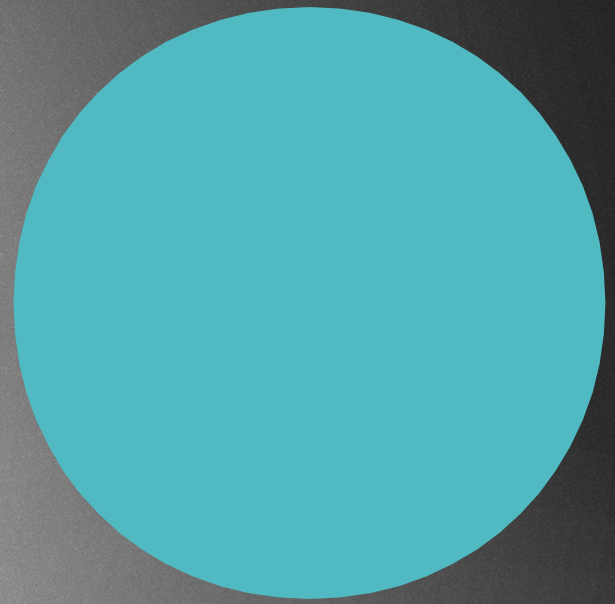
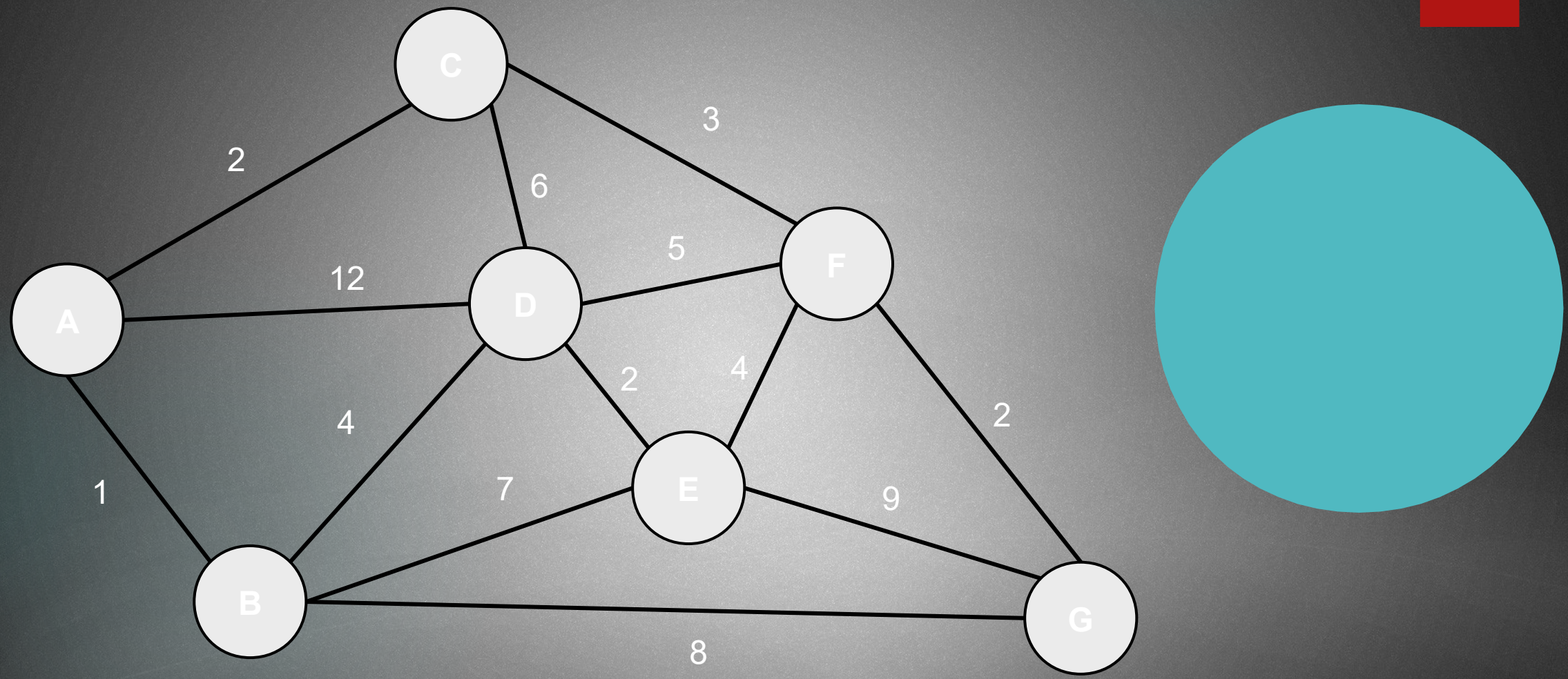
SPANNING TREES

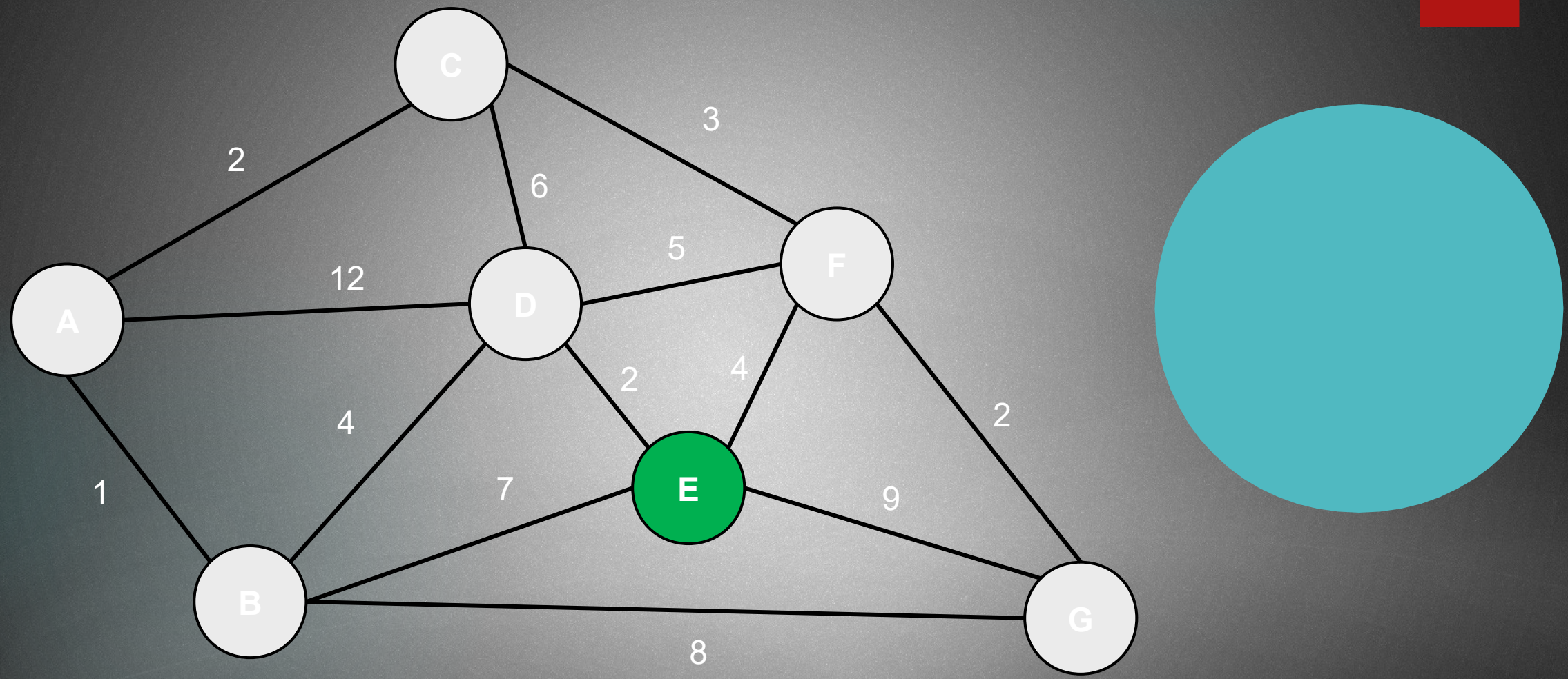


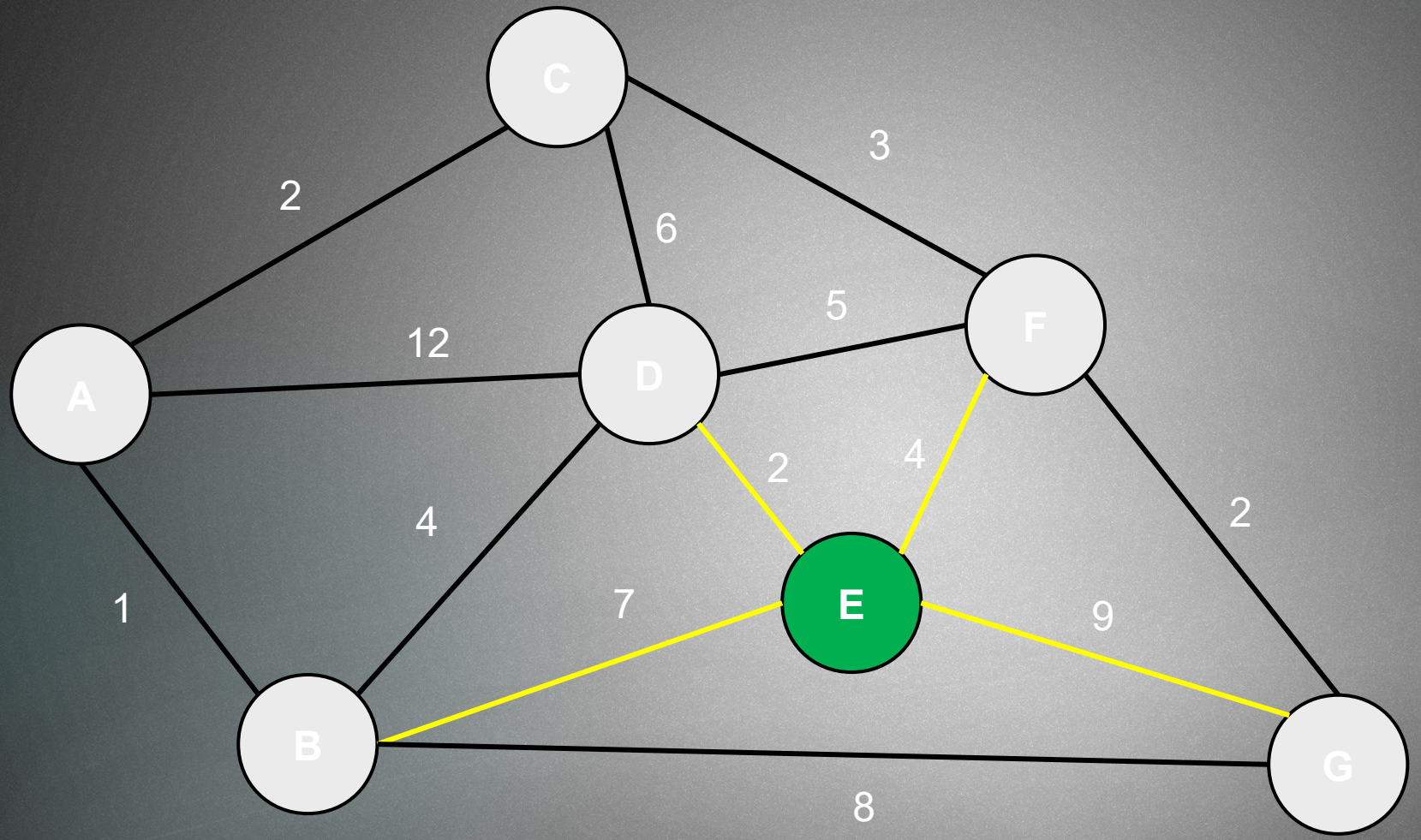
Eager prim’S ALGORITHM

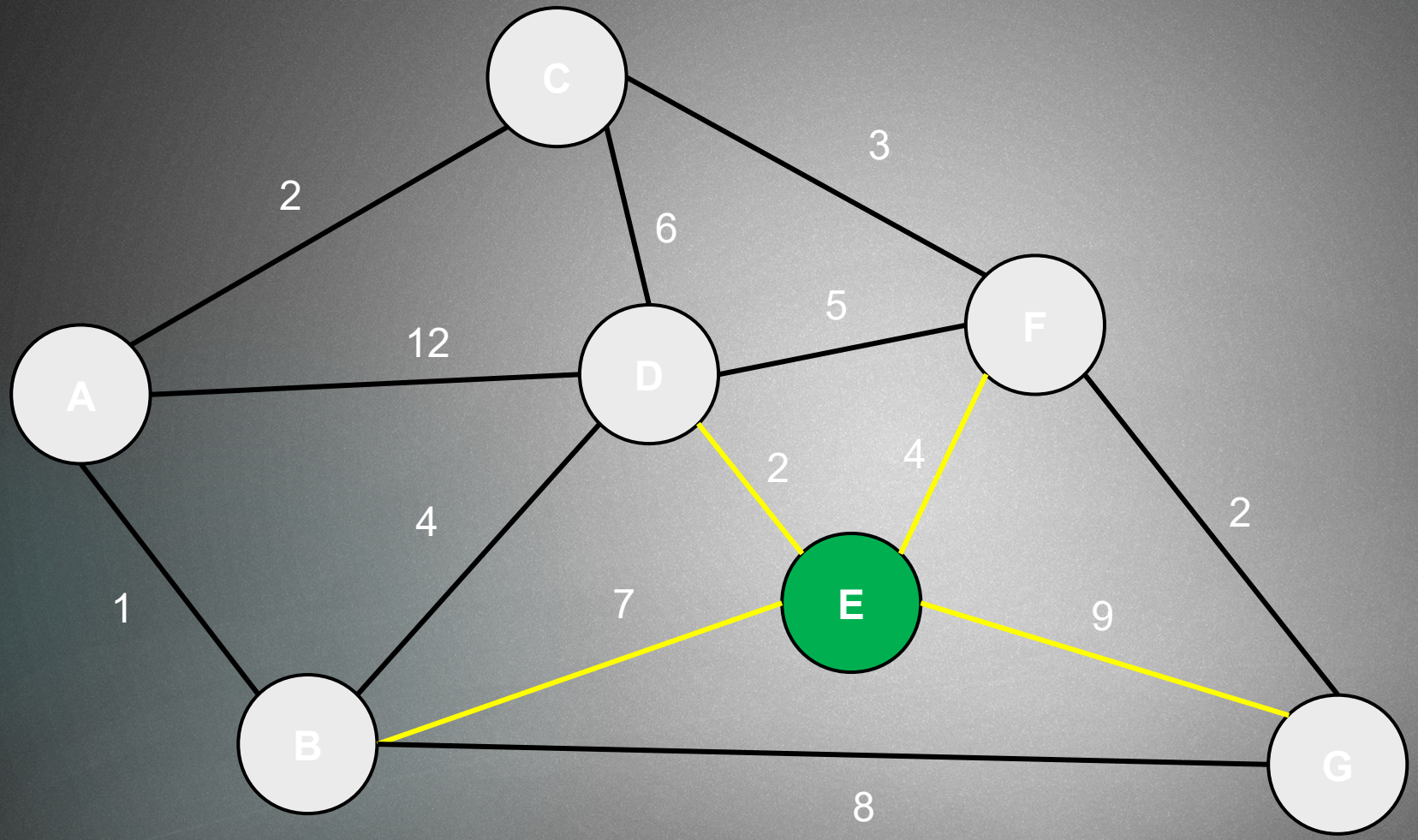
Eager version

- The aim is the same: we want to construct a minimum spanning tree
- The lazy version → we use a priority queue (heap) in order the get the minimum edge weights + we insert all the edges to the heap without modifying the content !!!
- Eager version → we update the content of the heap if necessary
- On every iteration we check whether is there already a shorter path to the spanninng tree


