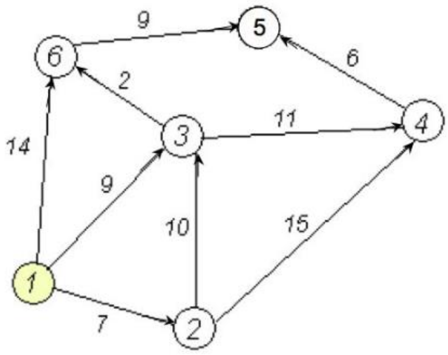
Practice 3.1: Shortest path
Artificial Intelligence

G.Guérard

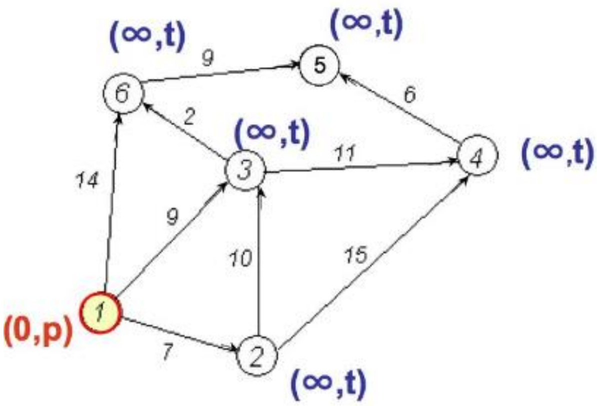
Dijkstra's algorithm

Let us compute the Dijkstra's algorithm step by step on the following graph starting at node 1. Let us remind the dynamic programming function:

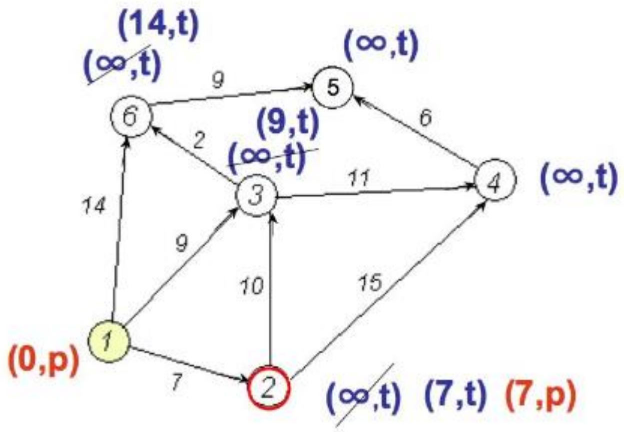
$$d[i] = \min\{d[i], d[selected\ node\ s] + weight(s, i)\}$$



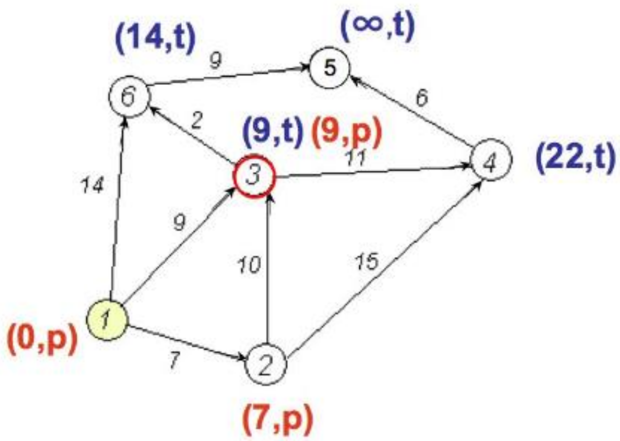
Node 1 is designated as the current node. At each node, we attribute two values (n, m) where n is the minimal distance found and m denote if the vertex is still visited or not (p or t). The state of node 1 is (0,p). Every other node has state (inf,t).



Node 2, 3 and 6 can be reached from the current node 1. Update distance values for these nodes: $d_2 = \min\{inf, 0 + 7\} = 7$, $d_3 = \min\{inf, 0 + 9\} = 9$, $d_6 = \min\{inf, 0 + 14\} = 14$. Among all non-visited vertices, node 2 has the smallest distance value. It is selected for the next iteration.



Now node 2 becomes the current node. Nodes 3 and 4 can be reached from the current node 2. Update distance values for these nodes: $d_3 = \min\{9, 7+10\} = 9$, $d_6 = \min\{inf, 7+15\} = 22$. Node 3 has the smallest distance value. It is selected for the next iteration.



The process can be simplified by a table.

Distance	1	2	3	4	5	6
Initiate	0	inf	inf	inf	inf	inf
1(0)		7	9	inf	inf	14
$2(7_1)$			9	22	inf	14
$3(9_1)$				20	inf	11
$6(11_3)$				20	20	
$4(20_3)$					20	

The first raw describe the initiation, the second raw the first iteration etc., value in red is the selected node. $2(7_1)$ means that the smallest path from 1 to 2 is equal to 7, and its predecessor is node 1. From this notation, we can compute the tree of smallest path. You can also add the predecessor for each value inside the table or see where is the first time (the line) you found the minimal value for each column – its predecessor is the selected node (first column of the line). We can deduce the following tree:

1---2
|
3---4
|
6---5.

Exercise

Test your skill on the following graphs: