Graph

$G = \langle V, E \rangle$

$E \subseteq V \times V$

an example

$V = \{A, B, C, D, E, F\}$

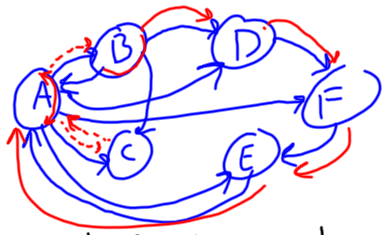
$E = \{ \langle A,B \rangle, \langle A,C \rangle, \langle B,A \rangle, \ldots \}$

Representing a graph:

- O.O.: class Vertex, includes a field for List<Edge>.
  class Edge
  include a field for start-vertex, end-vertex.
- book: Vertices 1 .. n.
  Edges: a matrix to of 0,1:

"Adjacency matrix"

Question: Can I get from a to z?



from F, to D:

DFS(E, D)?

DFS(A, D)?

DFS(C, D)? X
false

DFS(B, D)?

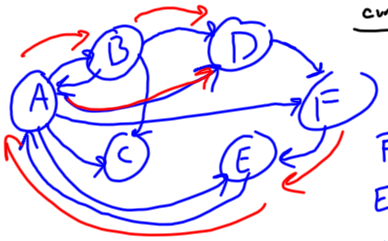
DFS(A, D)?

Depth-first search:

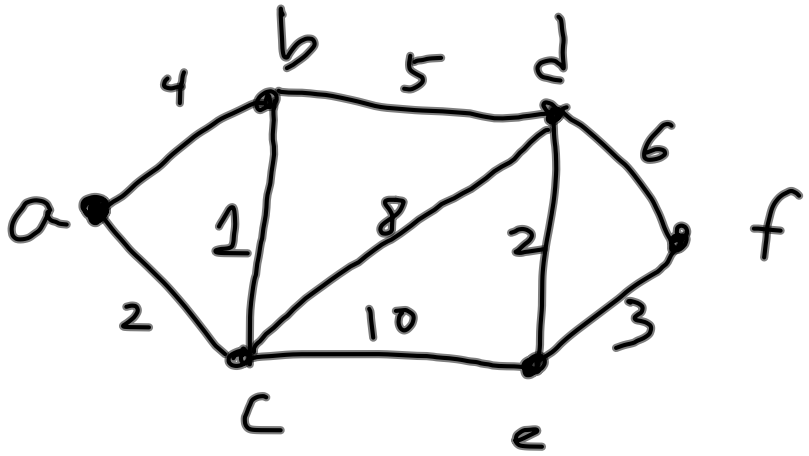
To search from a:

initialize → boolean[] seen to false.

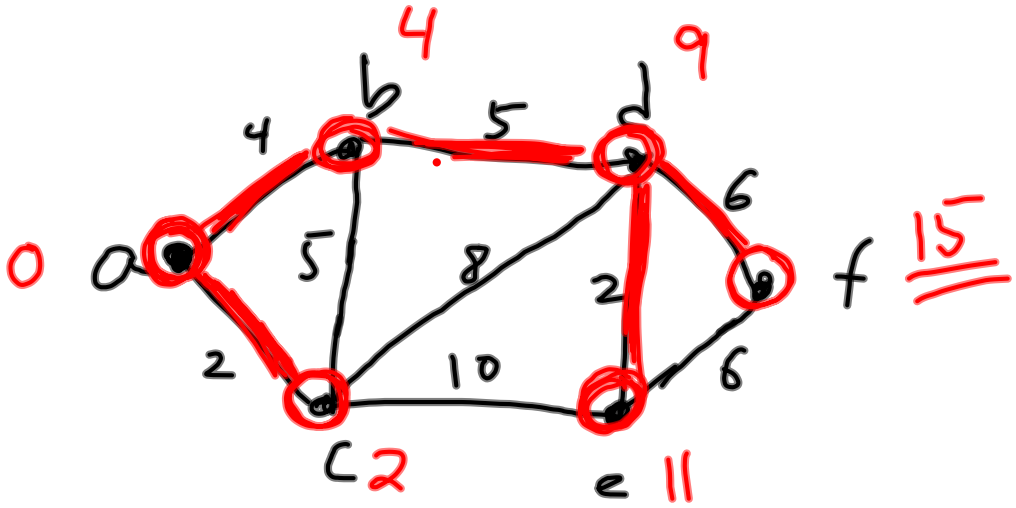
```
boolean DFS(Vertex from, Vertex to) {
  if (from == to) {
    return true;
  } if seen[from] return false
  seen[from] = true
  else {
    for each neighbor curr of from,
      if DFS(curr, to)
        return true
  }
  return false.
}
```

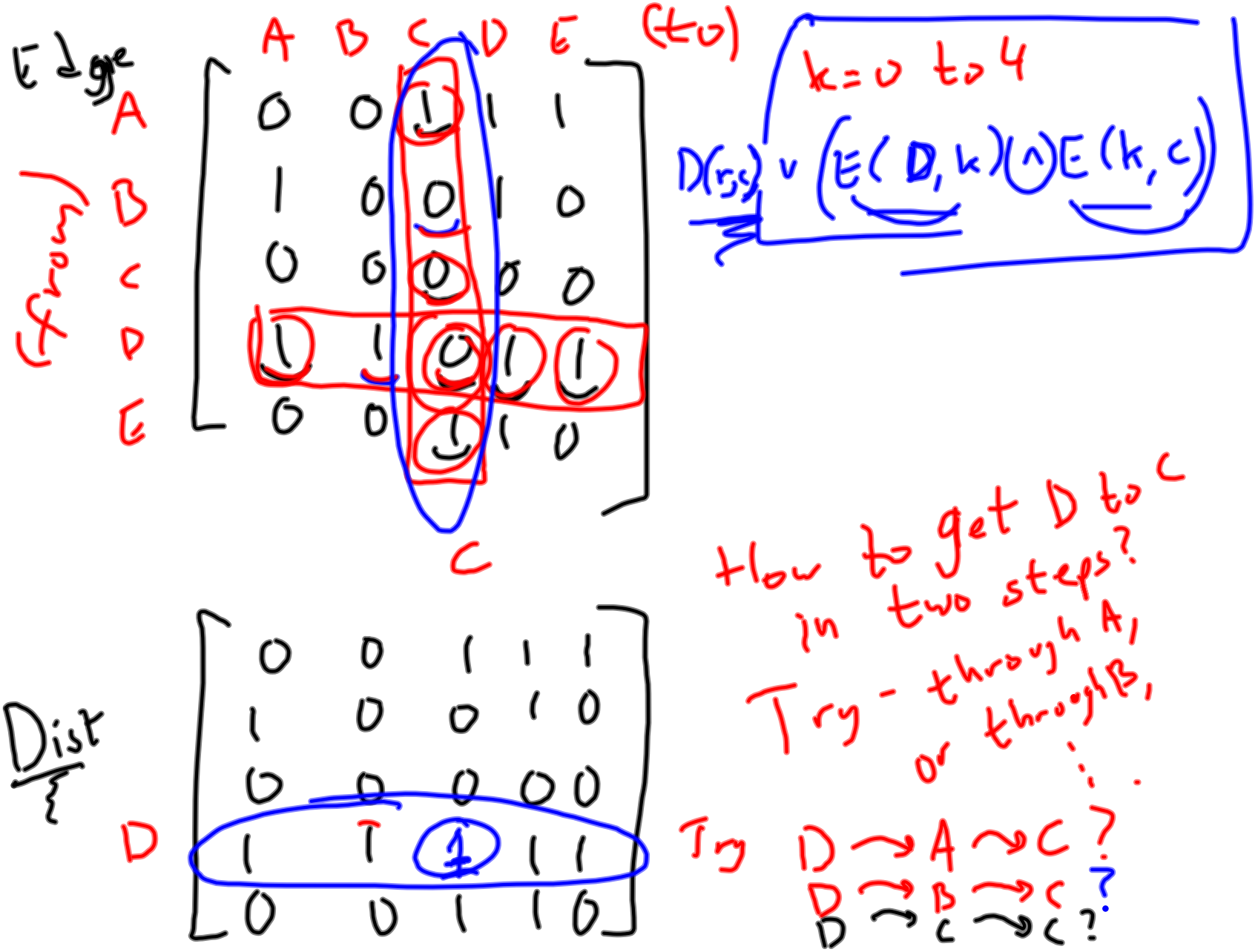


curr	seen	pending
	{ }	{D}
D	{D}	{F}
F	{D,F}	{E}
E	{D,E,F}	{A}
A	{A,D,E,F}	{C, B, E, D, F}
C	{A, C, D, E, F}	{B, E, D, F}
B	{A, B, C, D, E, F}	{E, D, F}
E		{D, F}
D		



Pijkstra's Algorithm




Edge
A B C D E
(to)
(from)
A
B
C
D
E
k=0 to 4
D(r,s) v (E(D,k) ^ E(k,c))
Dist
D
how to get D to C
in two steps?
Try - through A,
or through B,
Try D → A → C ?
D → B → C ?
D → C → C ?

Graphs represent binary relations on the domain (e.g. "likes", "is taller than", "same zodiac sign as", "has shaken hands with")

- reflexive: $\forall x.\ E(x,x)$
- Symmetric: $\forall x.\forall y.\ (E(x,y) \rightarrow E(y,x))$
- antisymmetric: $\forall x.\forall y.\ (E(x,y) \rightarrow \neg E(y,x))$
- transitive: $\forall x.\forall y.\forall z.\ (E(x,y) \wedge E(y,z) \rightarrow E(x,z))$

If a relation is symm., refl., and transitive, then it's an "equivalence relation"

In java, when you override a.equals(b), you want to make sure your code really does yield an equivalence relation!