C
A
D
F
B
E
G
2
3
6
12
5
4
2
4
2
1
7
9
8

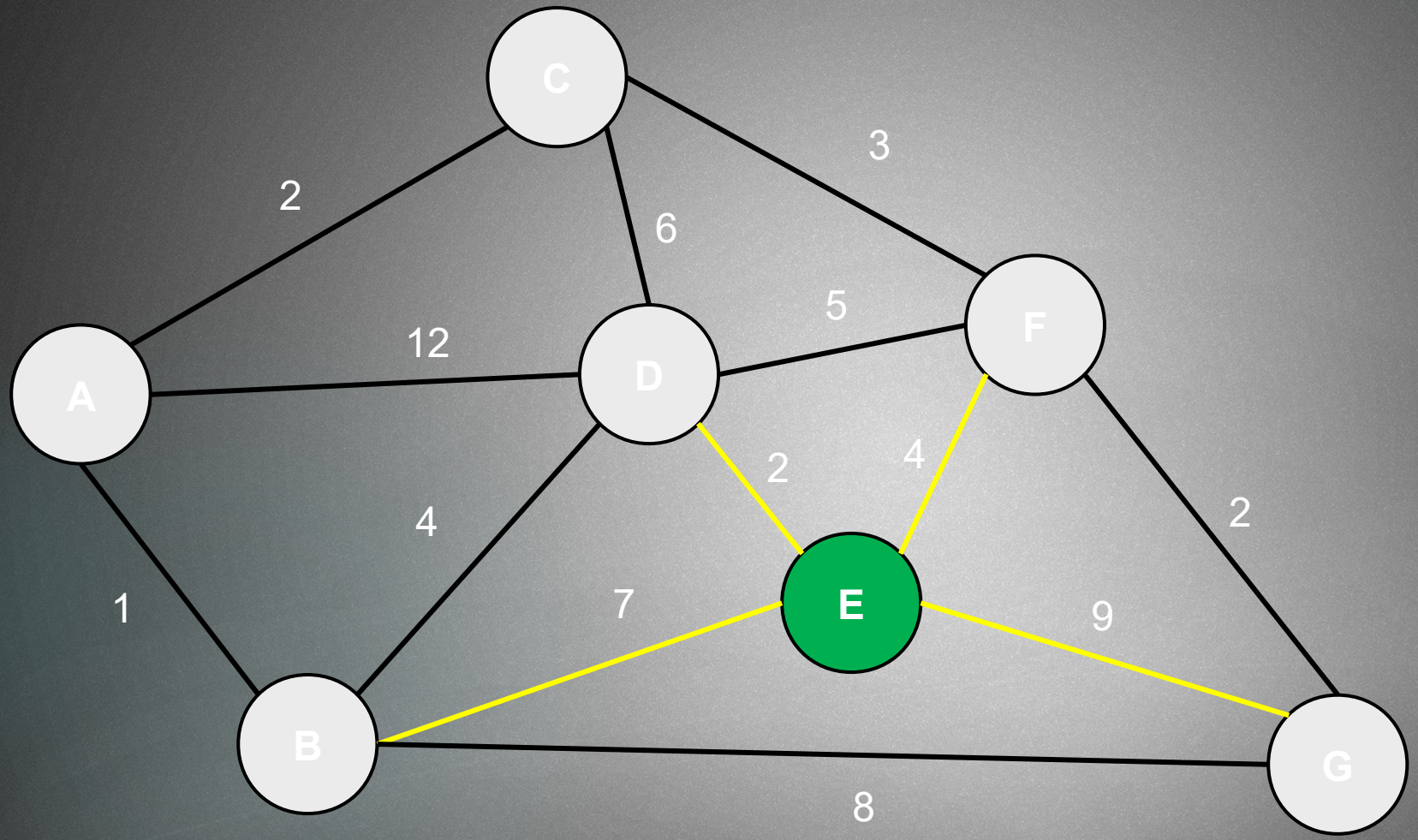

C
A
D
F
E
B
G
2
3
6
12
5
4
2
4
2
1
7
9
8


C
A
D
F
B
E
G
2
3
6
12
5
4
2
4
2
1
7
9
8



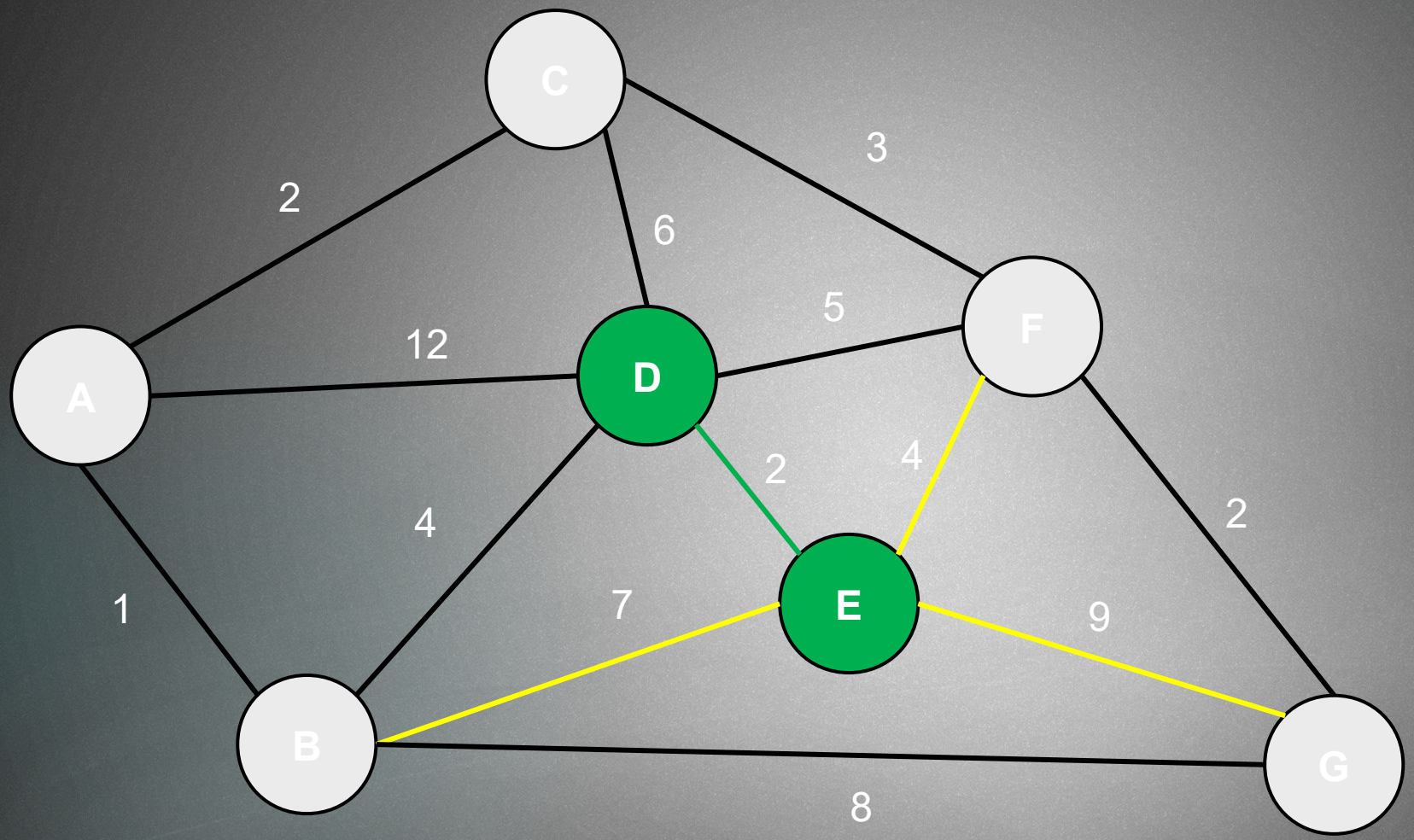
Heap

E-D 2
E-F 4
E-G 9
E-B 7



Heap

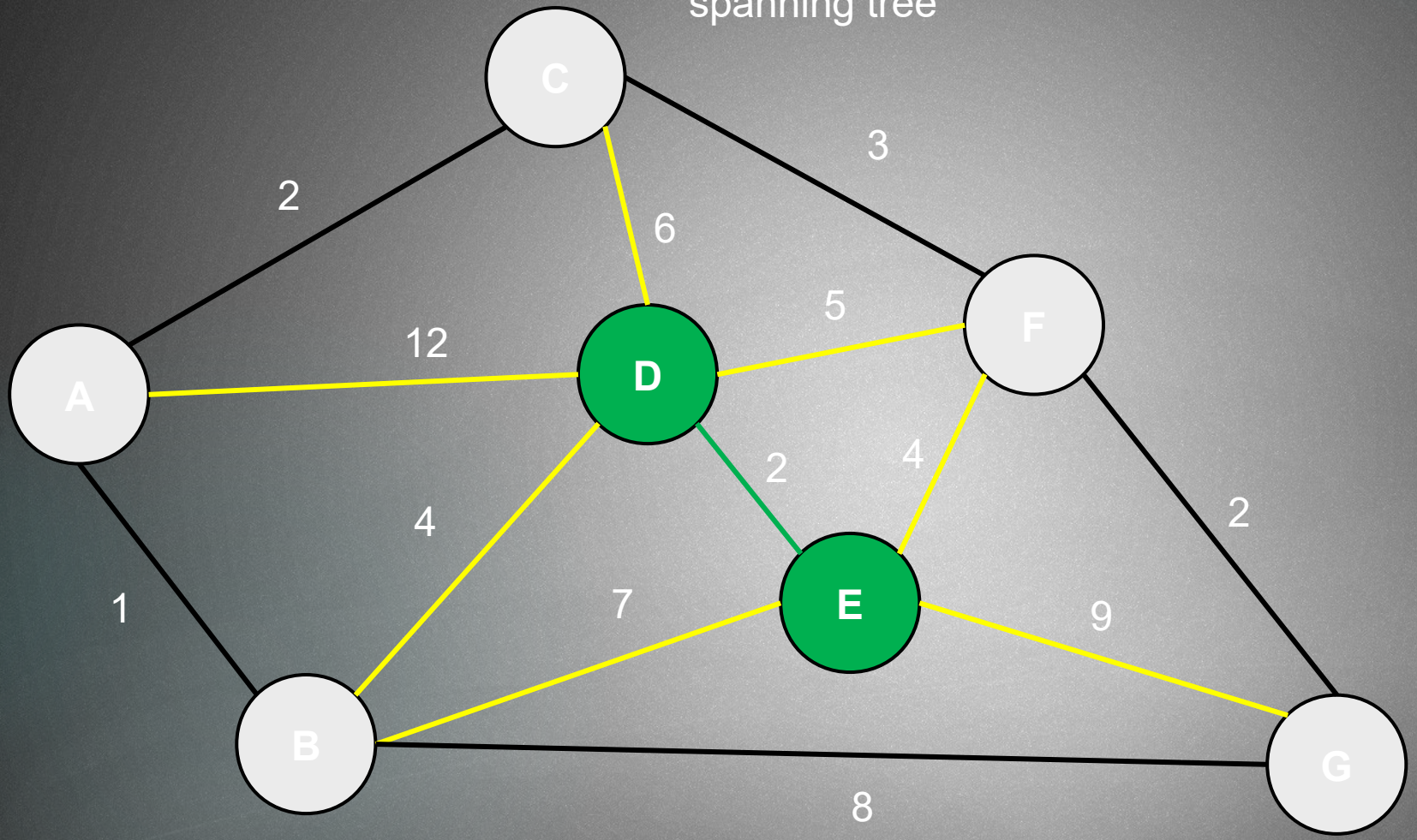
E-D 2
E-F 4
E-G 9
E-B 7



Heap

E-F 4
E-G 9
E-B 7

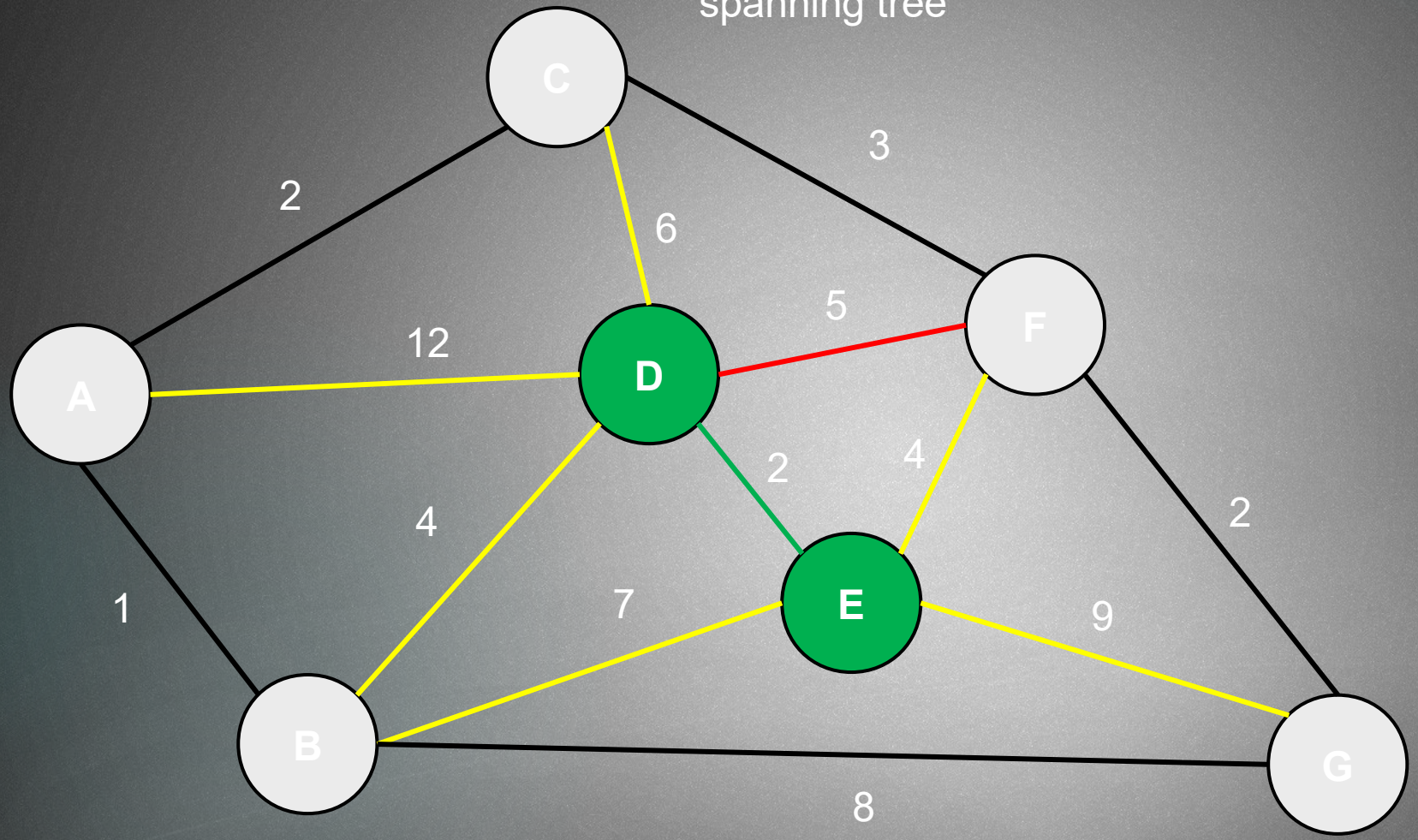
What about the D-F connection? We know that there is a cheaper path to the F node → directly from vertex E → so we exclude D-F as an option in the spanning tree



Heap

E-F 4
E-G 9
E-B 7

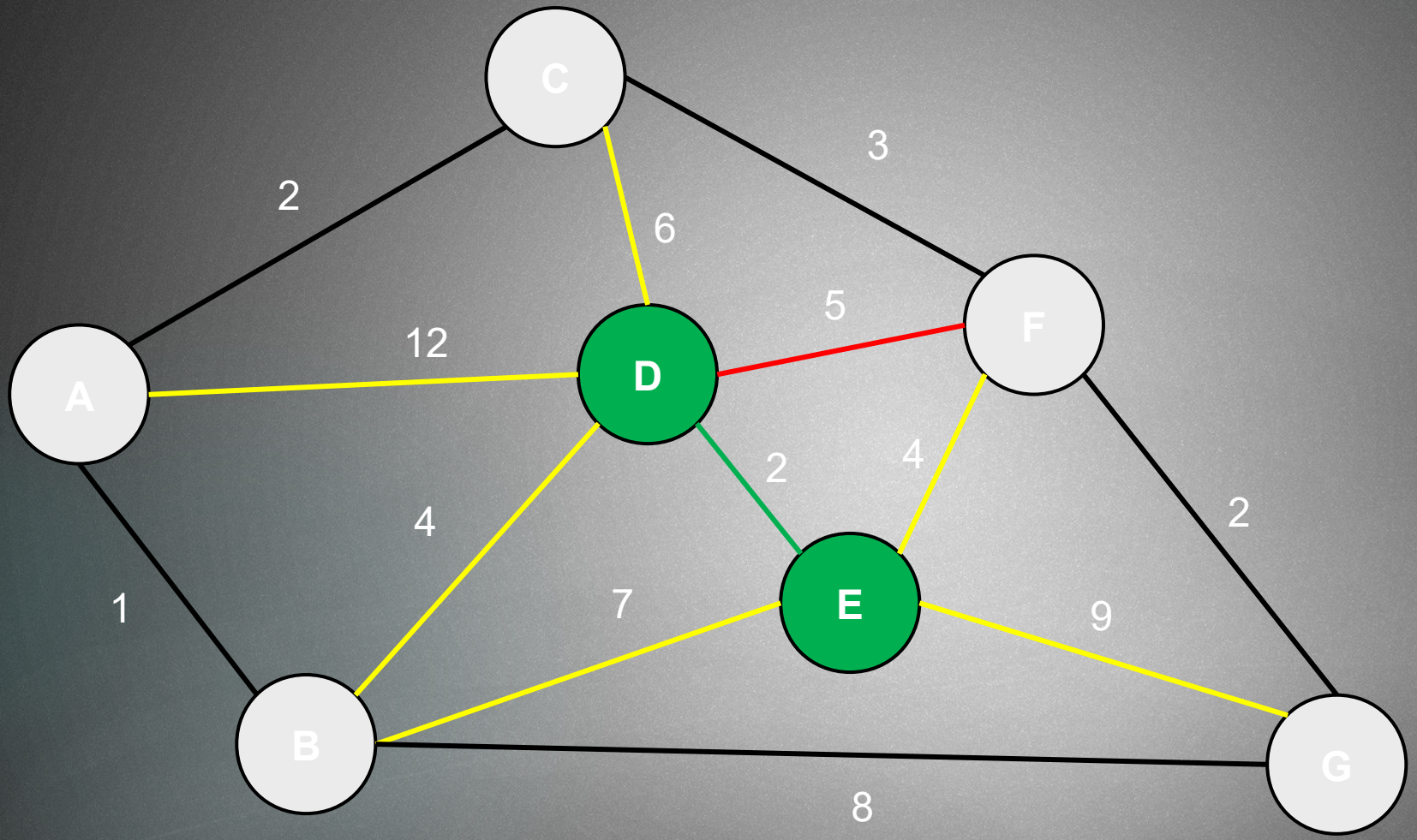
What about the D-F connection? We know that there is a cheaper path to the F node → fdrectly from vertex E → so we exclude D-F as an option in the spanning tree



Heap

E-F 4
E-G 9
E-B 7

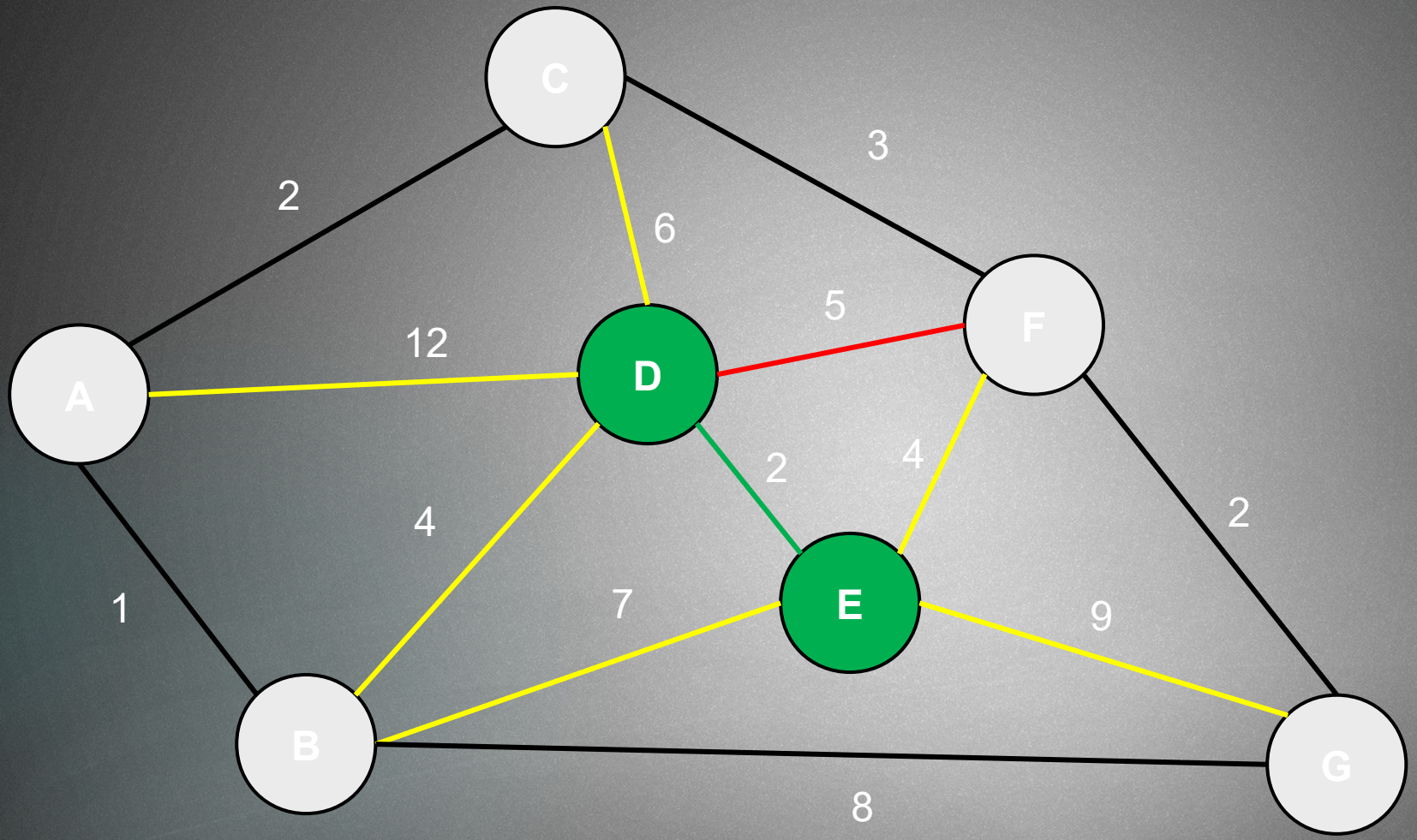
What about the D-C connection? It is the first time we consider vertex C so we add the D-C edge to the heap



Heap

E-F 4
E-G 9
E-B 7

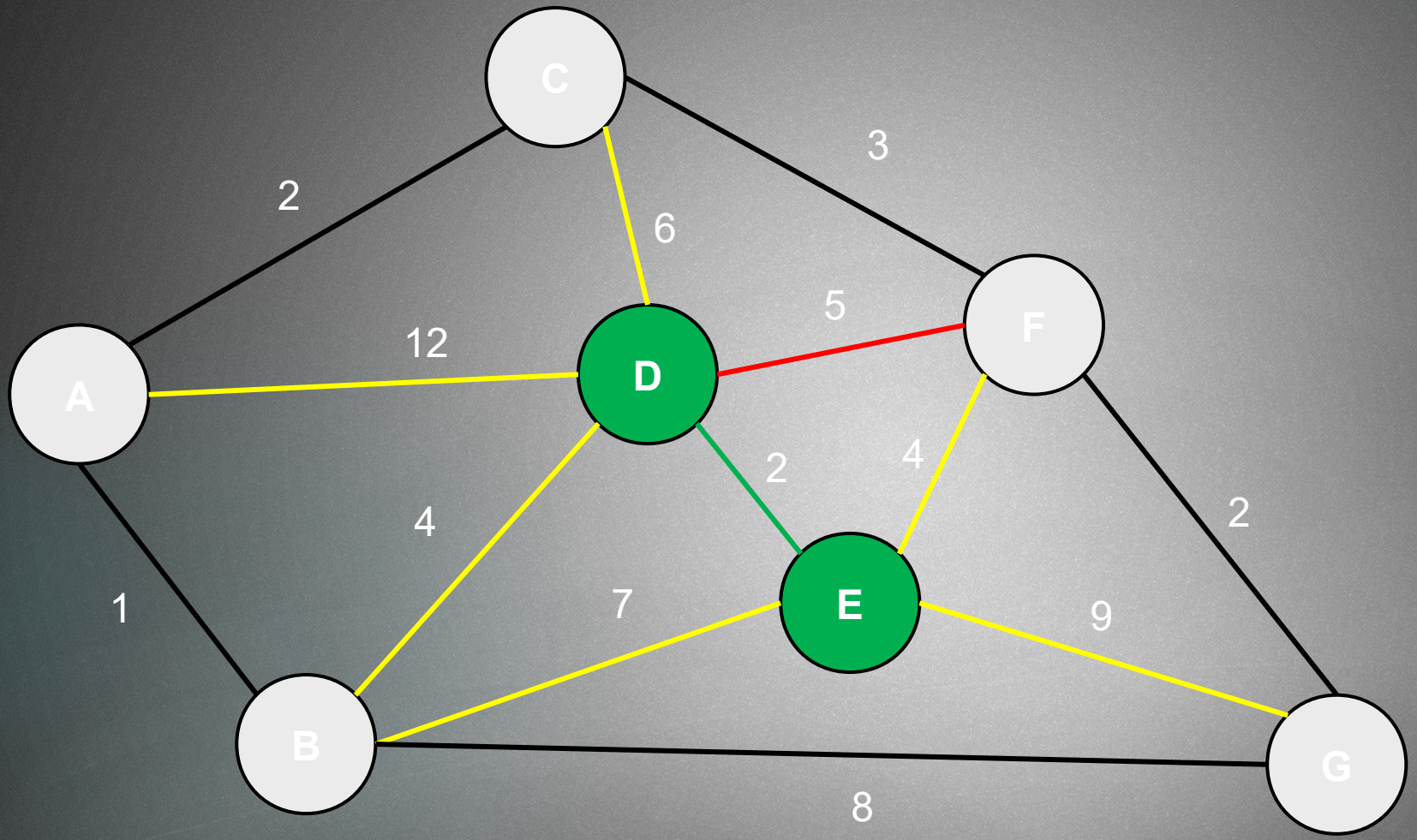
What about the D-C connection? It is the first time we consider vertex C so we add the D-C edge to the heap



Heap

E-F 4
E-G 9
E-B 7
D-C 6

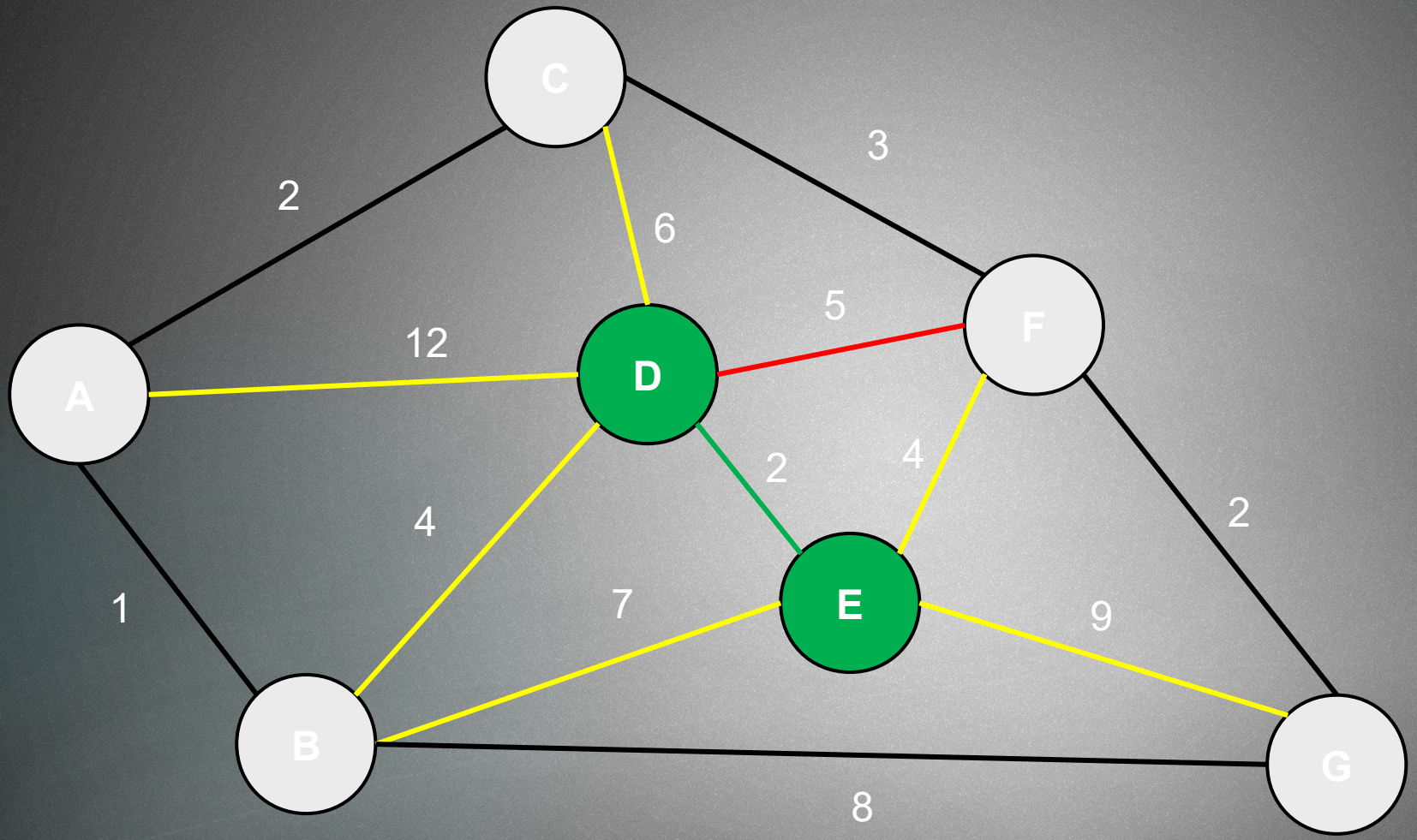
What about the D-A connection? It is the first time we consider vertex A so we add the D-A edge to the heap



