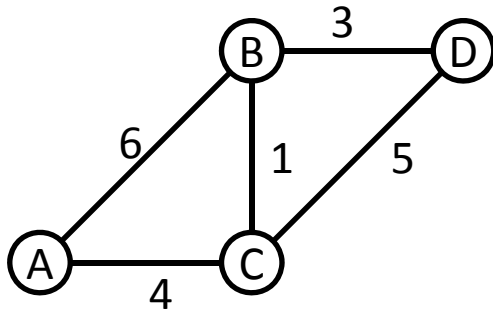


Dijkstra algorithm in time $O(N^2)$, small example



	A	B	C	D
A	0	6	4	0
B	6	0	1	3
C	4	1	0	5
D	0	3	5	0

	dist/pred
A	0/null
B	inf/null
C	inf/null
D	inf/null

Init: for each node X : $\text{dist}(X) = \text{infinity}$; $\text{pred}(X) = \text{null}$; $\text{dist}(\text{start}) = 0$

loop: for i in $[1..N-1]$

1. Find the open node X with smallest distance from start.

Apply linear (slowest!) search, **// $O(N)$**

2. Recalculate the distances of all non-closed neighbours of X .

for YY in neighbours X : **// $O(\text{dg}(X)) < O(N)$**

if $\text{dist}(YY) > \text{dist}(X) + \text{edgeWeight}(YY, X)$

$\text{dist}(YY) = \text{dist}(X) + \text{edgeWeight}(YY, X)$

$\text{pred}(YY) = X$

