Floyd's Algorithm
(All Sources Shortest paths)

Fall 2019

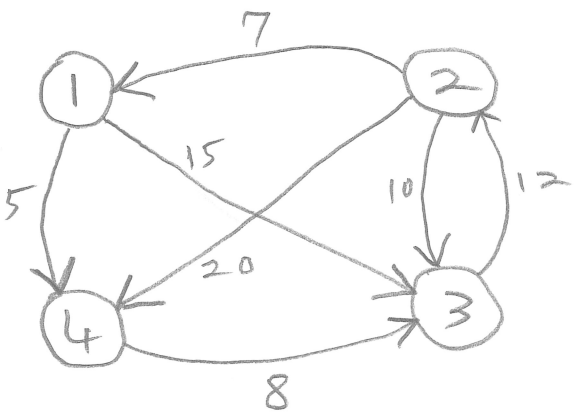
initialize "distance" array

$$\text{distance}[i][j] = \begin{cases} 0 & \text{if } i = j \\ \text{cost}[i][j] & \text{if there is an edge from } i \text{ to } j \\ \infty & \text{otherwise} \end{cases}$$

```
for pivot = 1 to n
   for source = 1 to n
      for destination = 1 to n
         if distance[s][d] > distance[s][p] + distance[p][d] then
            distance[s][d] = distance[s][p] + distance[p][d]
         end if
      end for
   end for
end for
```

Example:

	1	2	3	4
1	0	∞	15	5
2	7	0	10	20
3	∞	12	0	∞
4	∞	∞	8	0



```
Pivot = vertex 1
   for source = 1 to 4
      for destination = 1 to 4
```

```
distance[2][4] > distance[2][1] + distance[1][4]
```

	1	2	3	4
1	0	∞	15	5
2	7	0	10	12
3	∞	12	0	∞
4	∞	∞	8	0

```
Pivot = vertex 2
   for source = 1 to 4
      for destination = 1 to 4
```

```
distance[3][1] > distance[3][2] + distance[2][1]

distance[3][4] > distance[3][2] + distance[2][4]
```

	1	2	3	4
1	0	∞	15	5
2	7	0	10	12
3	19	12	0	24
4	∞	∞	8	0

```
Pivot = vertex 3
   for source = 1 to 4
      for destination = 1 to 4
```

```
distance[1][2] > distance[1][3] + distance[3][2]

distance[4][1] > distance[4][3] + distance[3][1]

distance[4][2] > distance[4][3] + distance[3][2]
```

	1	2	3	4
1	0	27	15	5
2	7	0	10	12
3	19	12	0	24
4	27	20	8	0

```
Pivot = vertex 4
   for source = 1 to 4
      for destination = 1 to 4
```

```
distance[1][2] > distance[1][4] + distance[4][2]

distance[1][3] > distance[1][4] + distance[4][3]
```

	1	2	3	4
1	0	25	13	5
2	7	0	10	12
3	19	12	0	24
4	27	20	8	0