Heap

E-F 4
E-G 9
E-B 7
D-C 6

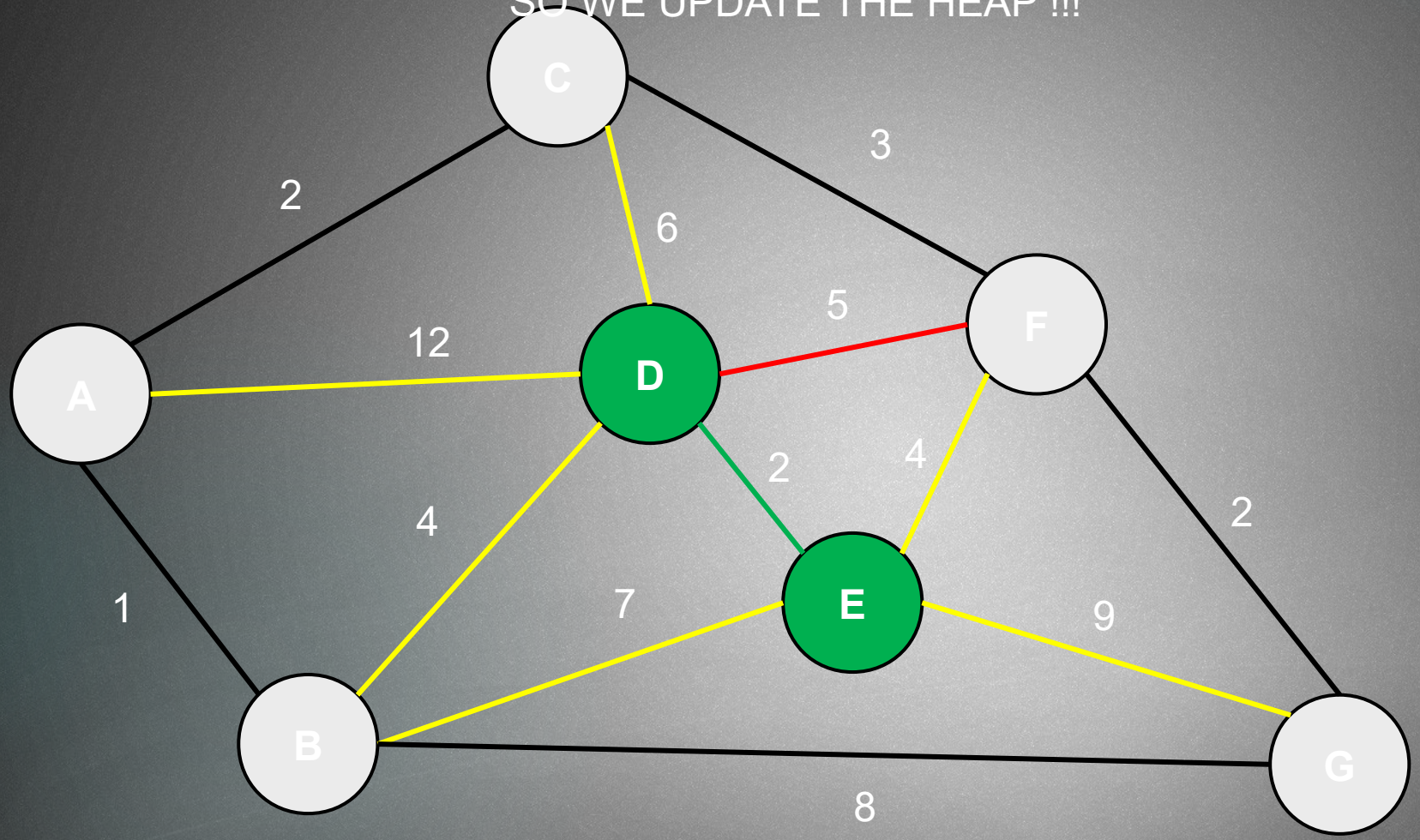
What about the D-A connection? It is the first time we consider vertex A so we add the D-A edge to the heap



Heap

E-F 4
E-G 9
E-B 7
D-C 6
D-A 12

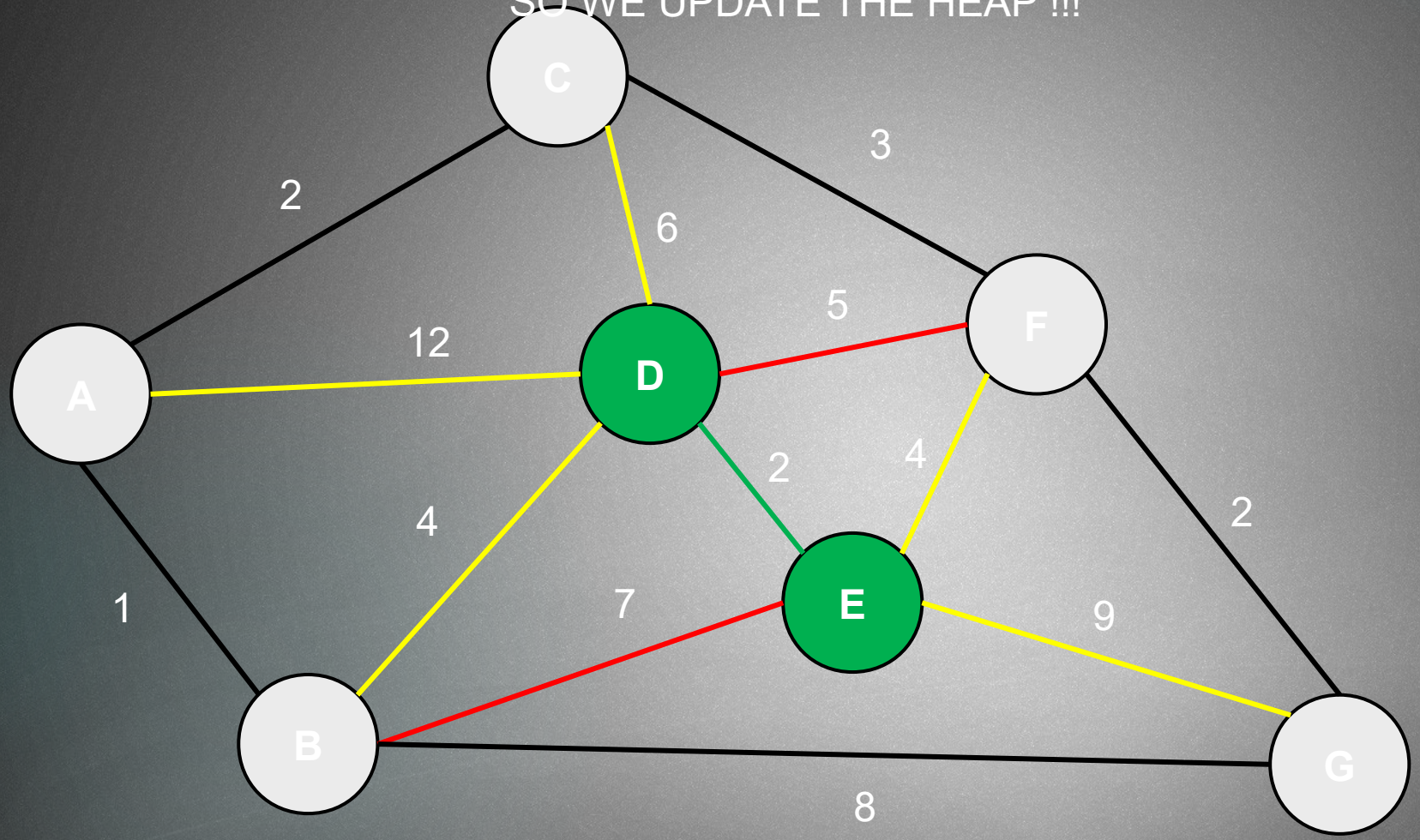
What about the D-B connection? We have considered a path to vertex B so far from vertex E. The cost from vertex E is 7, the cost from vertex D is 4
SO WE UPDATE THE HEAP !!!



Heap

E-F 4
E-G 9
E-B 7
D-C 6
D-A 12

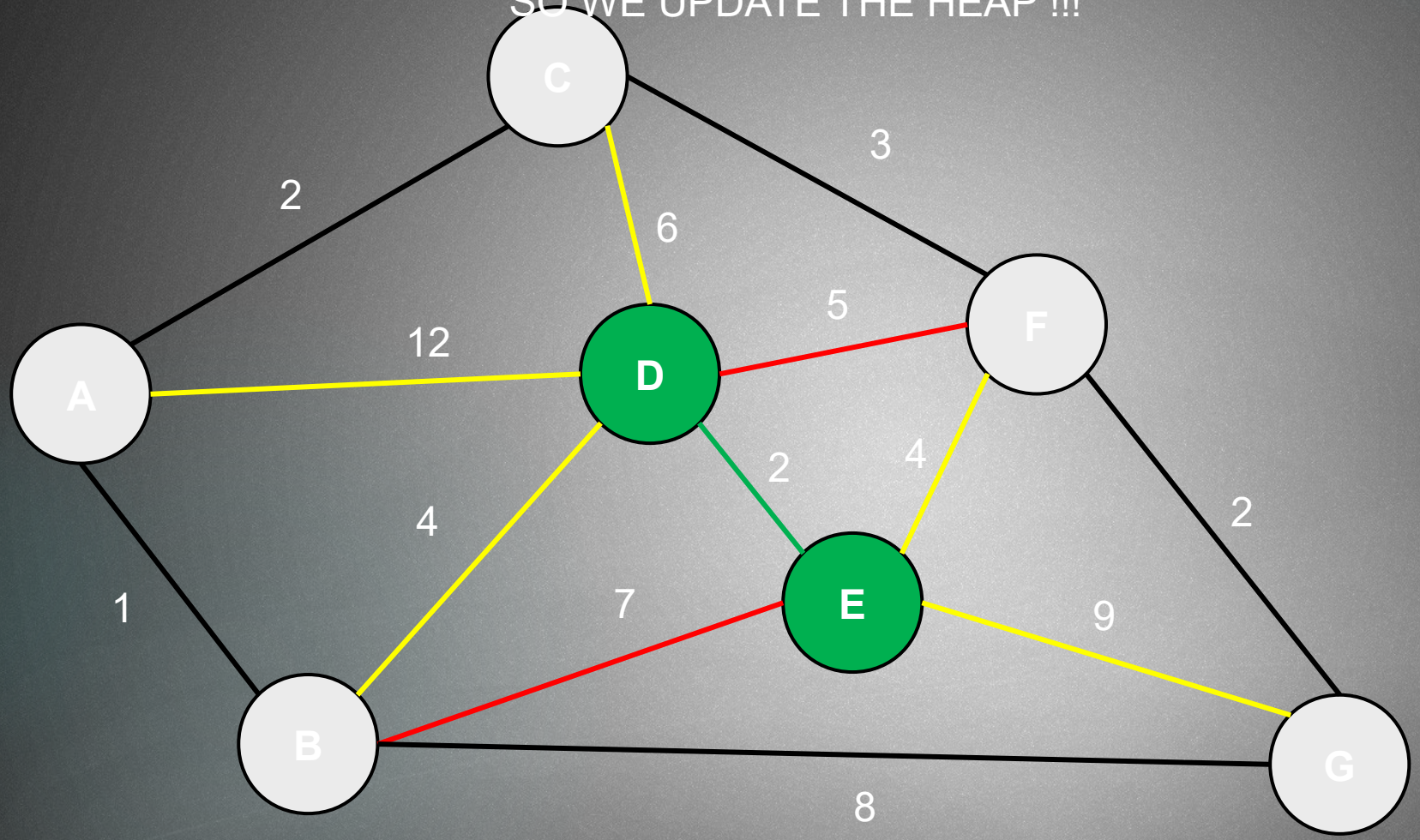
What about the D-B connection? We have considered a path to vertex B so far from vertex E. The cost from vertex E is 7, the cost from vertex D is 4
SO WE UPDATE THE HEAP !!!



Heap

E-F 4
E-G 9
E-B 7
D-C 6
D-A 12

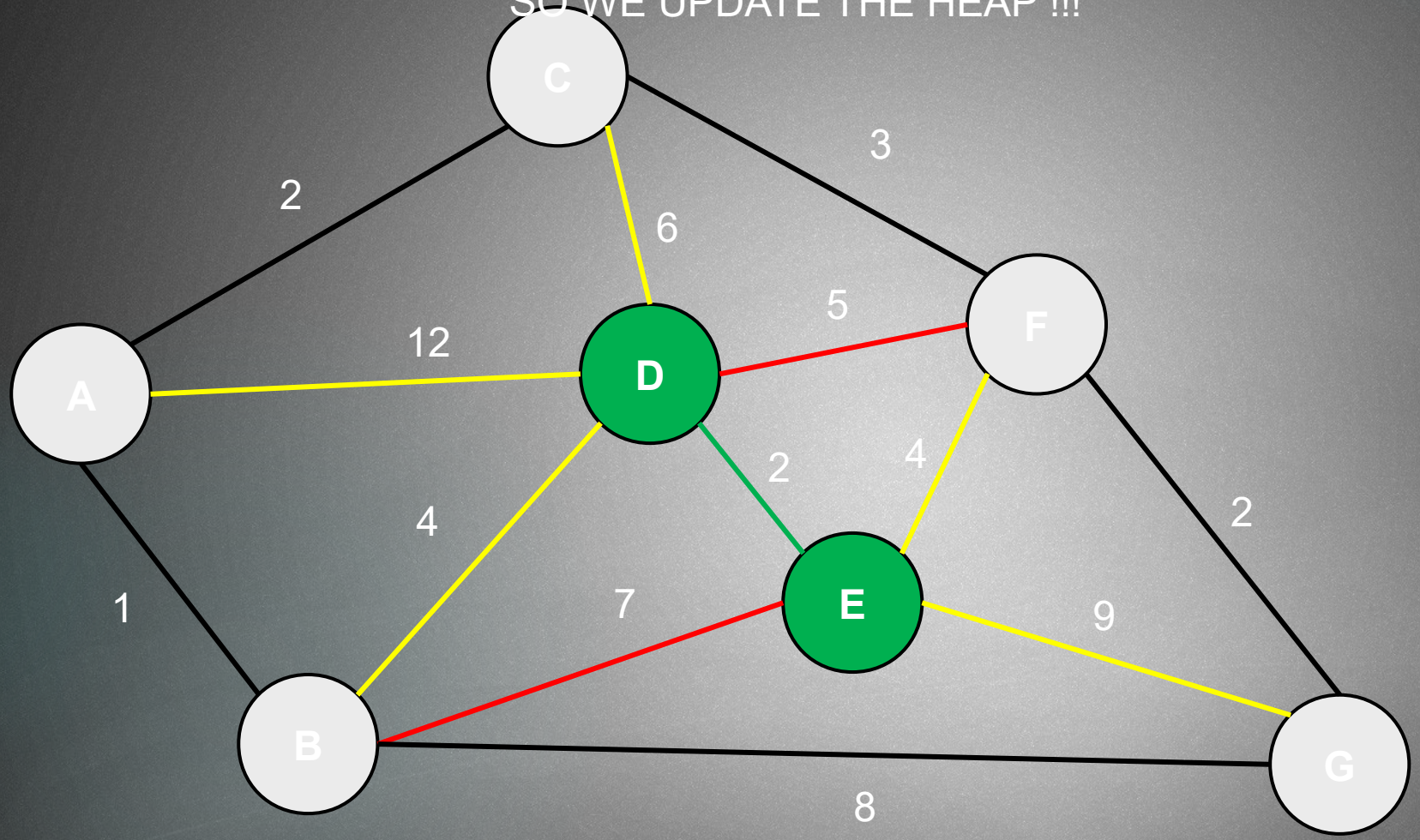
What about the D-B connection? We have considered a path to vertex B so far from vertex E. The cost from vertex E is 7, the cost from vertex D is 4
SO WE UPDATE THE HEAP !!!



Heap

E-F 4
E-G 9
D-C 6
D-A 12

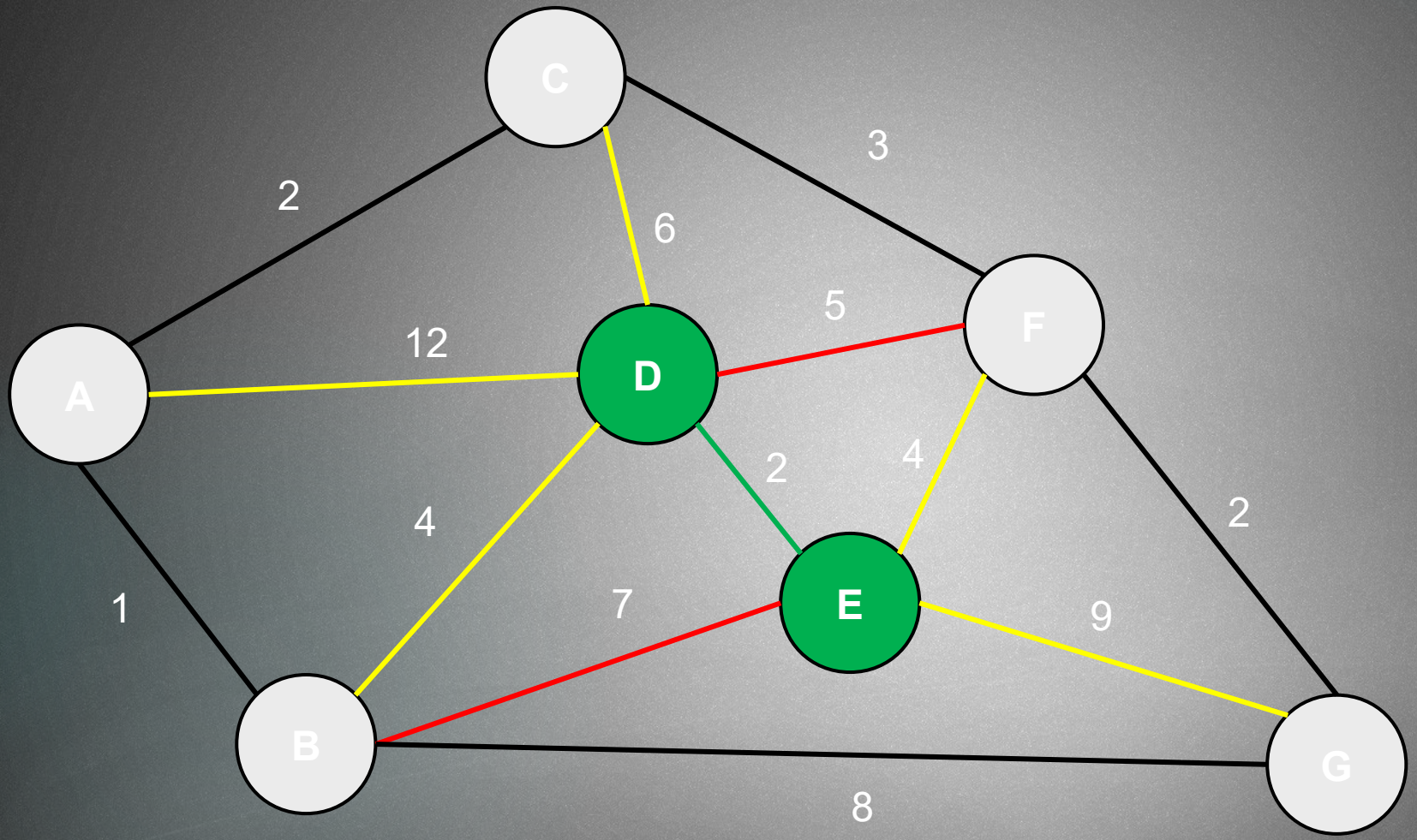
What about the D-B connection? We have considered a path to vertex B so far from vertex E. The cost from vertex E is 7, the cost from vertex D is 4
SO WE UPDATE THE HEAP !!!



Heap

E-F 4
E-G 9
D-C 6
D-A 12
D-B 4

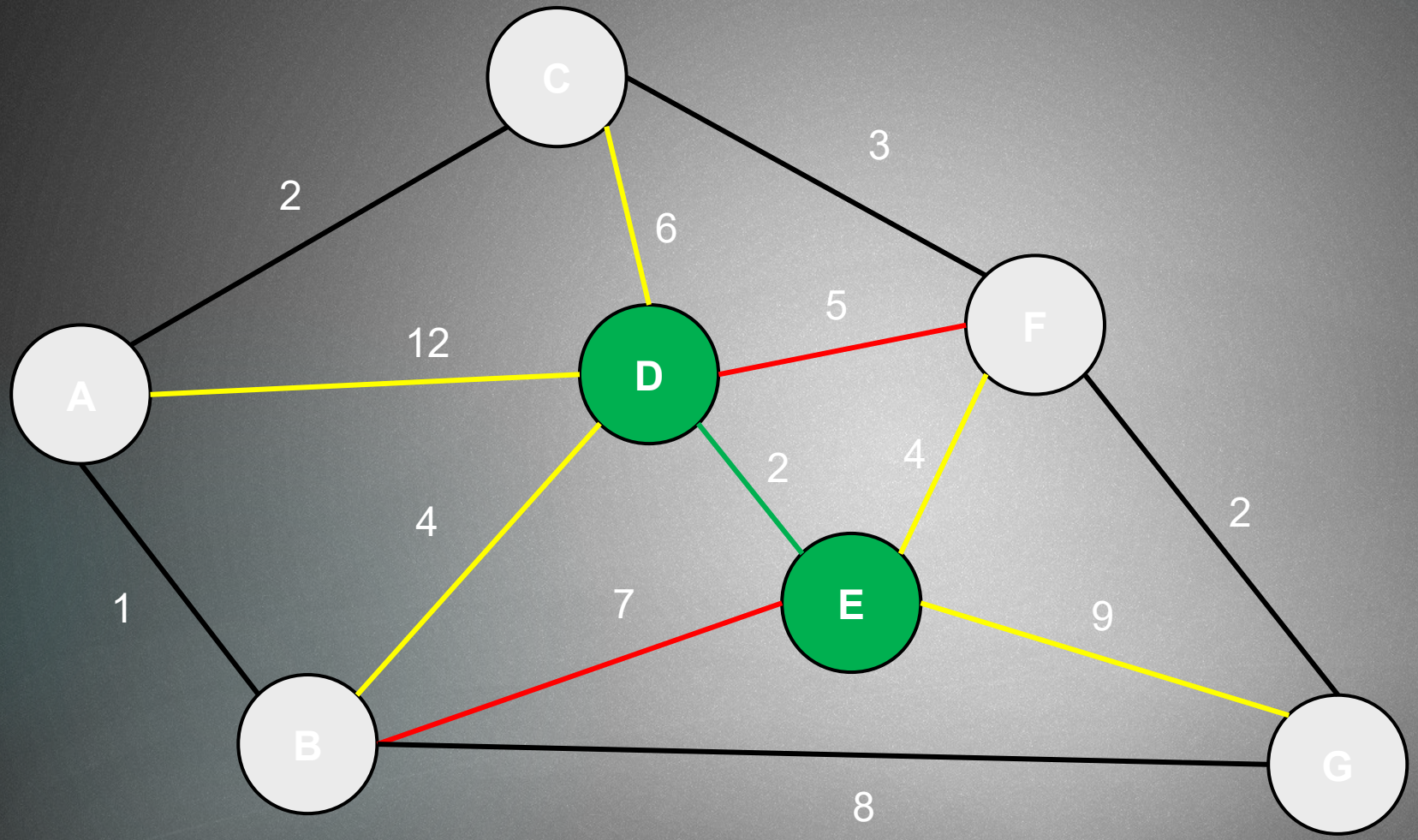
We iterate as far as we have not included all the vertexes in the spanning tree
So we iterate until all the vertexes are green !!!



Heap

E-F 4
E-G 9
D-C 6
D-A 12
D-B 4

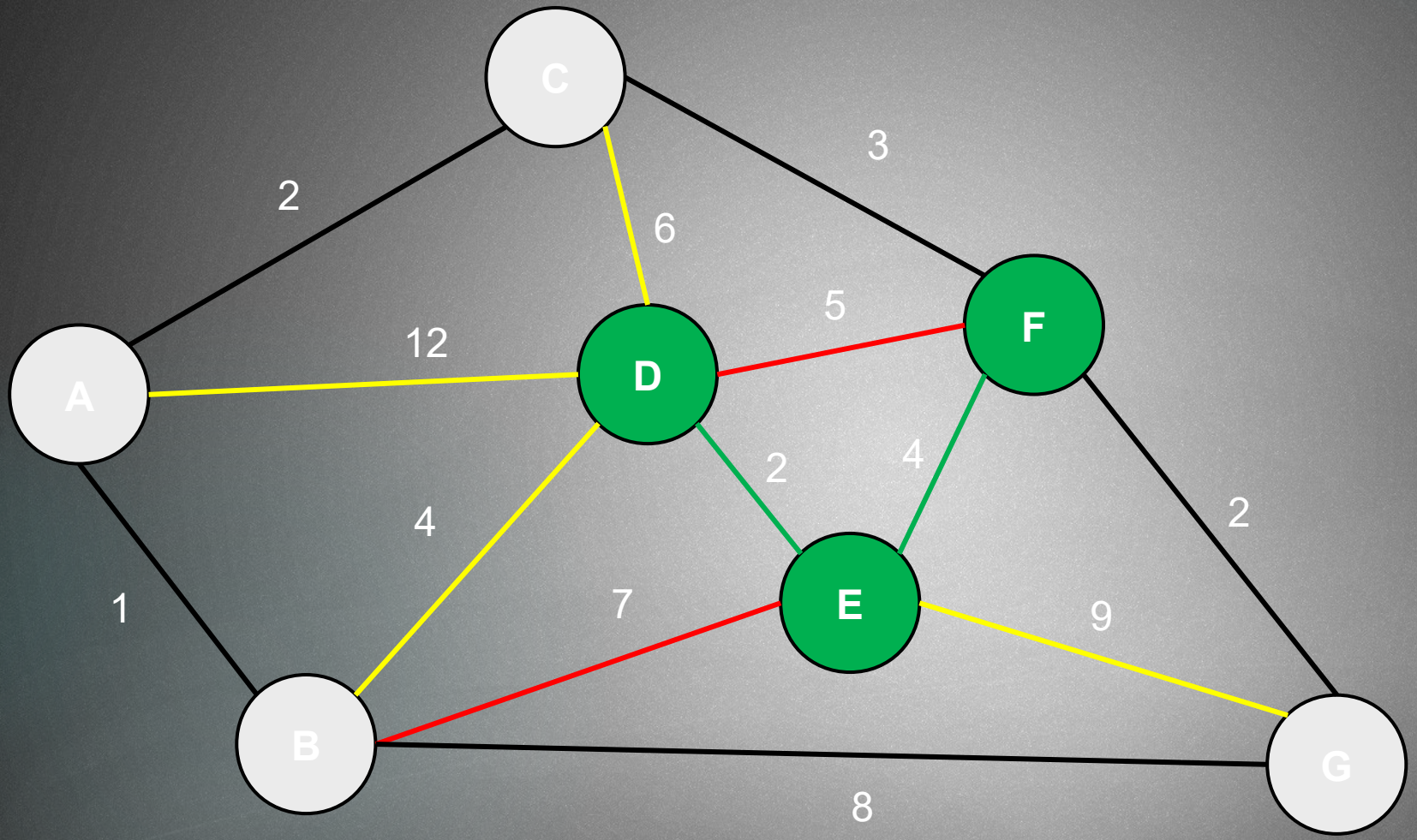
We iterate as far as we have not included all the vertexes in the spanning tree
So we iterate until all the vertexes are green !!!



