$ (heaps)

  What if we want to find the shortest path from each point to ALL other points?

8

Dynamic Programming

Algorithmic technique that systematically records the answers to sub-problems in a table and re-uses those recorded results (rather than re-computing them).

Simple Example: Calculating the Nth Fibonacci number.

Fib(N) = Fib(N-1) + Fib(N-2)

9

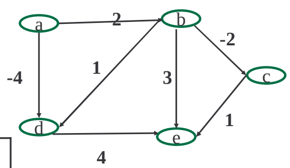
Floyd-Warshall

```
for (int k = 1; k =< V; k++)
 for (int i = 1; i =< V; i++)
  for (int j = 1; j =< V; j++)
   if ( ( M[i][k]+ M[k][j] ) < M[i][j] )
     M[i][j] =  M[i][k]+ M[k][j]
```

Invariant: After the kth iteration, the matrix includes the shortest paths for all pairs of vertices (i,j) containing only vertices 1..k as intermediate vertices

10

Initial state of the matrix:

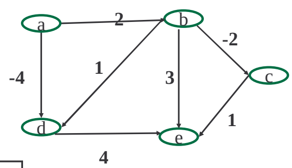


	a	b	c	d	e
a	0	2	-	-4	-
b	-	0	-2	1	3
c	-	-	0	-	1
d	-	-	-	0	4
e	-	-	-	-	0

M[i][j] = min(M[i][j], M[i][k]+ M[k][j])

11

Floyd-Warshall - for All-pairs shortest path



	a	b	c	d	e
a	0	2	0	-4	0
b	-	0	-2	1	-1
c	-	-	0	-	1
d	-	-	-	0	4
e	-	-	-	-	0

Final Matrix Contents

12

Floyd-Warshall

Performance

- Time = $O(|V|^3)$
- Space = $O(|V|^2)$

13