Heap

E-F 4
E-G 9
D-C 6
D-A 12
D-B 4

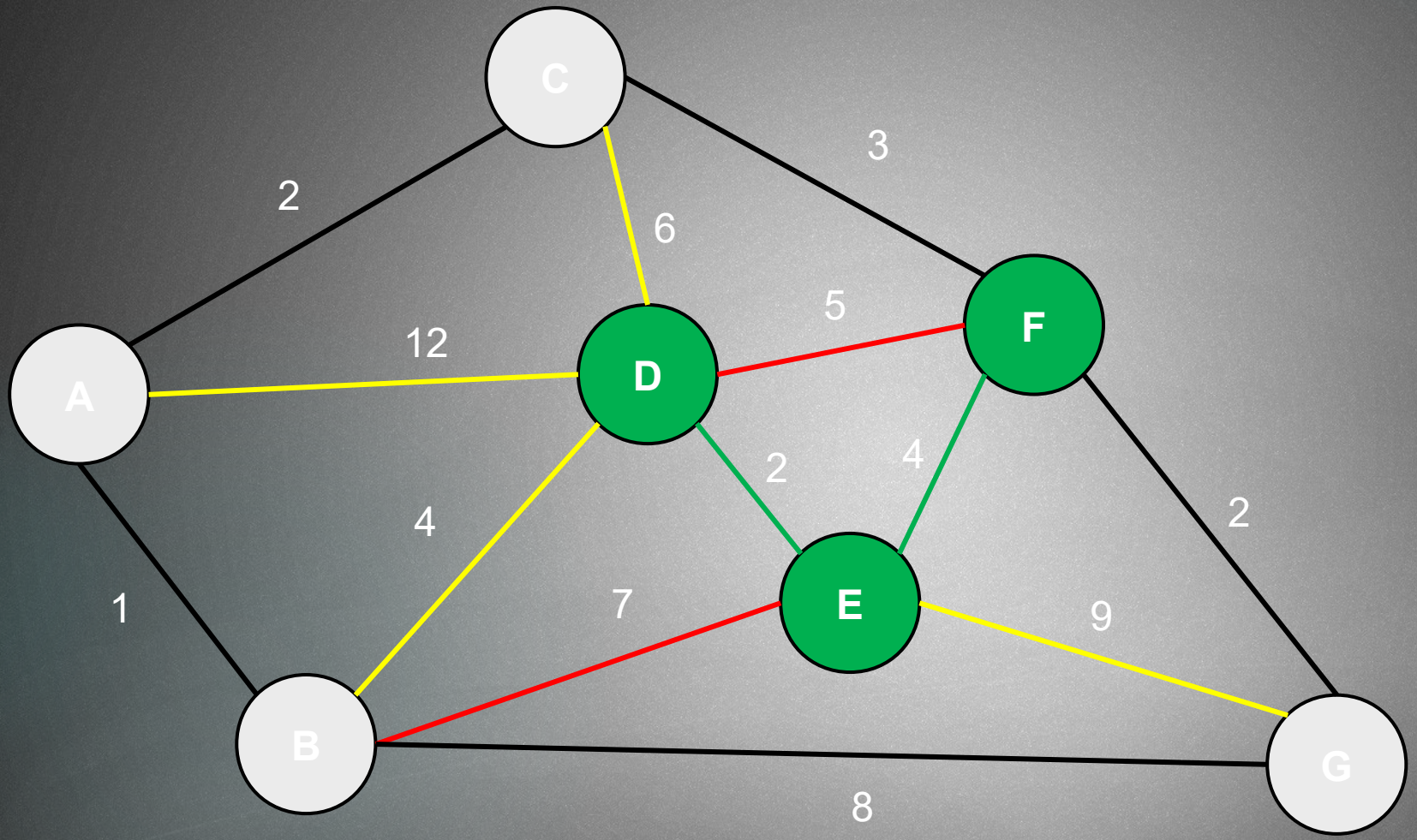
We iterate as far as we have not included all the vertexes in the spanning tree
So we iterate until all the vertexes are green !!!



Heap

E-F 4
E-G 9
D-C 6
D-A 12
D-B 4

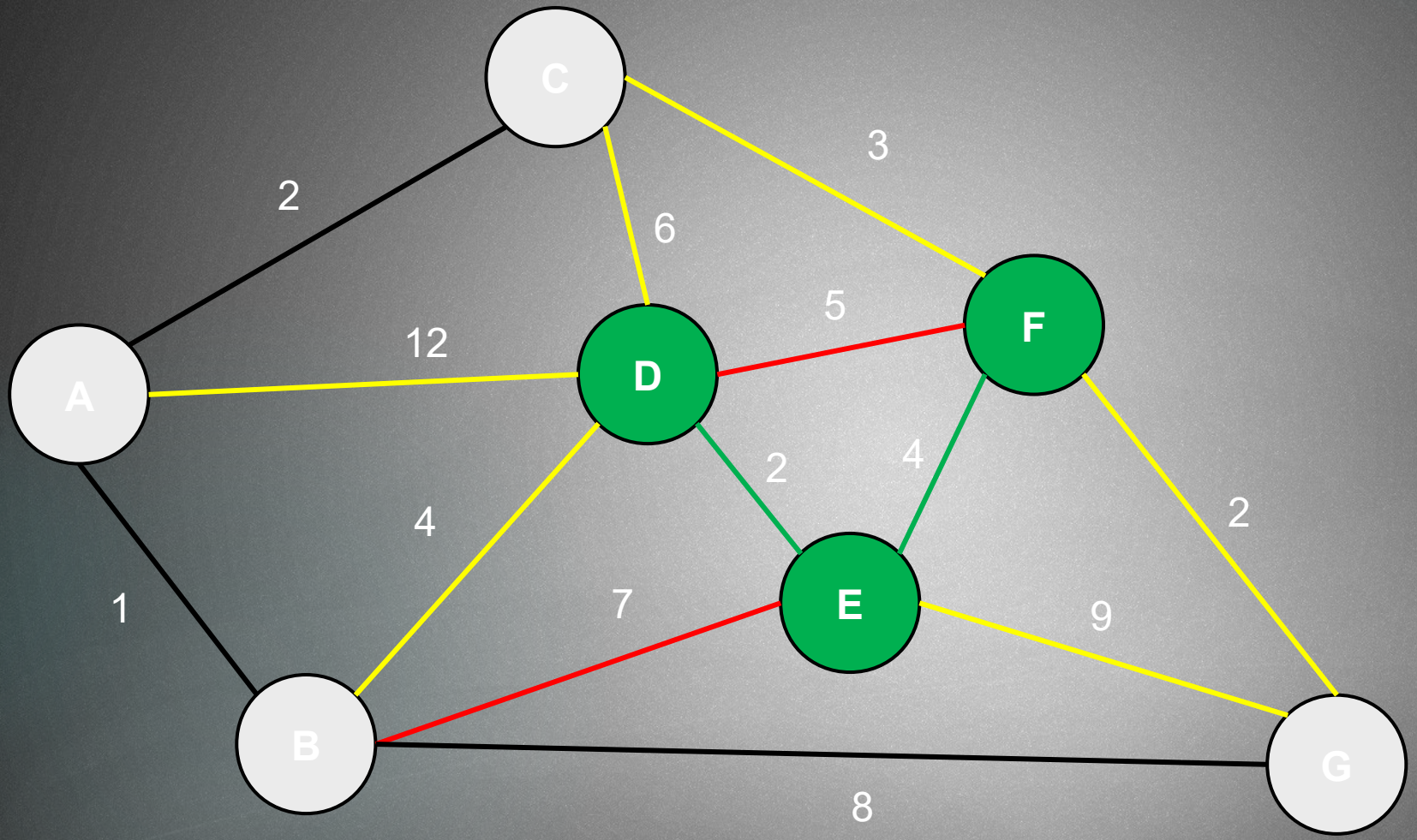
We iterate as far as we have not included all the vertexes in the spanning tree
So we iterate until all the vertexes are green !!!



Heap

E-G 9
D-C 6
D-A 12
D-B 4

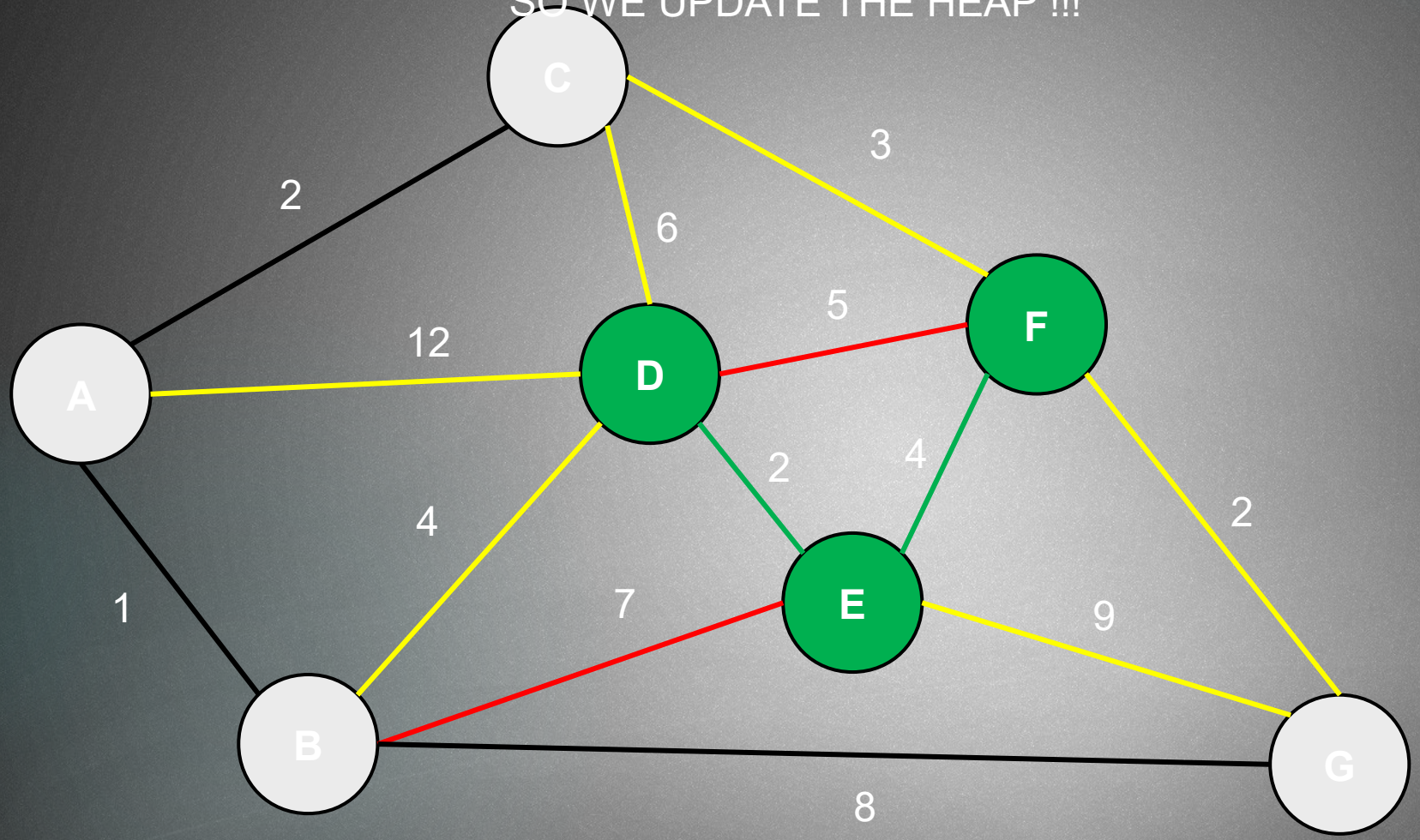
We iterate as far as we have not included all the vertexes in the spanning tree
So we iterate until all the vertexes are green !!!



Heap

E-G 9
D-C 6
D-A 12
D-B 4

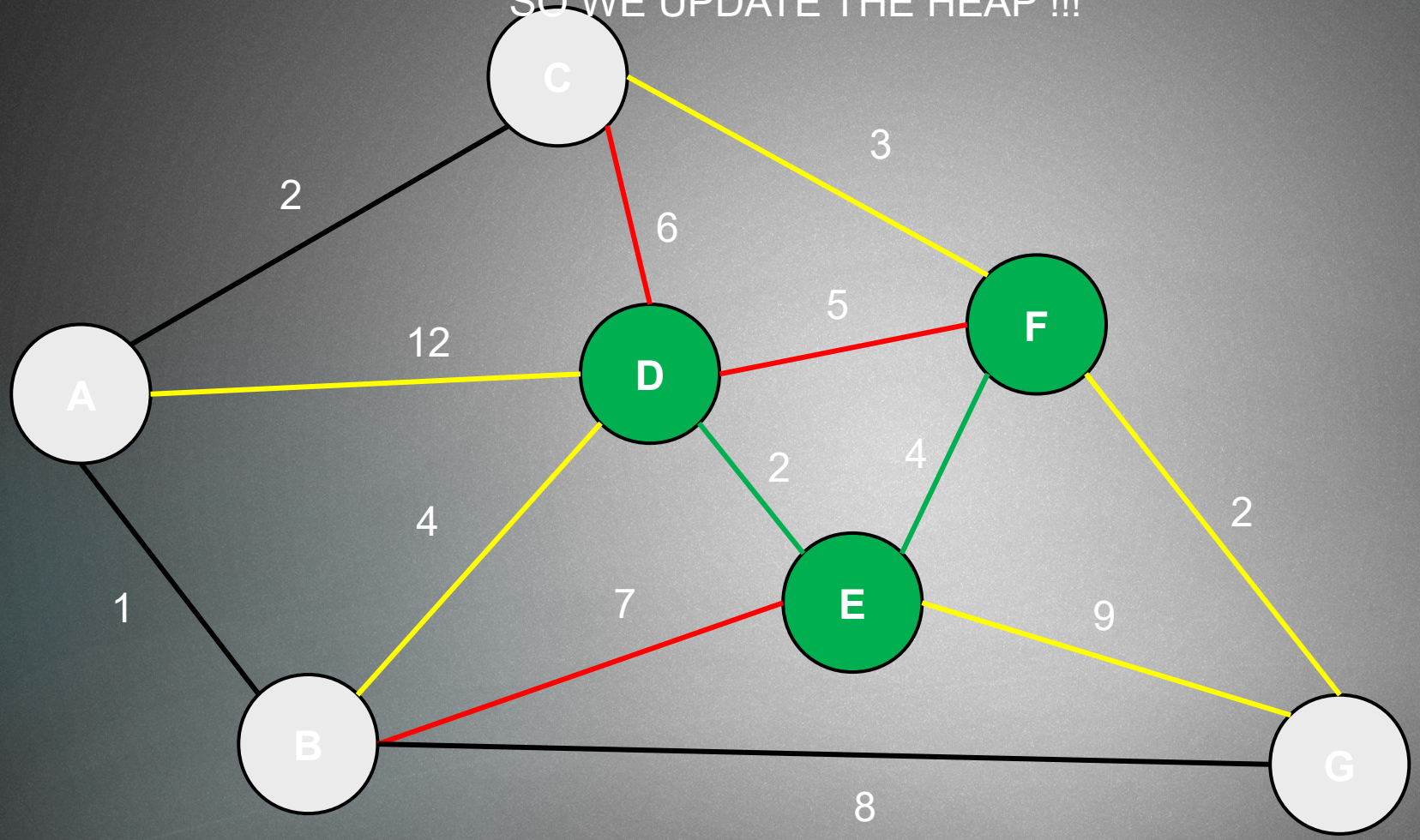
What about the F-C connection? We have considered a path to vertex C so far from vertex D. The cost from vertex D is 6, the cost from vertex F is 3
SO WE UPDATE THE HEAP !!!



Heap

E-G 9
D-C 6
D-A 12
D-B 4

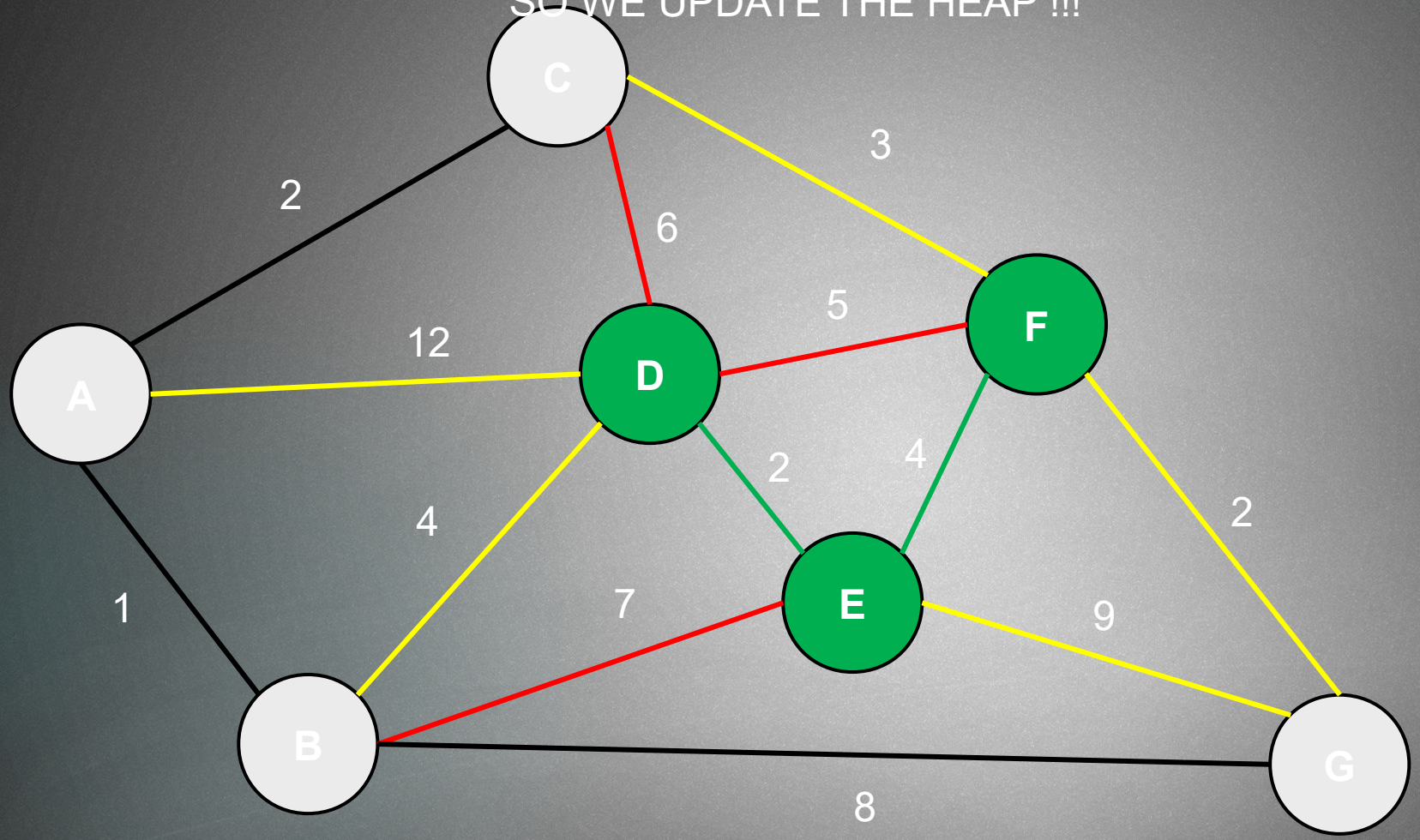
What about the F-C connection? We have considered a path to vertex C so far from vertex D. The cost from vertex D is 6, the cost from vertex F is 3
SO WE UPDATE THE HEAP !!!



Heap

E-G 9
D-C 6
D-A 12
D-B 4

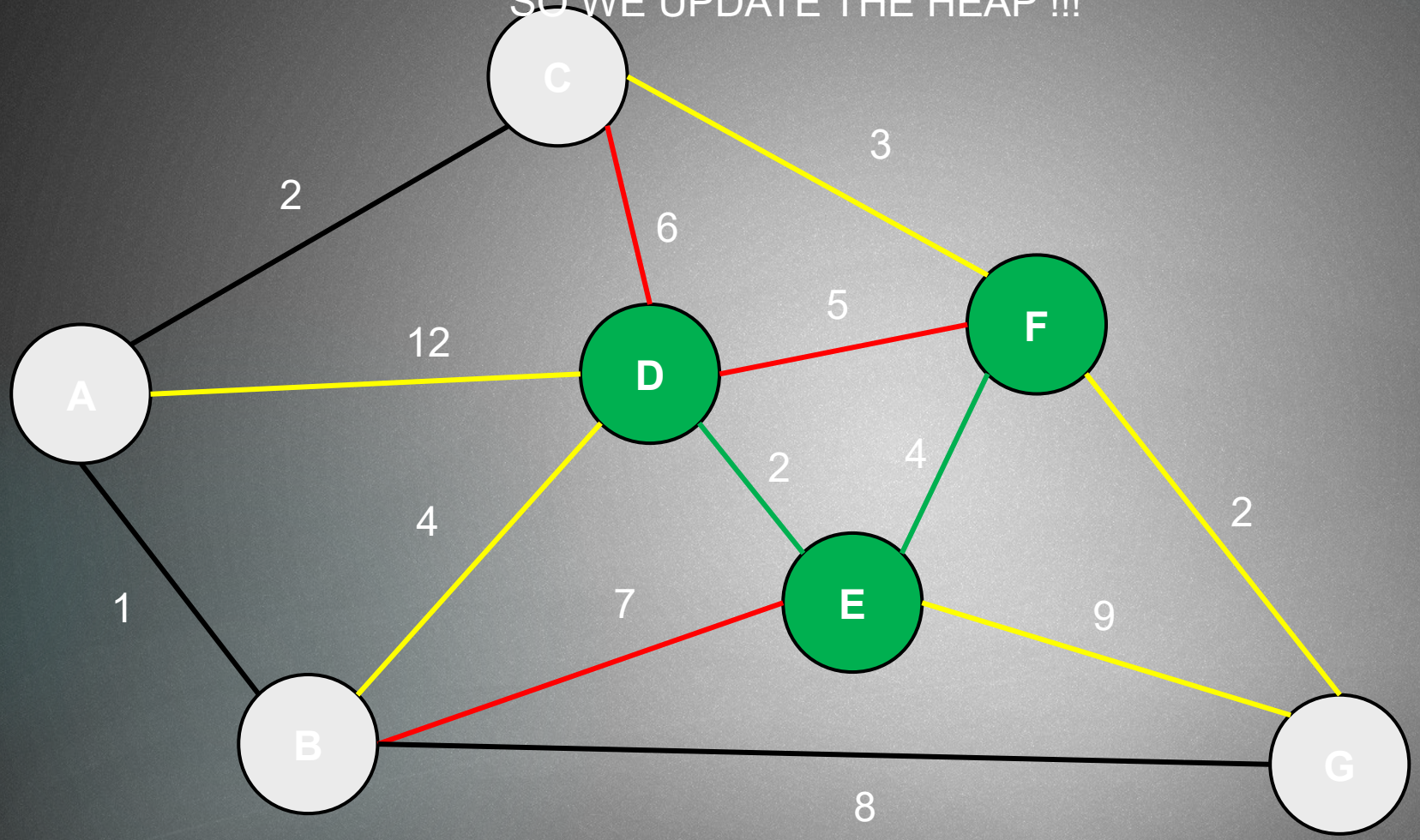
What about the F-C connection? We have considered a path to vertex C so far from vertex D. The cost from vertex D is 6, the cost from vertex F is 3
SO WE UPDATE THE HEAP !!!



Heap

E-G 9
D-A 12
D-B 4

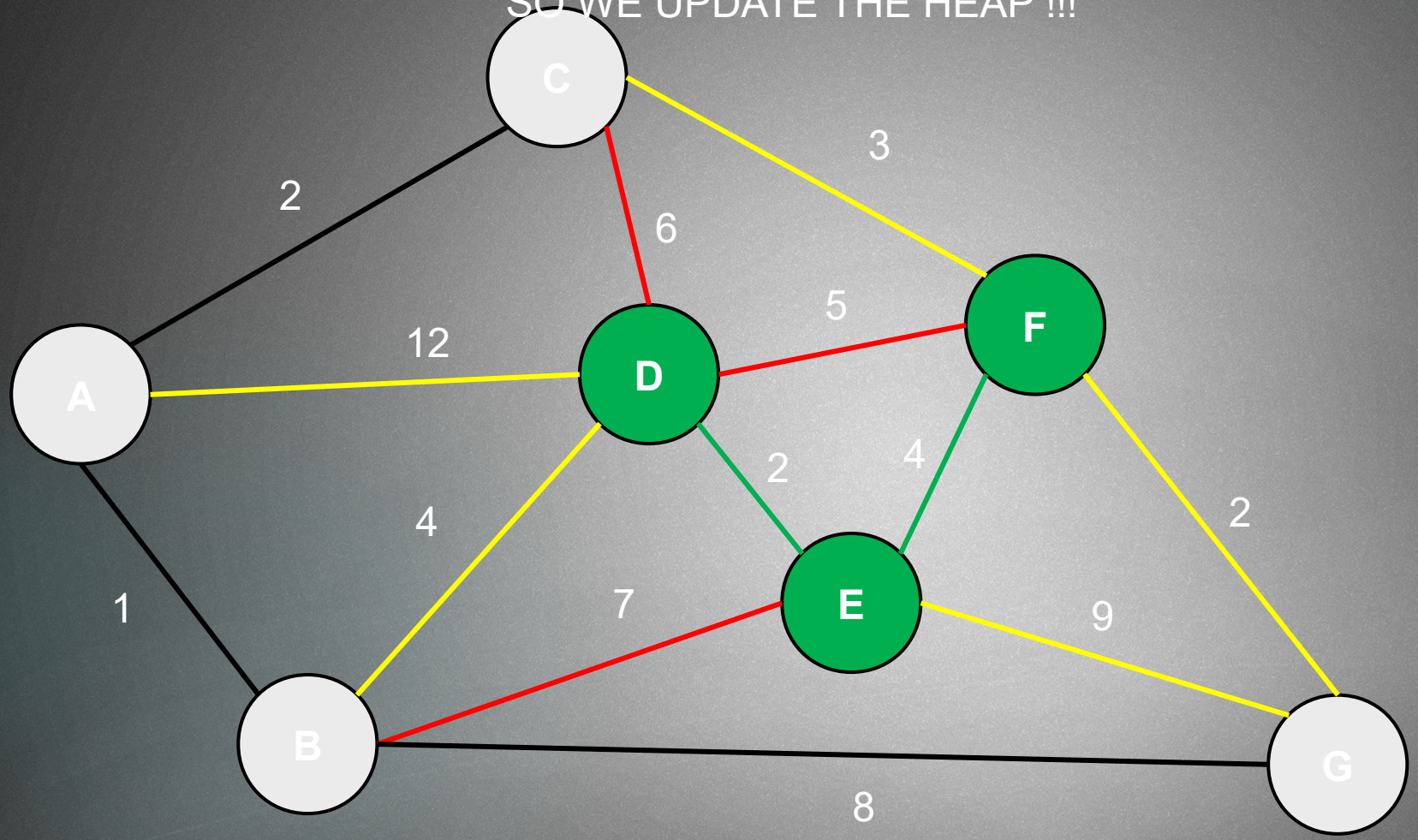
What about the F-C connection? We have considered a path to vertex C so far from vertex D. The cost from vertex D is 6, the cost from vertex F is 3
SO WE UPDATE THE HEAP !!!



Heap

E-G 9
D-A 12
D-B 4
F-C 3

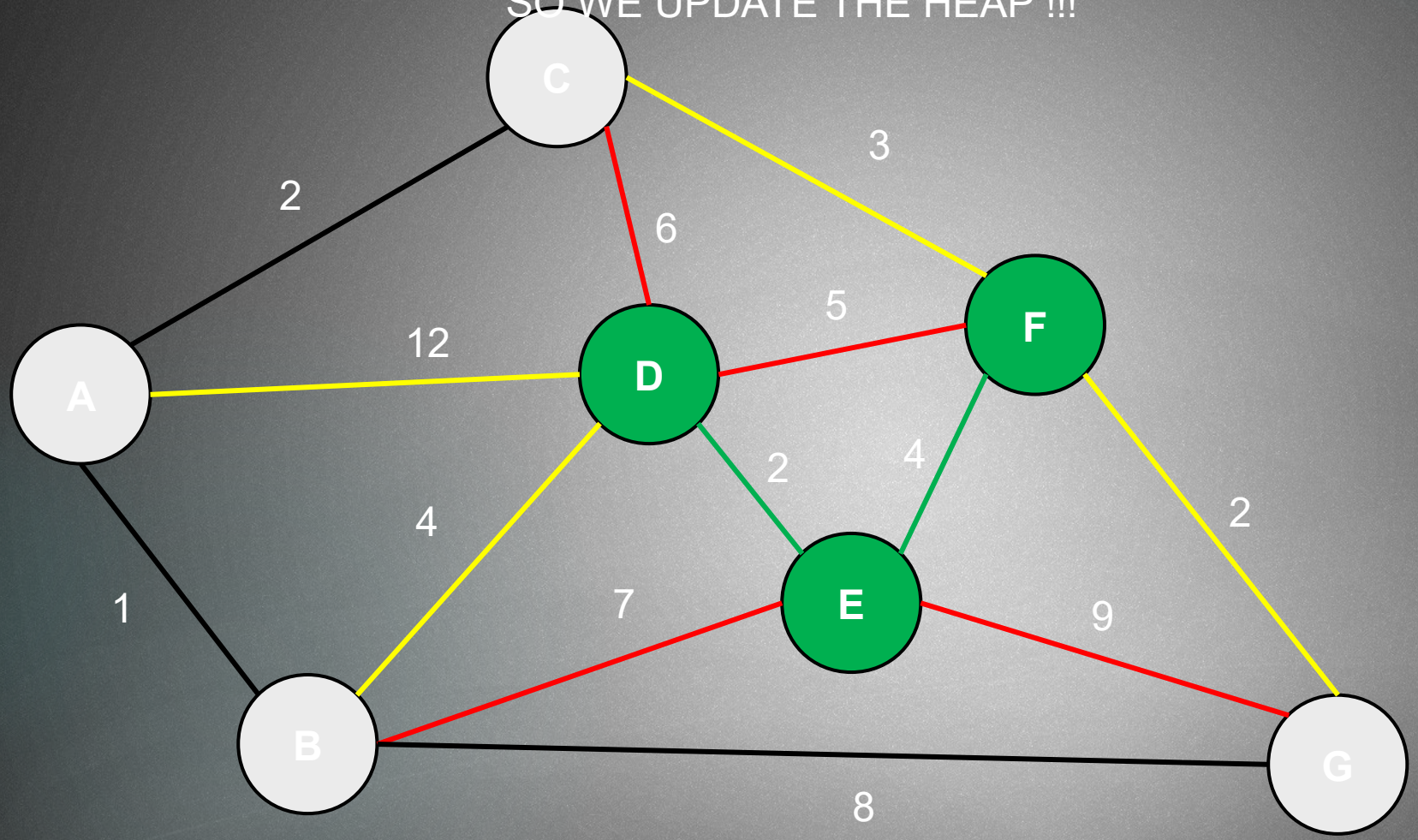
What about the F-G connection? We have considered a path to vertex G so far from vertex E. The cost from vertex E is 9, the cost from vertex F is 2
SO WE UPDATE THE HEAP !!!



Heap

E-G 9
D-A 12
D-B 4
F-C 3

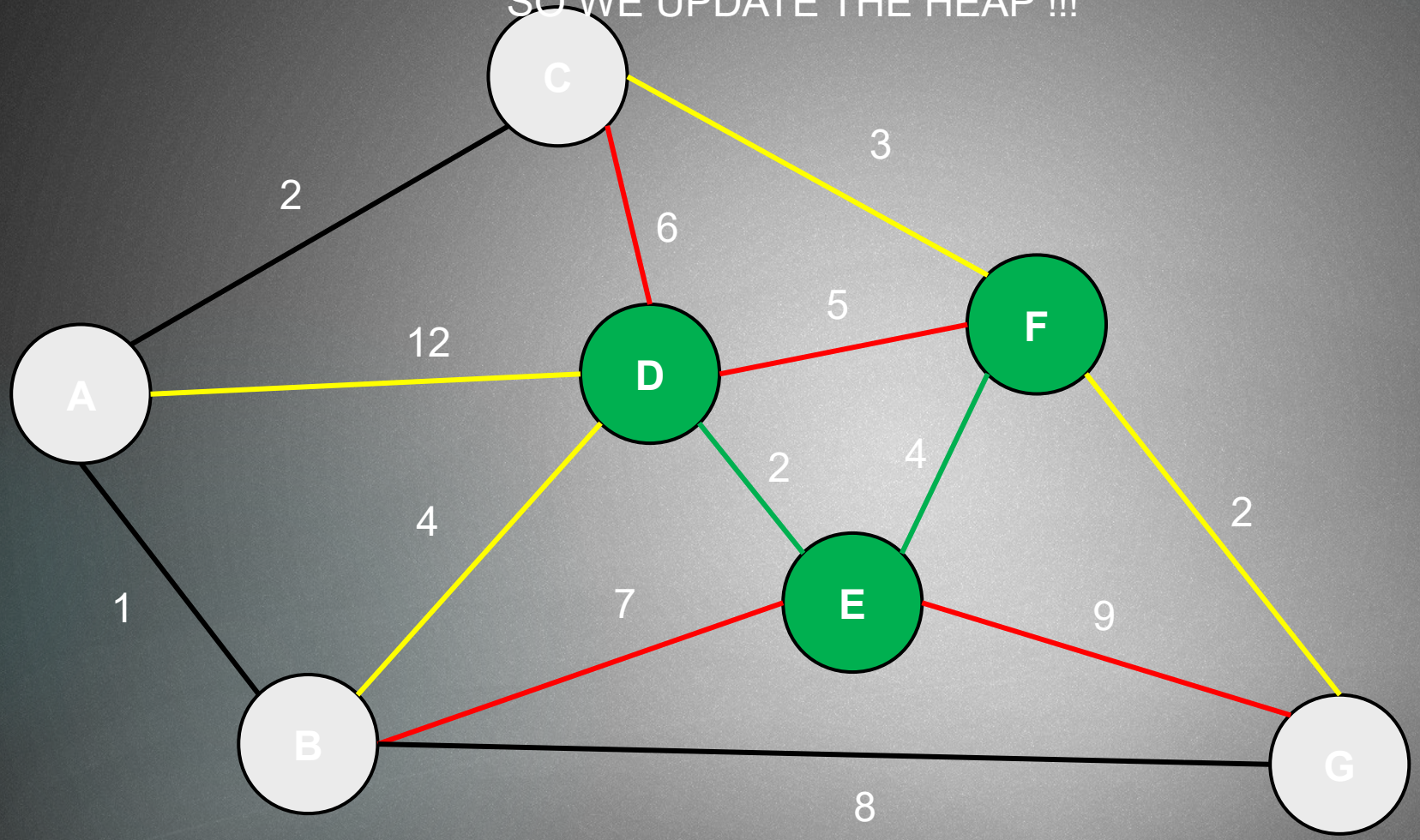
What about the F-G connection? We have considered a path to vertex G so far from vertex E. The cost from vertex E is 9, the cost from vertex F is 2
SO WE UPDATE THE HEAP !!!



Heap

E-G 9
D-A 12
D-B 4
F-C 3

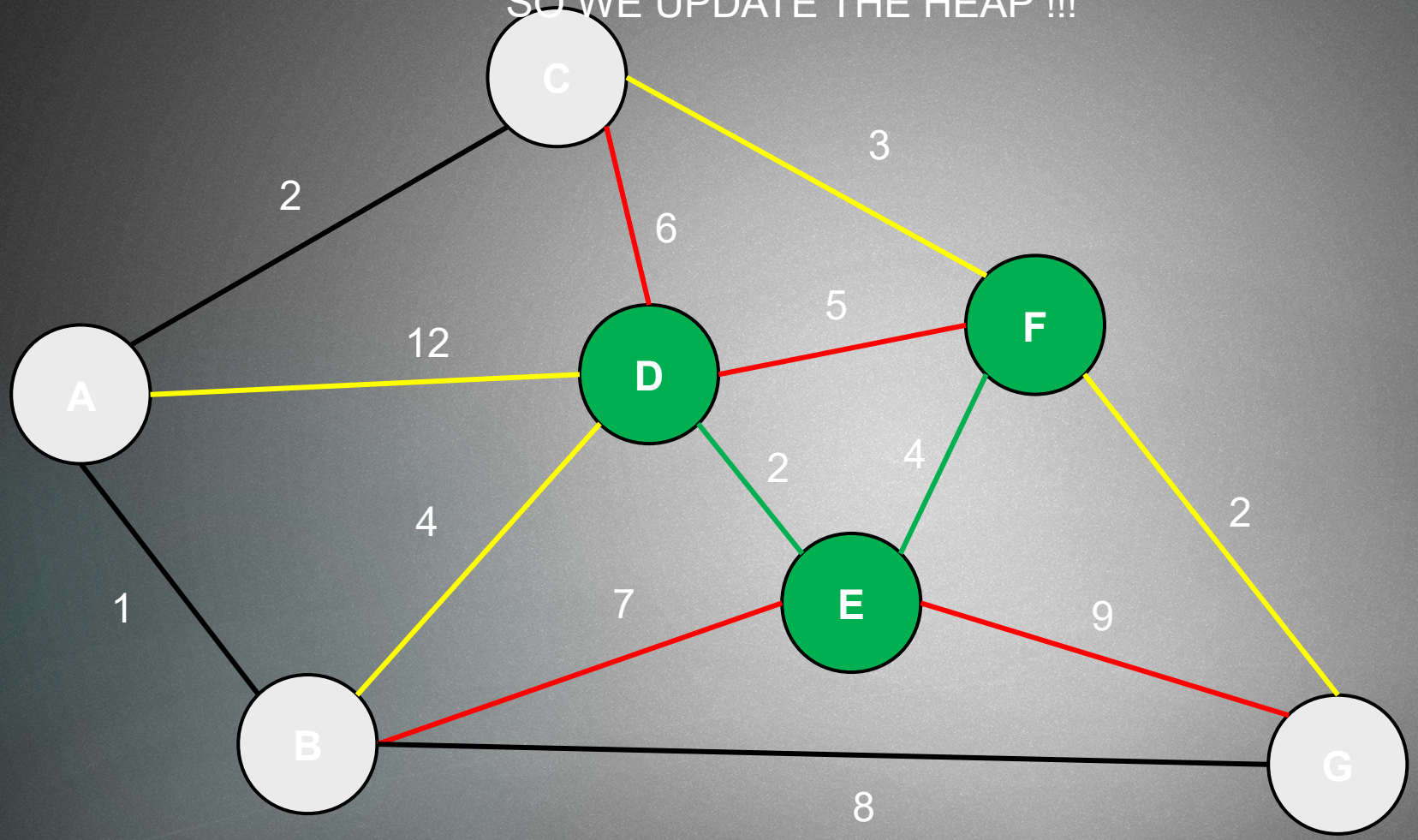
What about the F-G connection? We have considered a path to vertex G so far from vertex E. The cost from vertex E is 9, the cost from vertex F is 2
SO WE UPDATE THE HEAP !!!



Heap

D-A 12
D-B 4
F-C 3

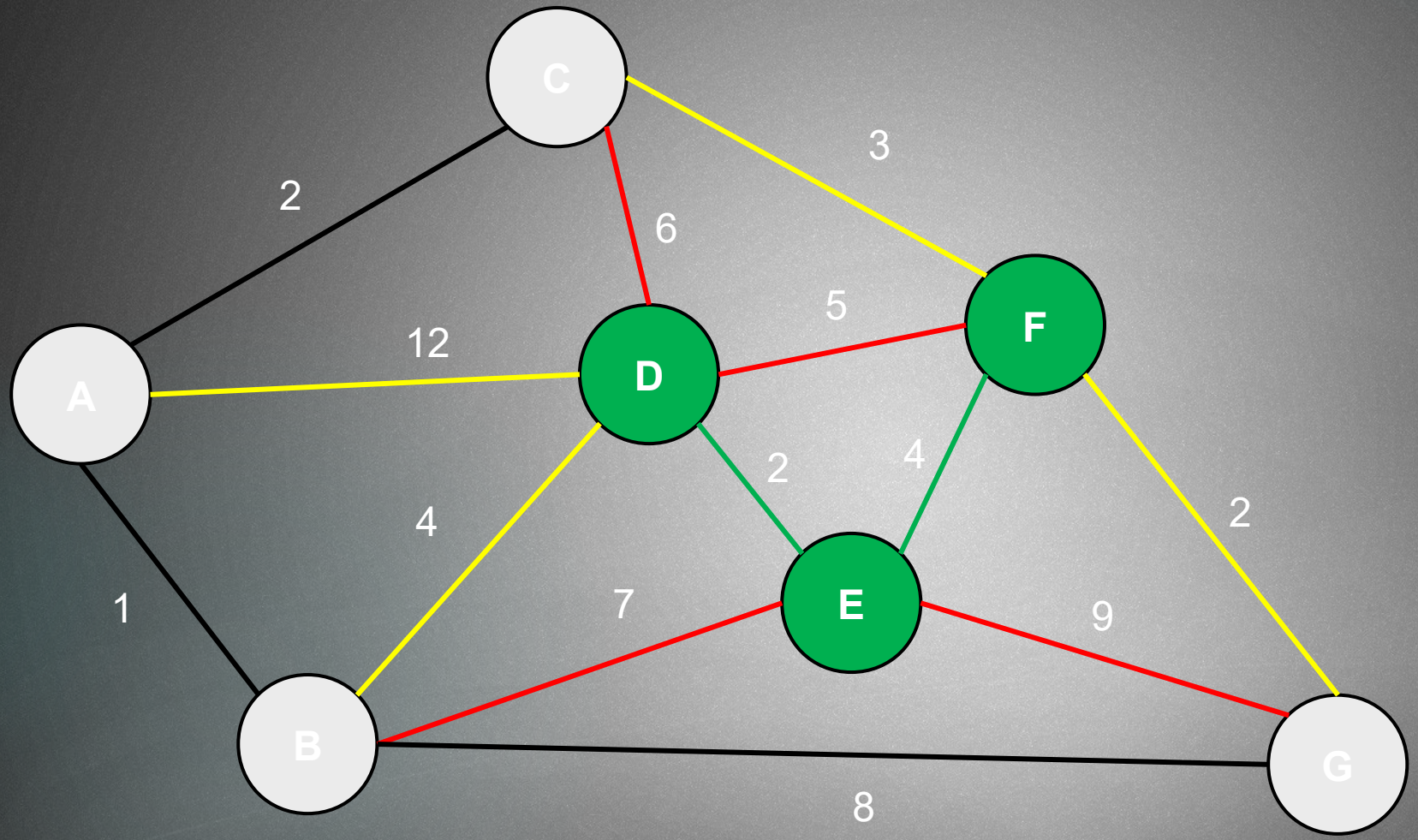
What about the F-G connection? We have considered a path to vertex G so far from vertex E. The cost from vertex E is 9, the cost from vertex F is 2
SO WE UPDATE THE HEAP !!!



Heap

D-A 12
D-B 4
F-C 3
F-G 2

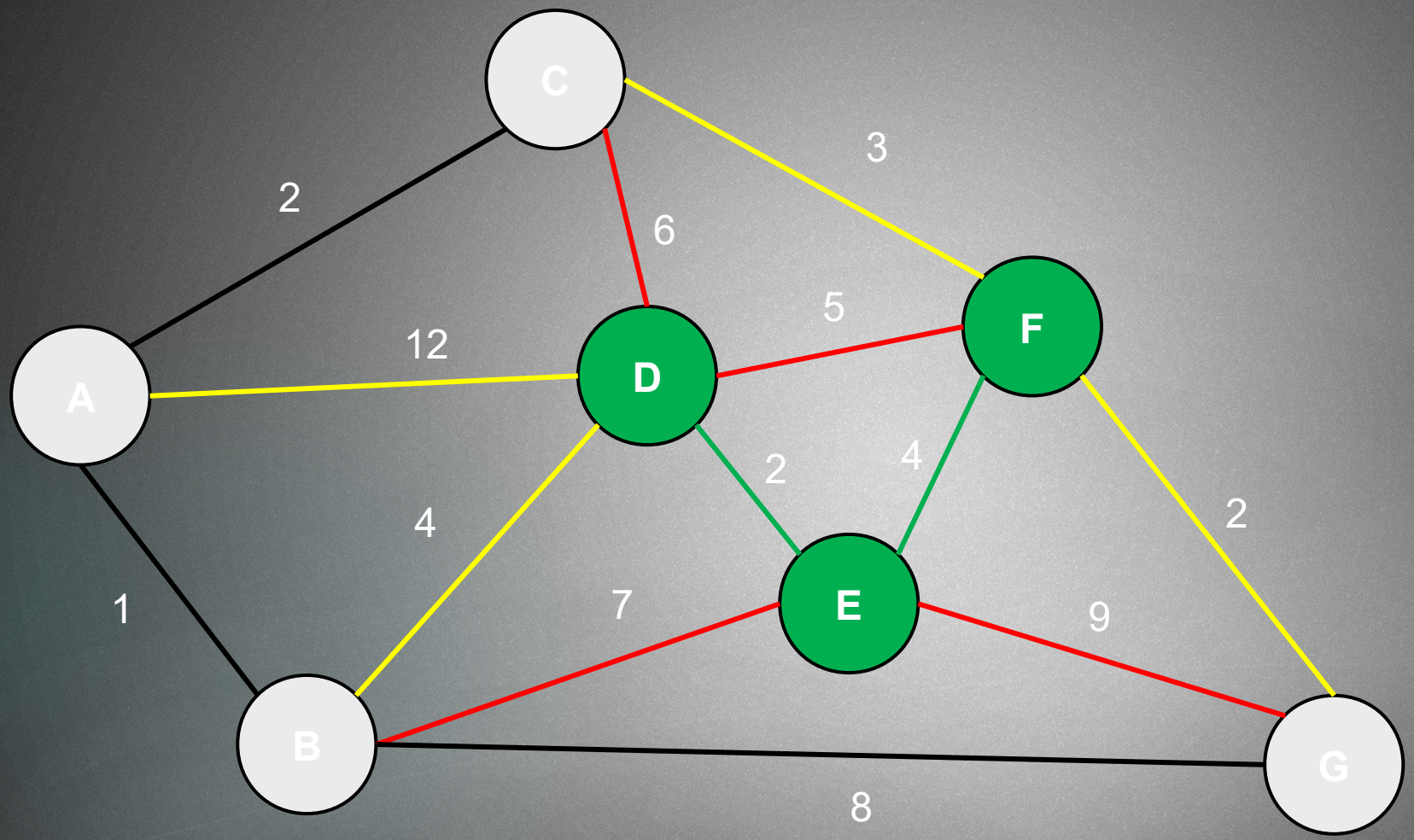
We keep going because we do not have all the vertexes in the spanning tree



Heap

D-A 12
D-B 4
F-C 3
F-G 2

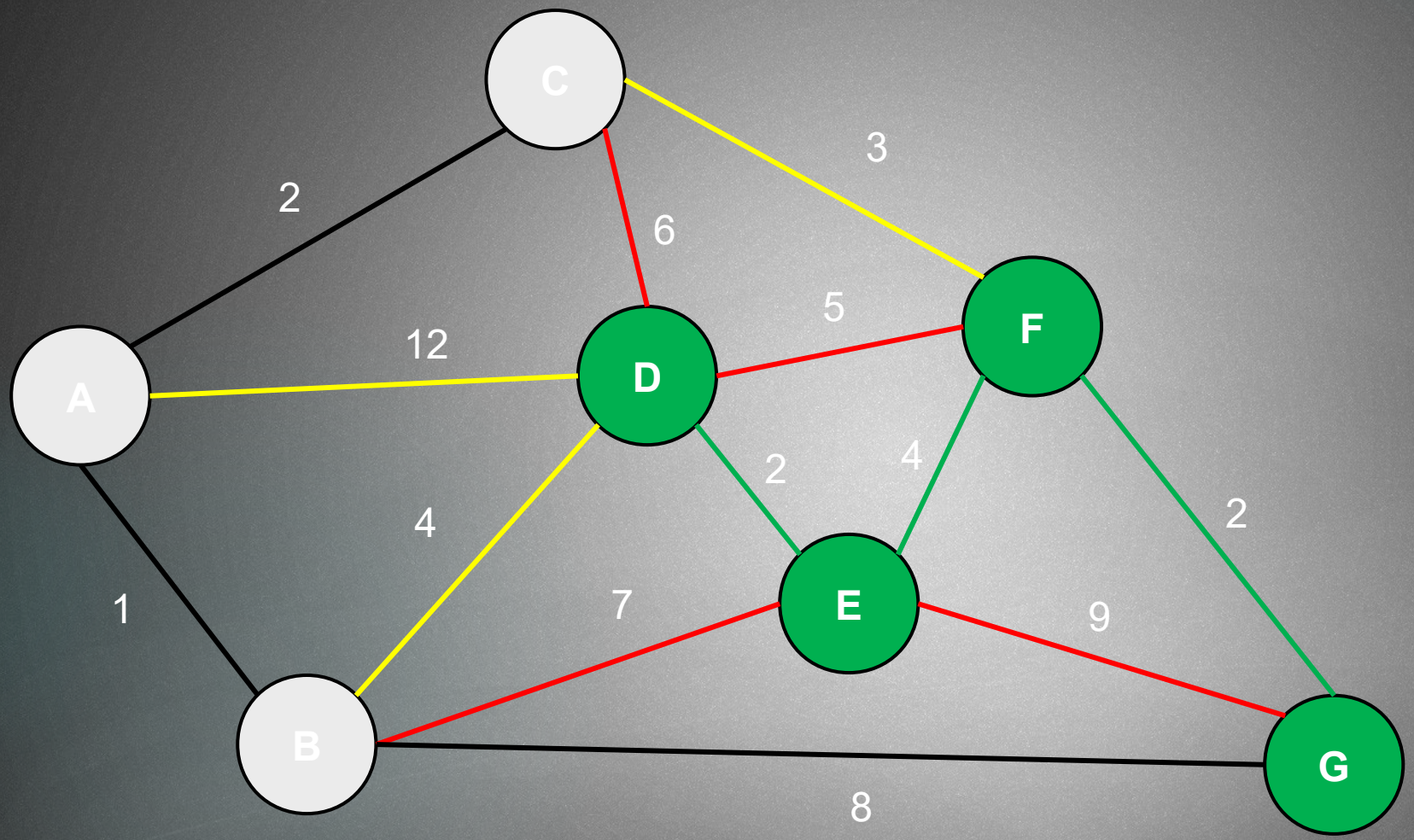
We keep going because we do not have all the vertexes in the spanning tree



Heap

D-A 12
D-B 4
F-C 3
F-G 2

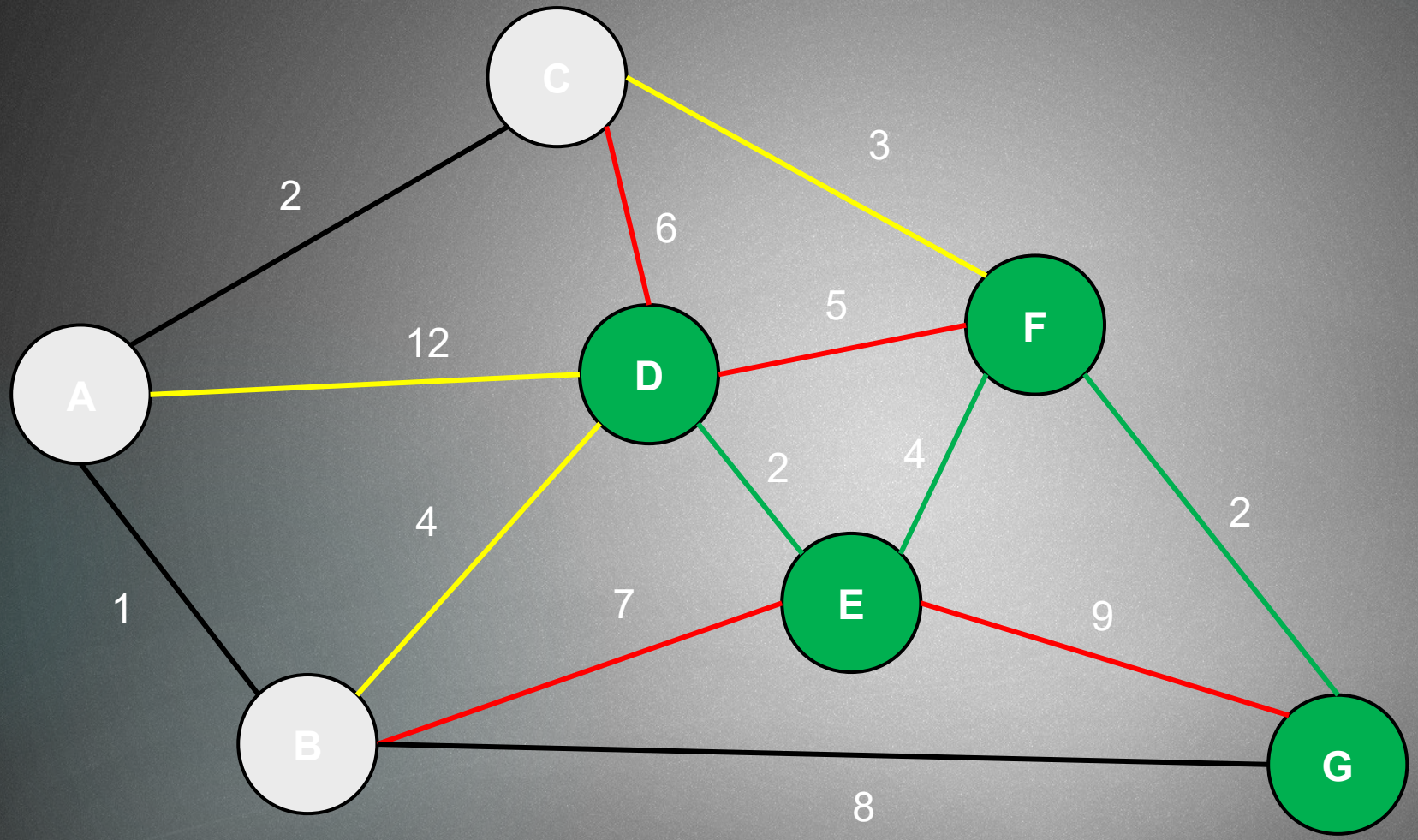
We keep going because we do not have all the vertexes in the spanning tree



Heap

D-A 12
D-B 4
F-C 3
F-G 2

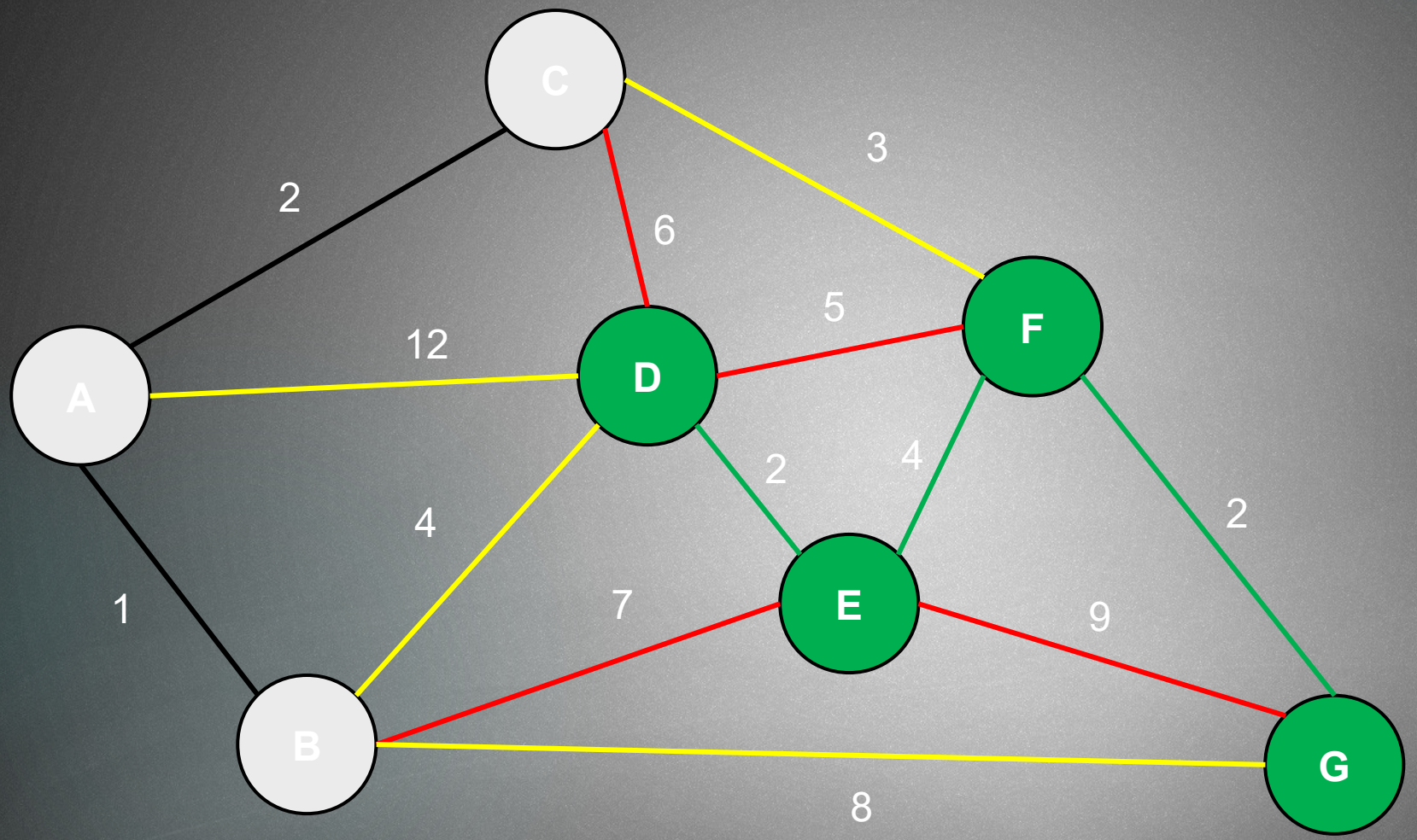
We keep going because we do not have all the vertexes in the spanning tree



Heap

D-A 12
D-B 4
F-C 3

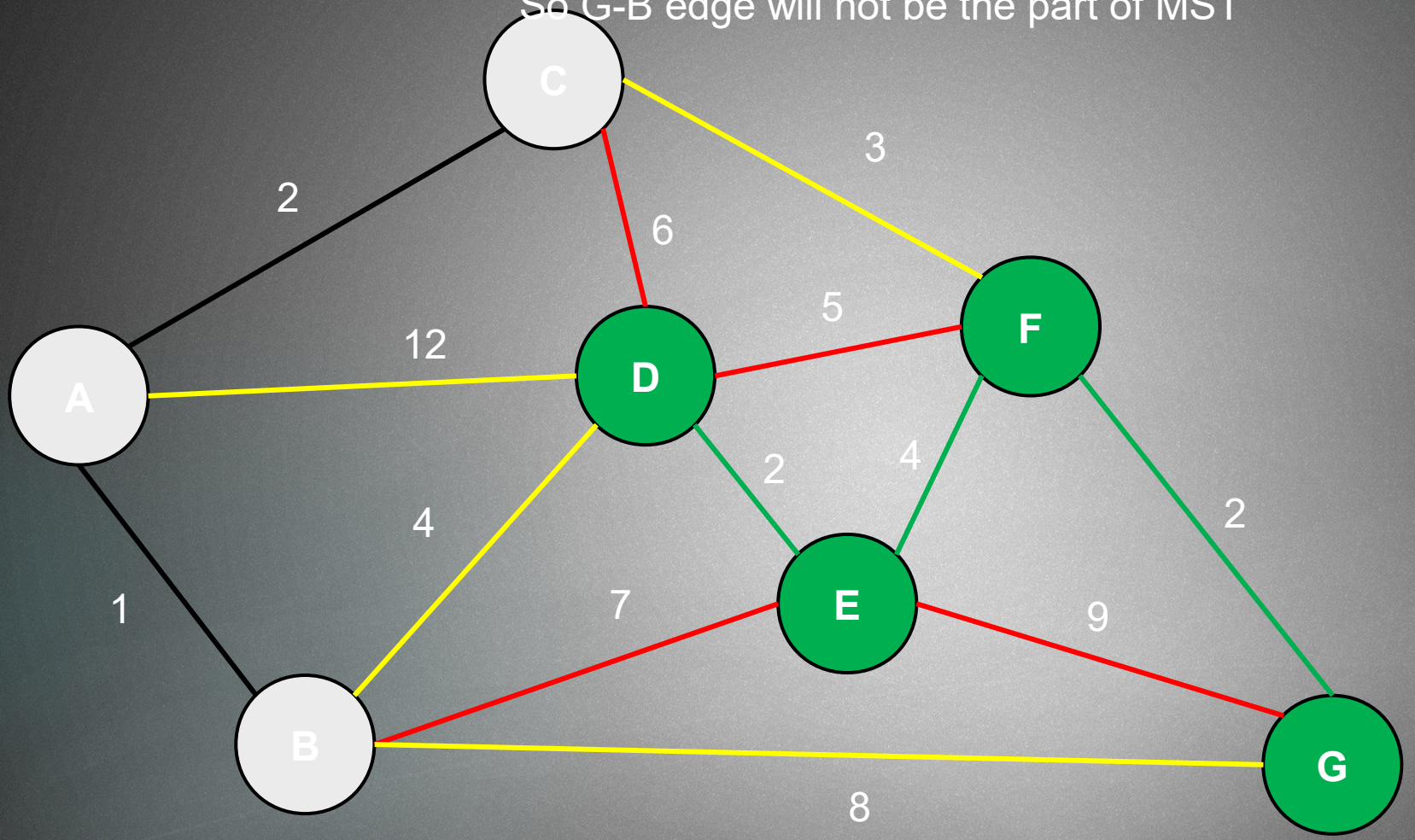
We keep going because we do not have all the vertexes in the spanning tree



Heap

D-A 12
D-B 4
F-C 3

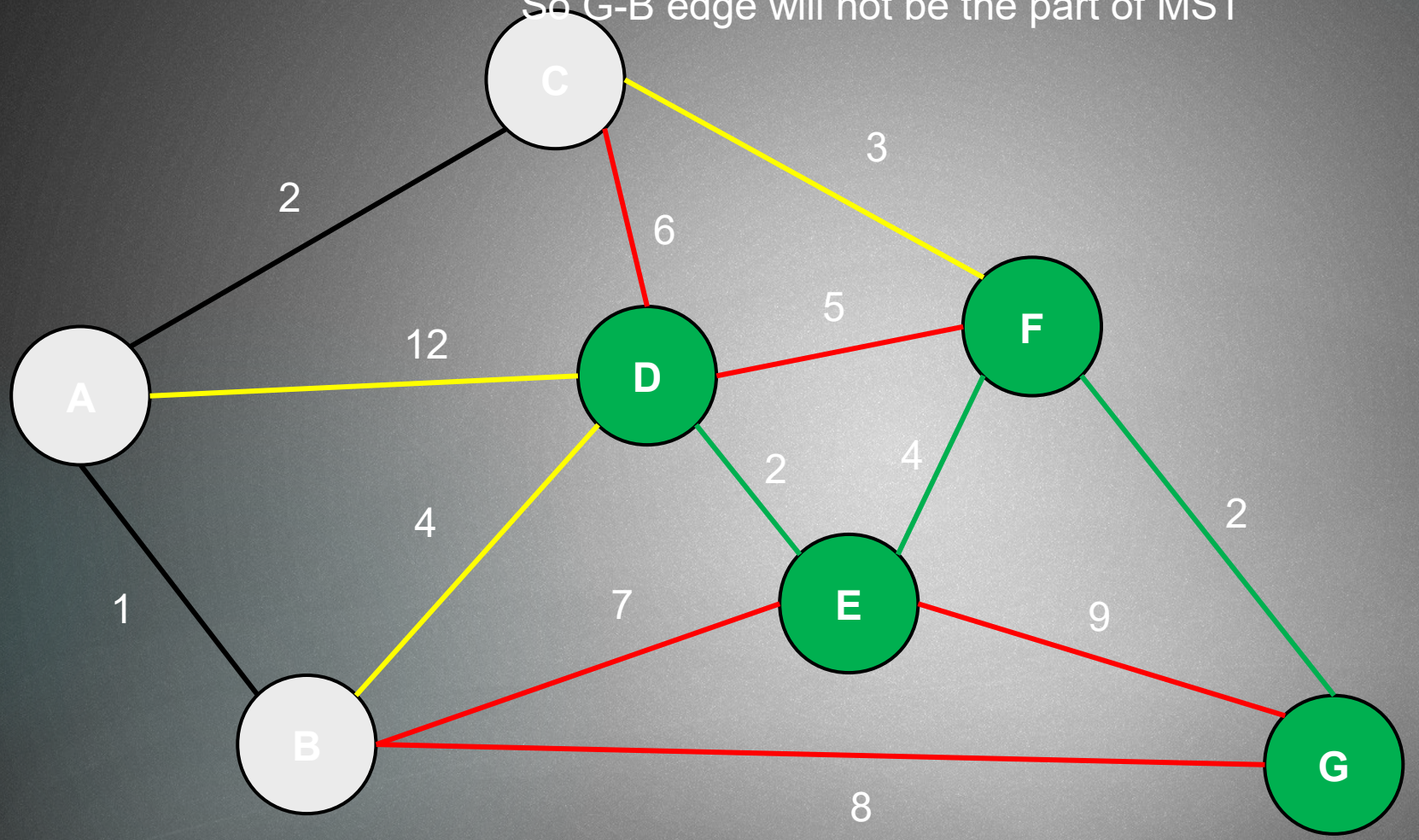
What about the G-B connection? We have considered a path to vertex B so far from vertex D. The cost from vertex D is 4 , the cost from vertex G is 8
So G-B edge will not be the part of MST



Heap

D-A 12
D-B 4
F-C 3

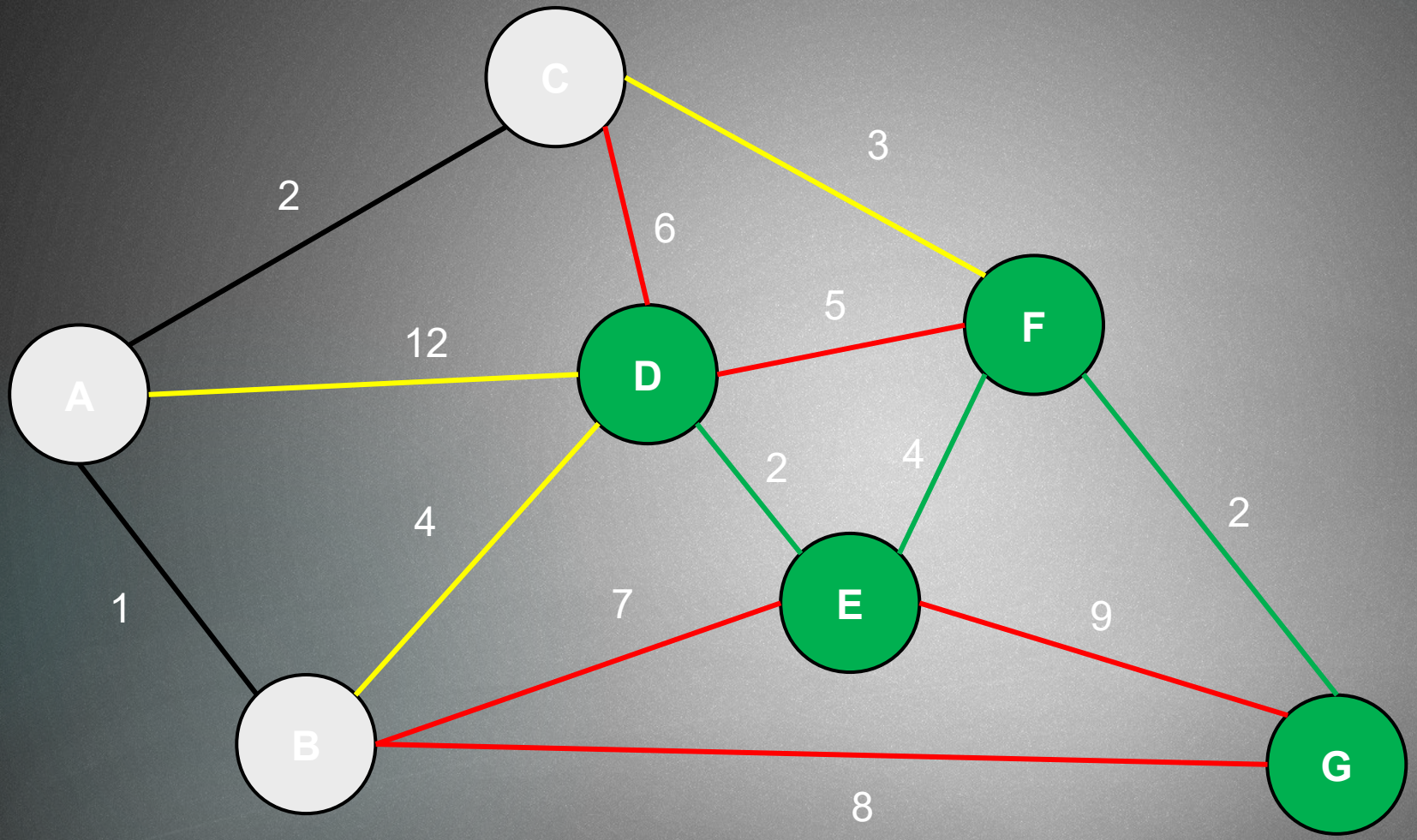
What about the G-B connection? We have considered a path to vertex B so far from vertex D. The cost from vertex D is 4 , the cost from vertex G is 8
So G-B edge will not be the part of MST



Heap

D-A 12
D-B 4
F-C 3

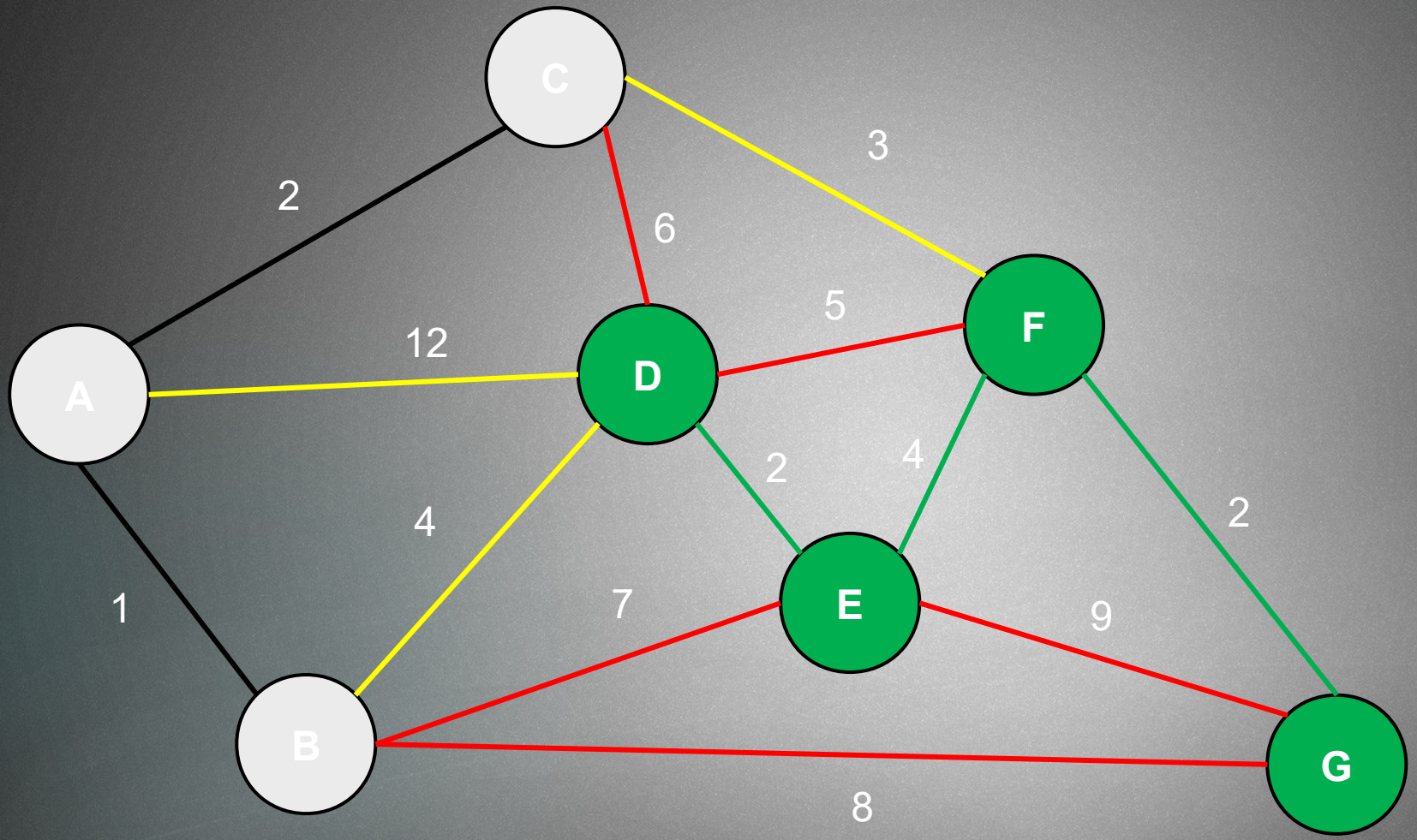
We keep going because not all the vertexes are included in the MST



Heap

D-A 12
D-B 4
F-C 3

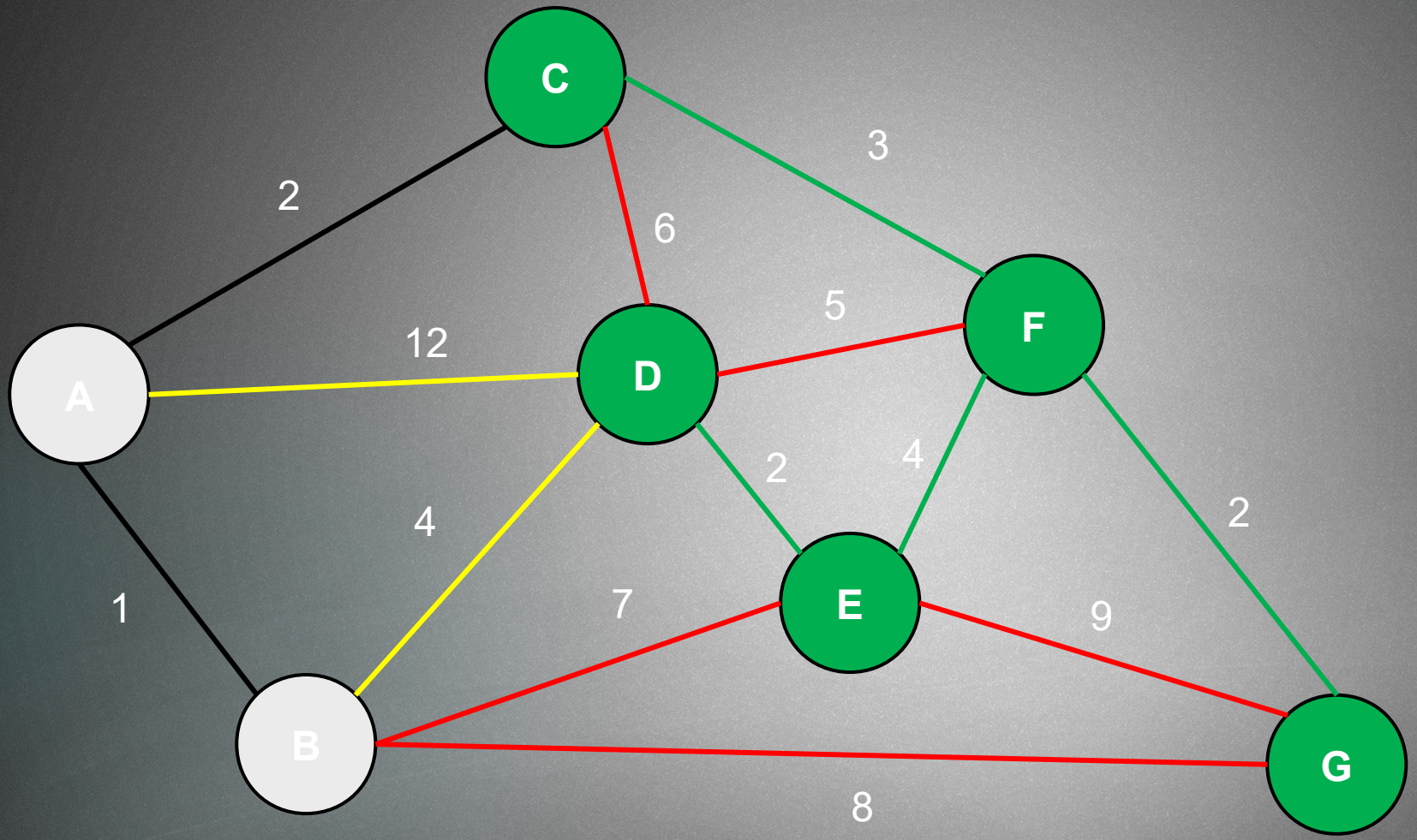
We keep going because not all the vertexes are included in the MST



Heap

D-A 12
D-B 4
F-C 3

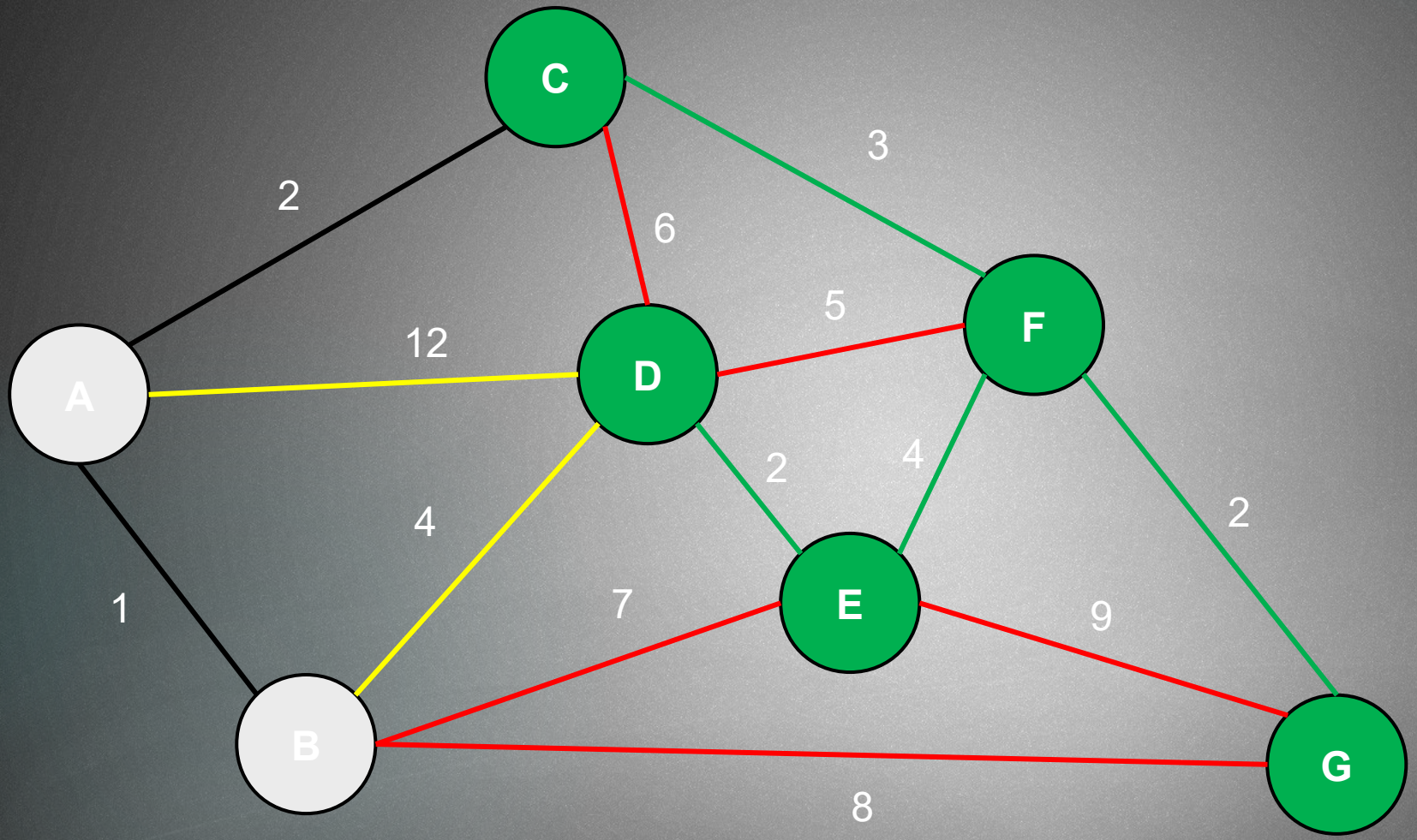
We keep going because not all the vertexes are included in the MST



Heap

D-A 12
D-B 4
F-C 3

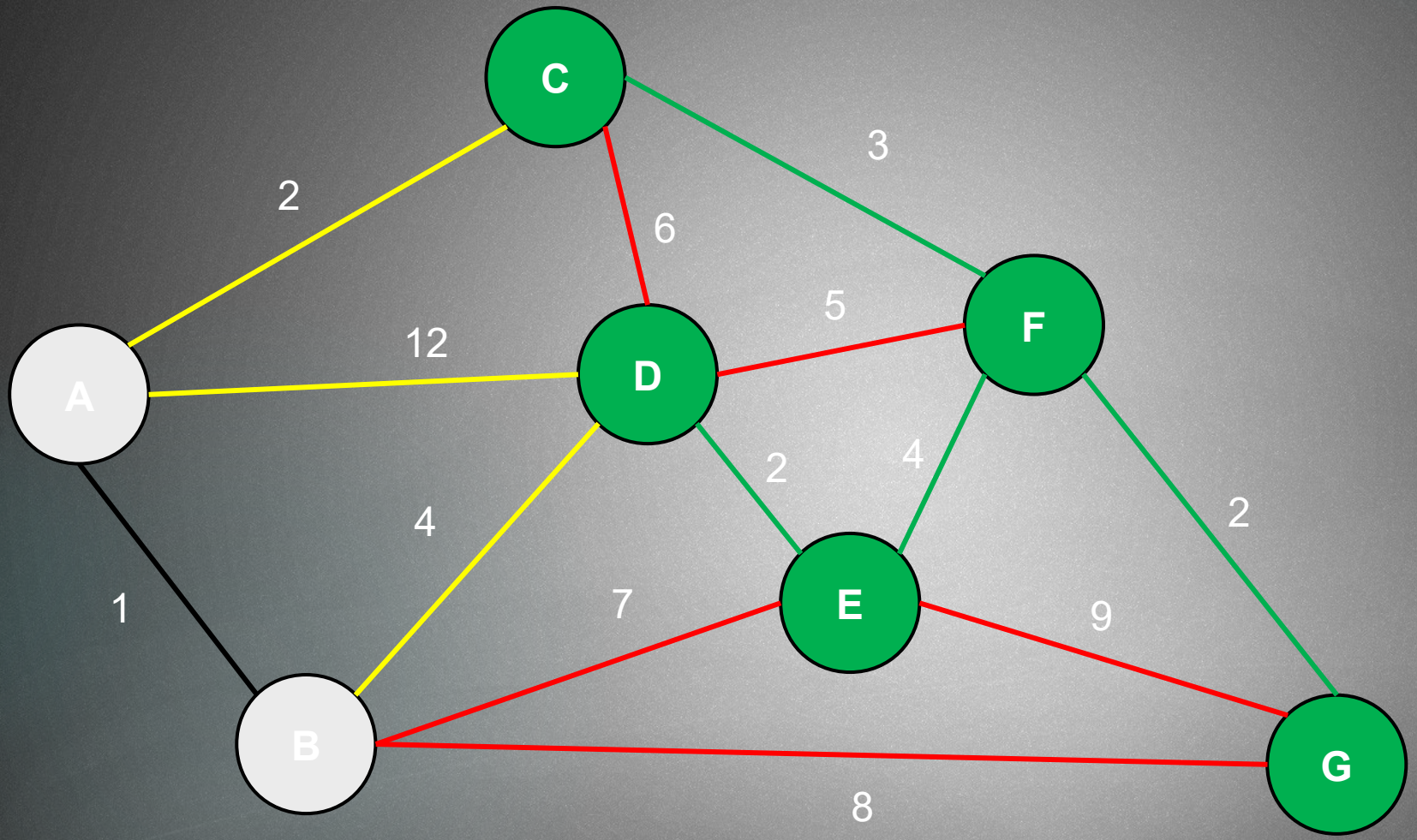
We keep going because not all the vertexes are included in the MST



Heap

D-A 12
D-B 4

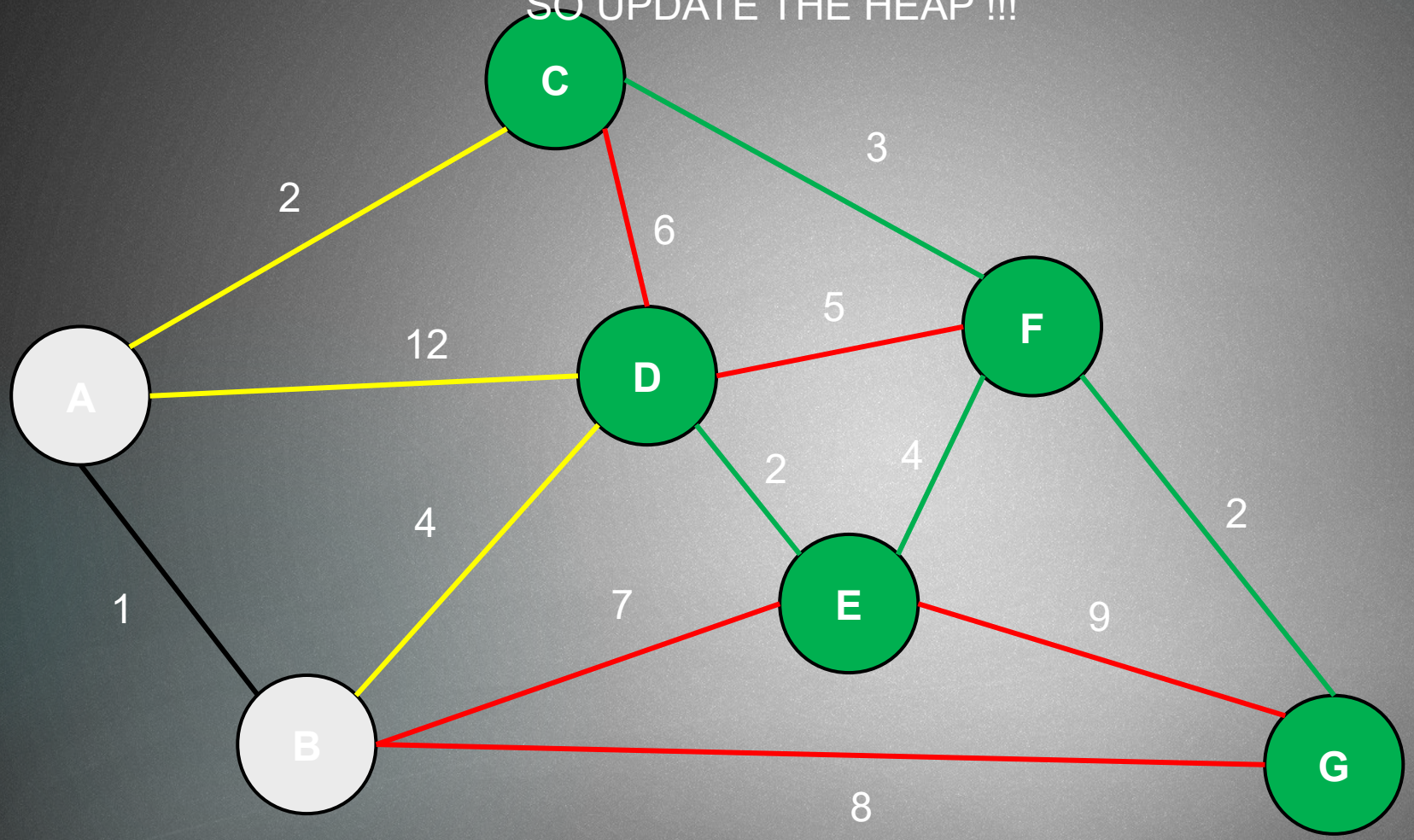
We keep going because not all the vertexes are included in the MST



Heap

D-A 12
D-B 4

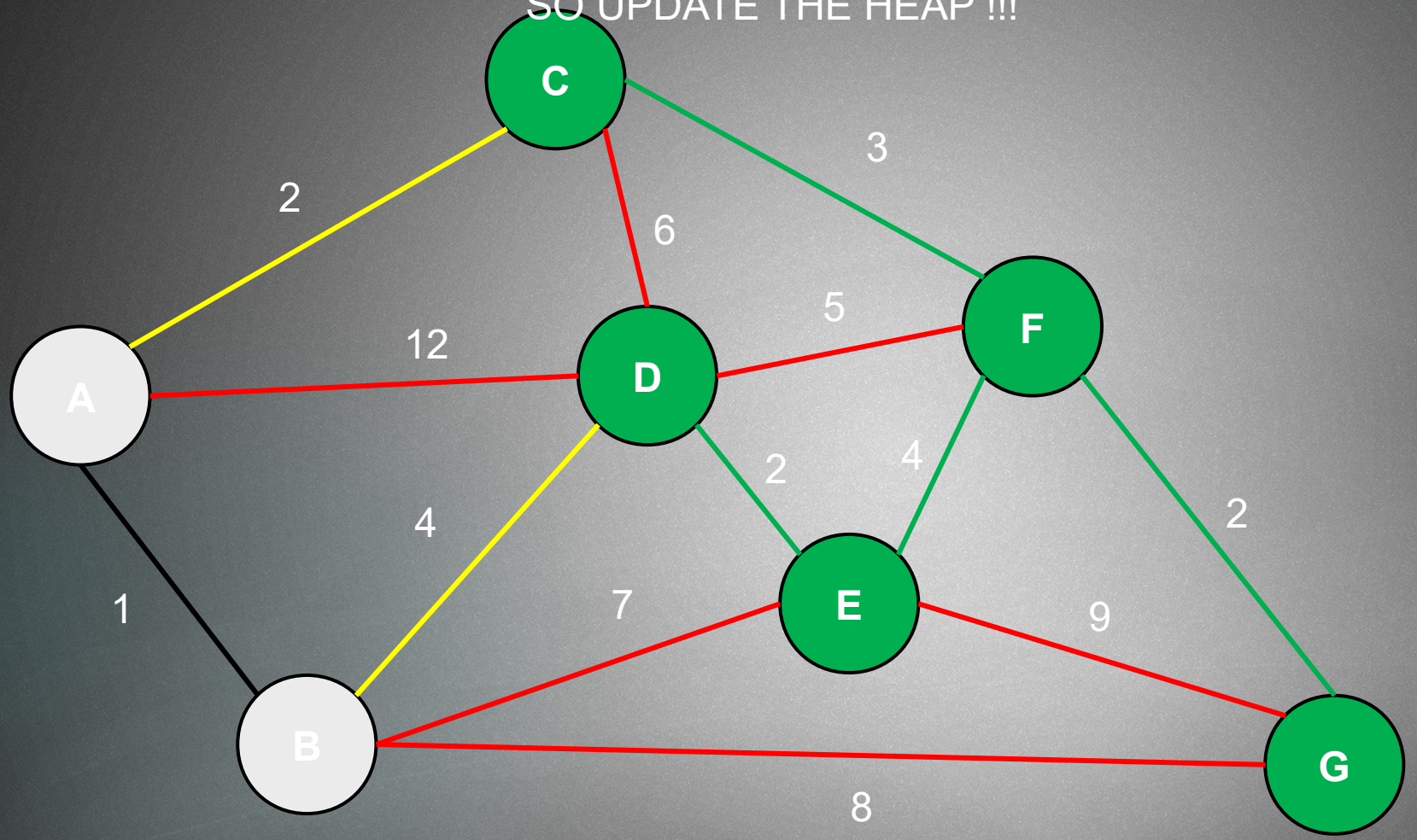
What about the C-A connection? We have considered a path to vertex A so far from vertex D. The cost from vertex D is 12 , the cost from vertex C is 2 SO UPDATE THE HEAP !!!



Heap

D-A 12
D-B 4

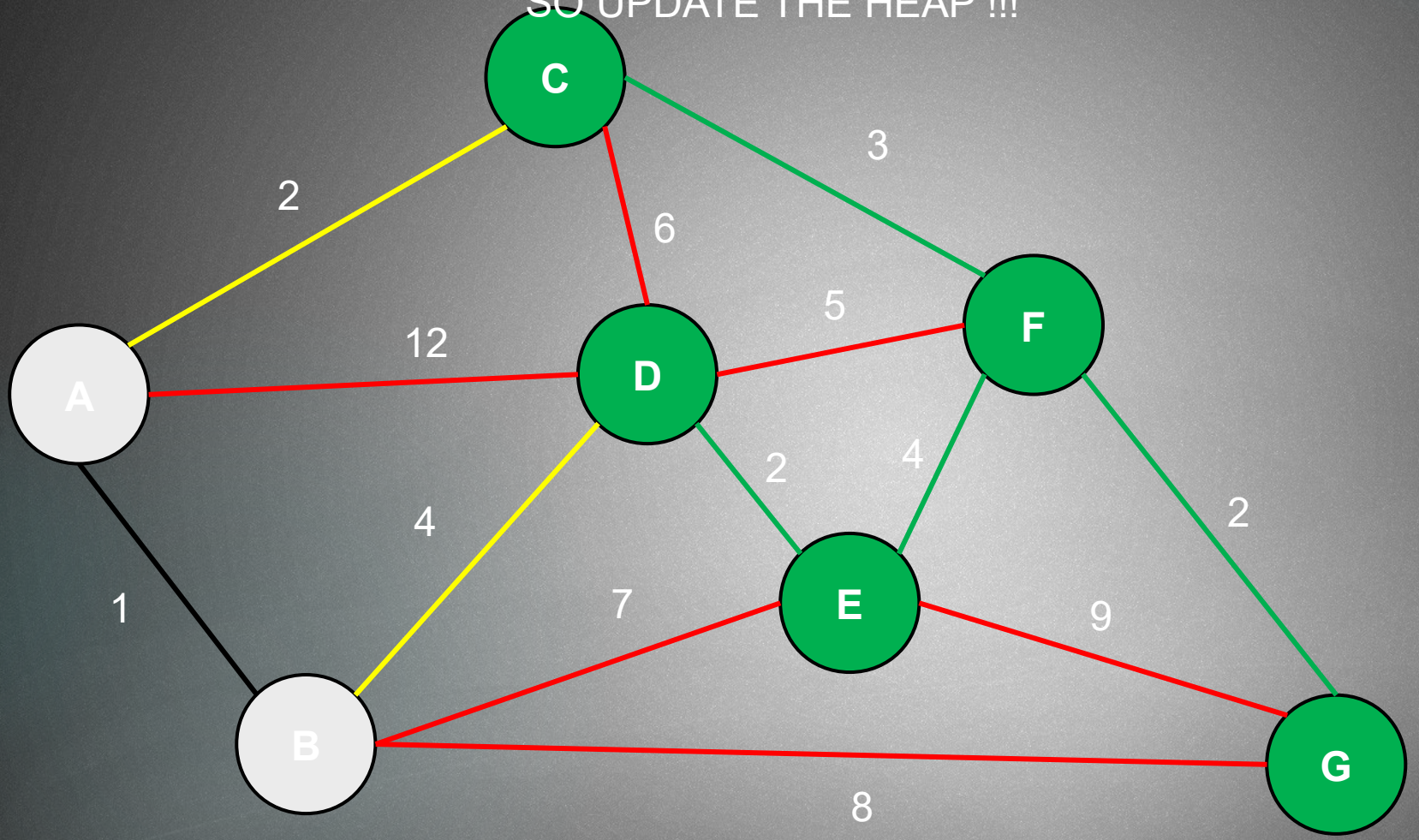
What about the C-A connection? We have considered a path to vertex A so far from vertex D. The cost from vertex D is 12 , the cost from vertex C is 2 SO UPDATE THE HEAP !!!



Heap

D-A 12
D-B 4

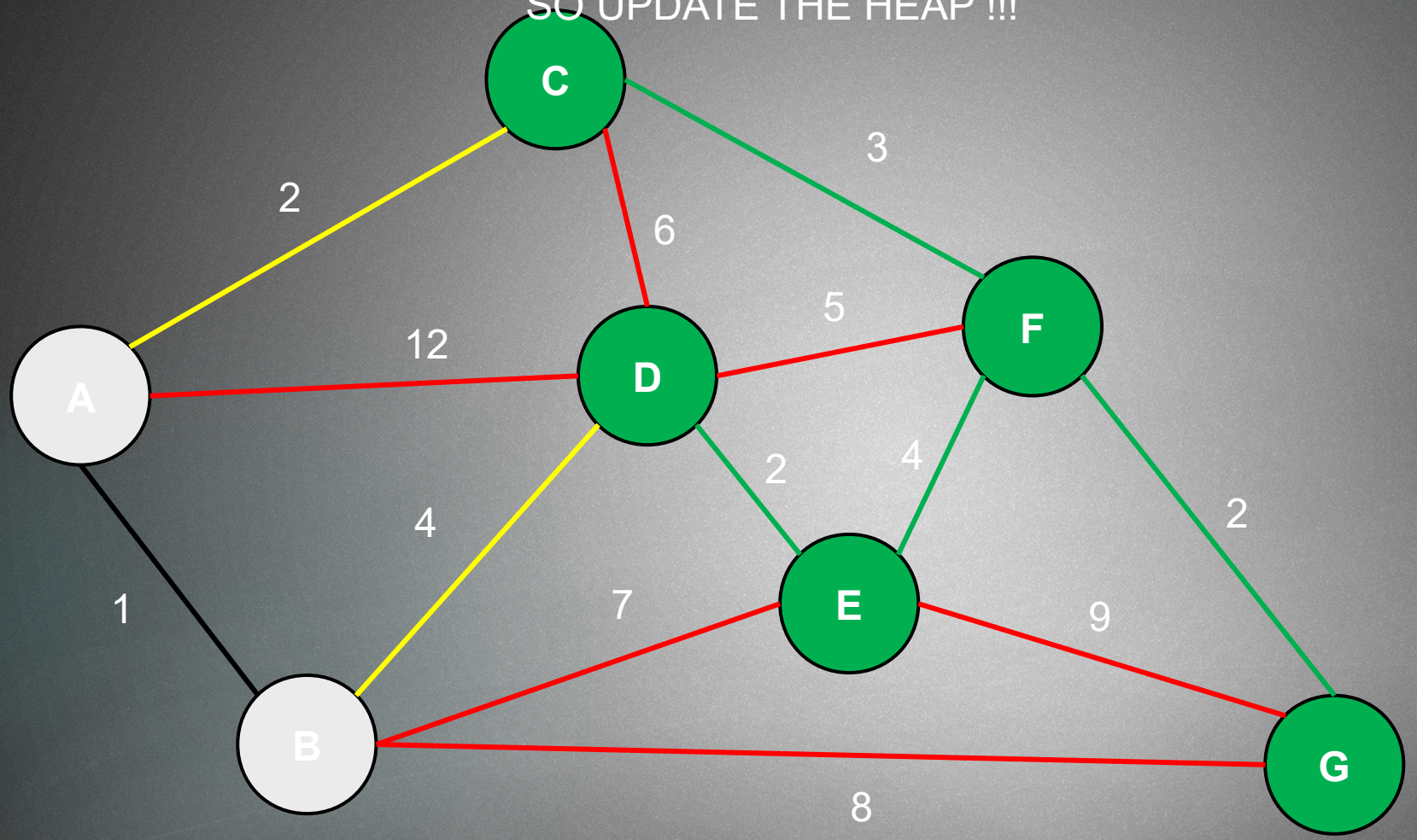
What about the C-A connection? We have considered a path to vertex A so far from vertex D. The cost from vertex D is 12 , the cost from vertex C is 2 SO UPDATE THE HEAP !!!



Heap

D-B 4

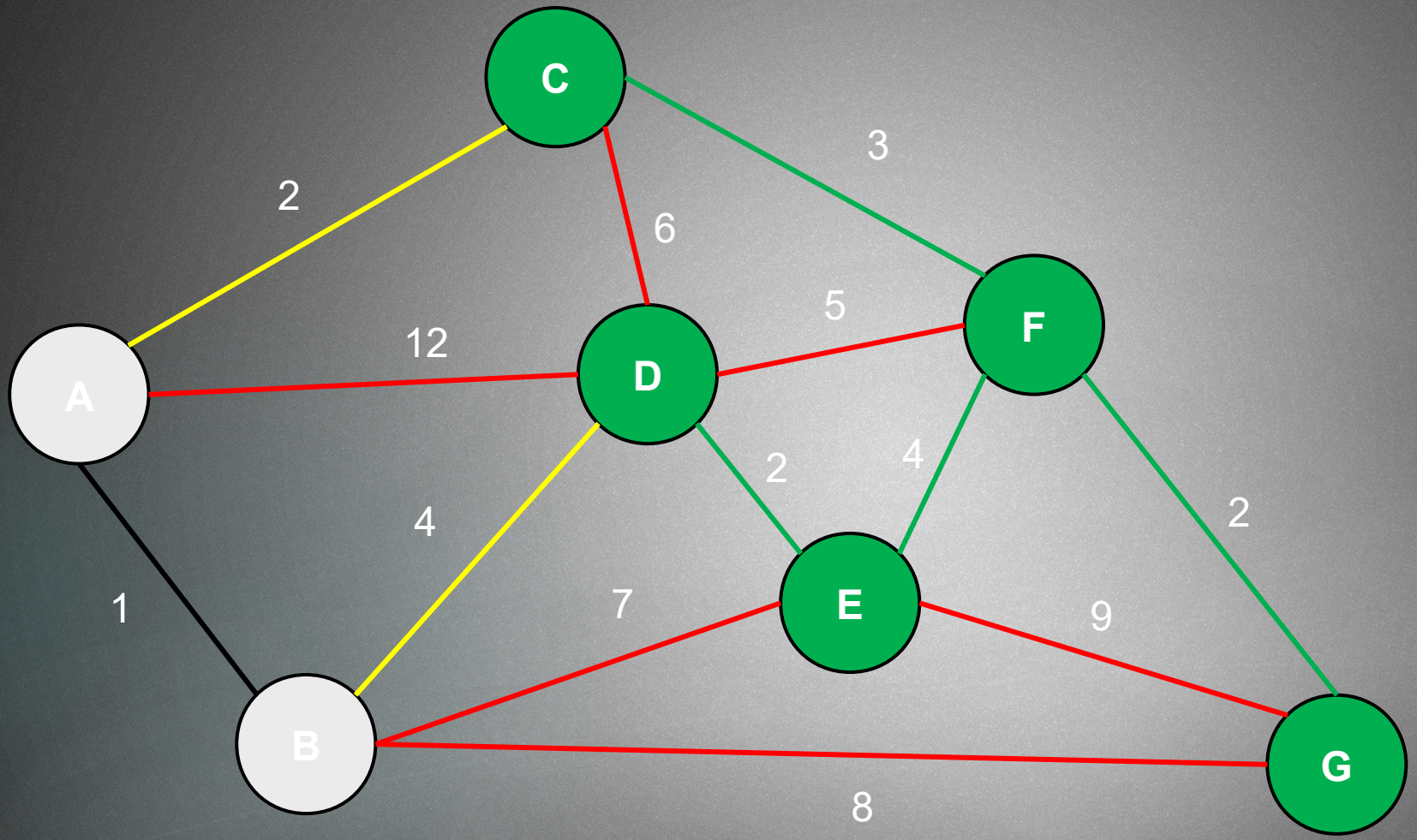
What about the C-A connection? We have considered a path to vertex A so far from vertex D. The cost from vertex D is 12 , the cost from vertex C is 2 SO UPDATE THE HEAP !!!



Heap

D-B 4
C-A 2

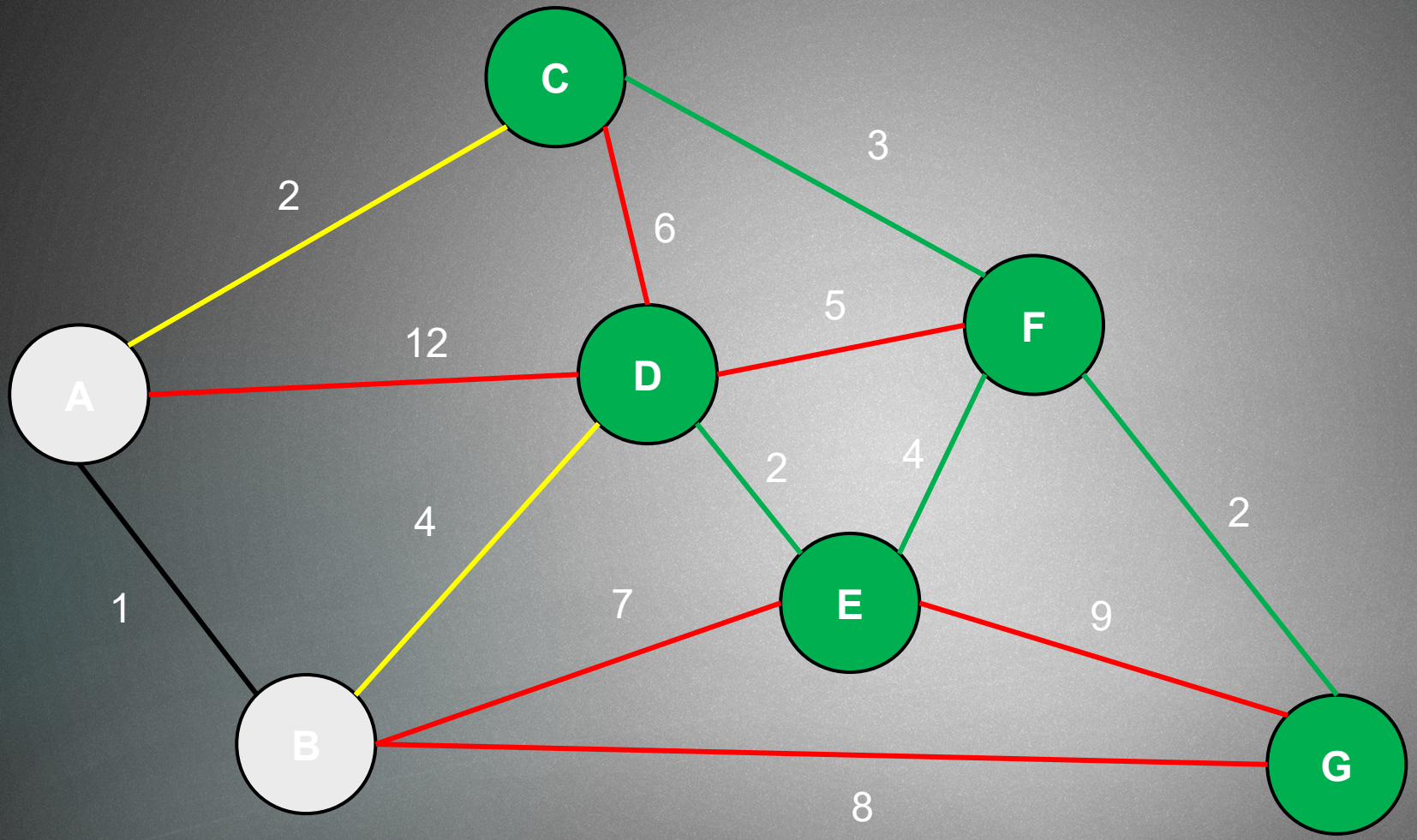
Keep going !!!



Heap

D-B 4
C-A 2

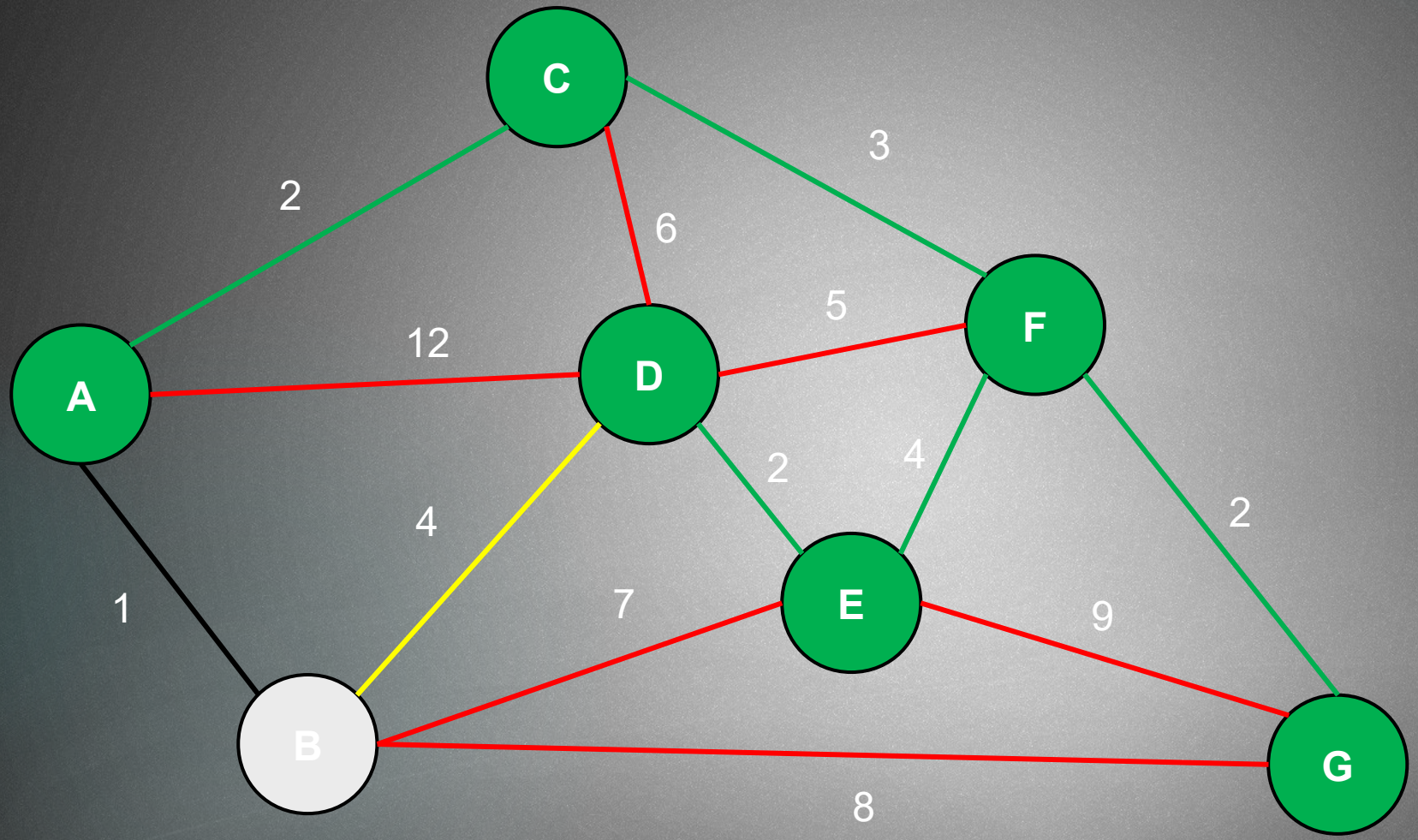
Keep going !!!



Heap

D-B 4
C-A 2

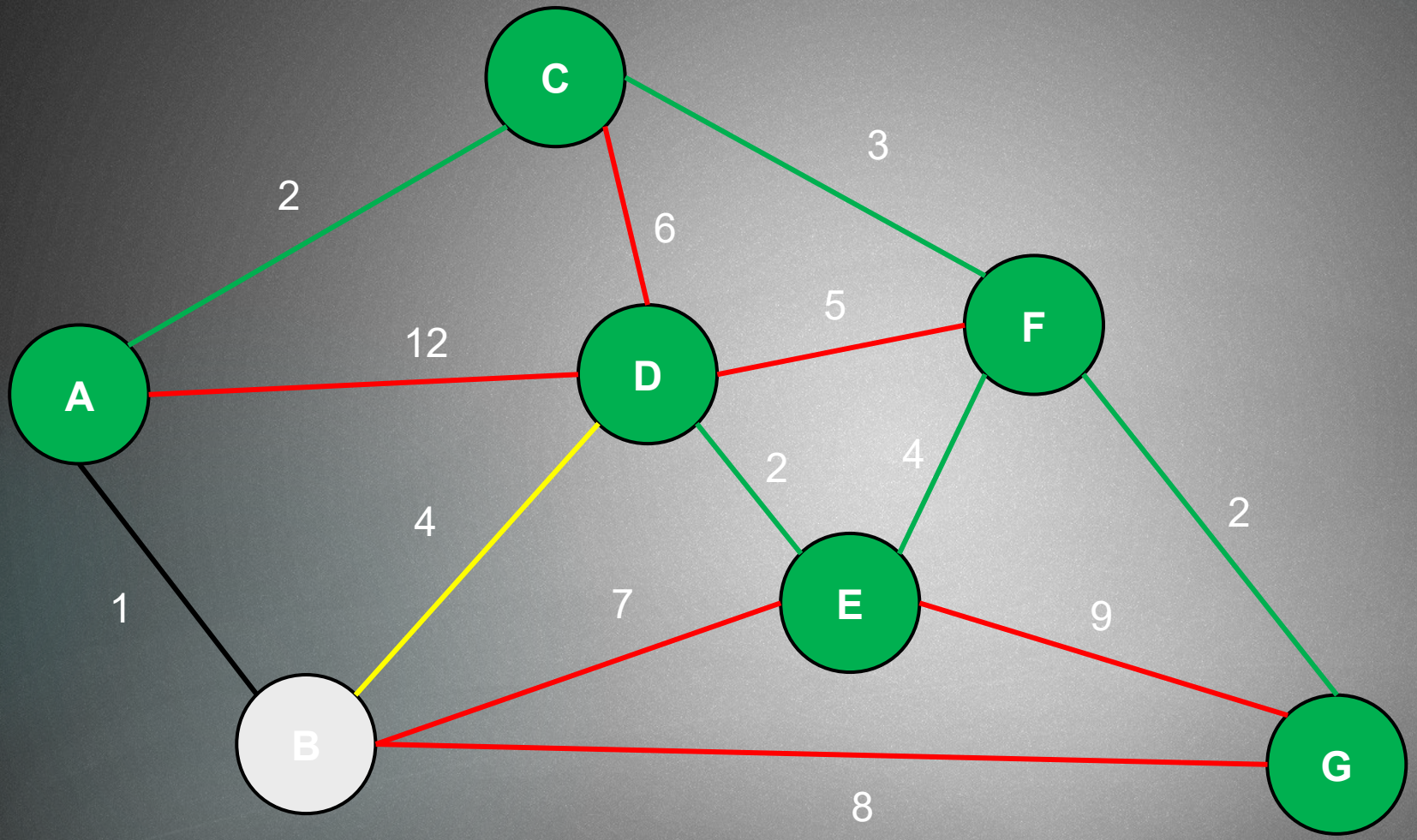
Keep going !!!



Heap

D-B 4
C-A 2

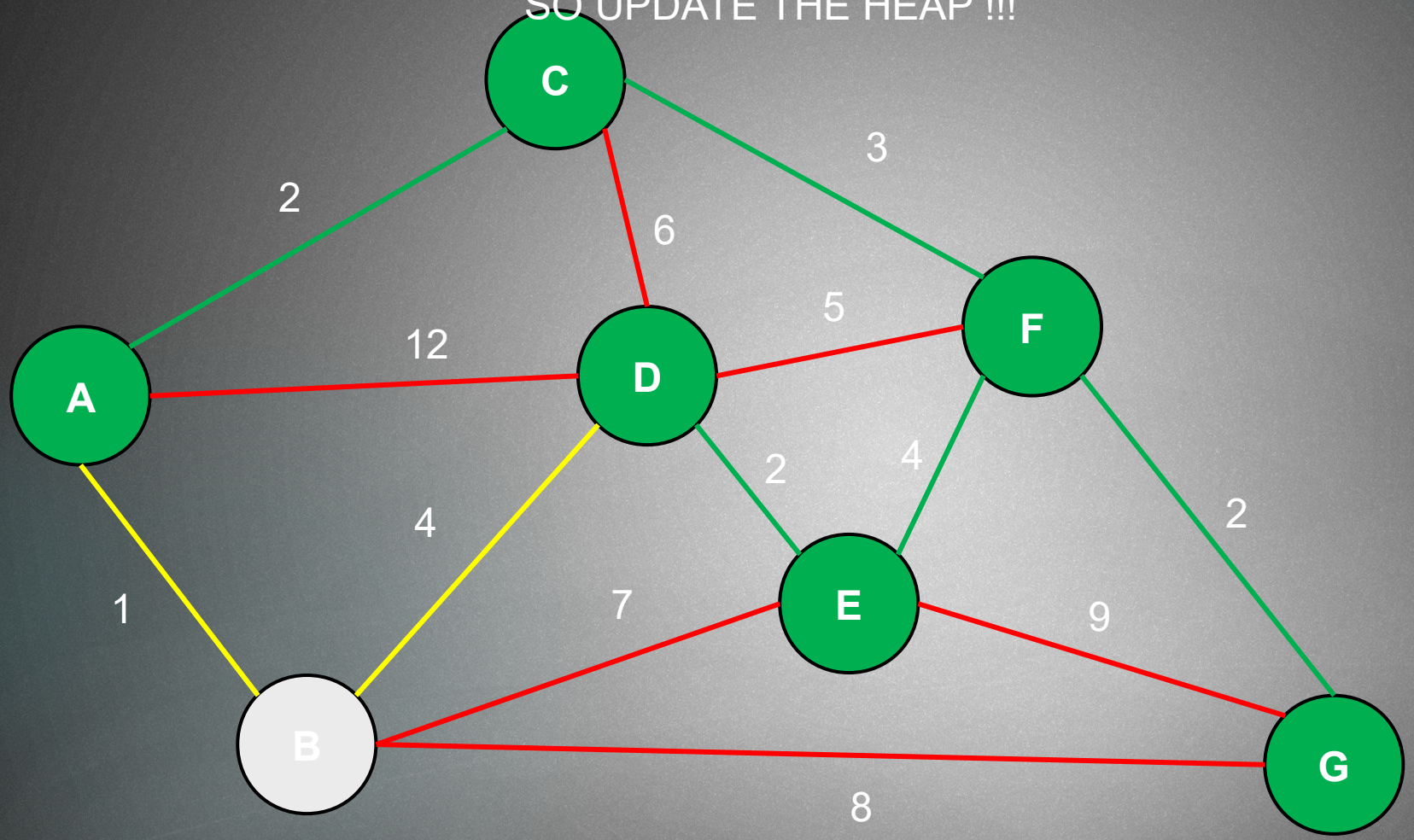
Keep going !!!



Heap

D-B 4

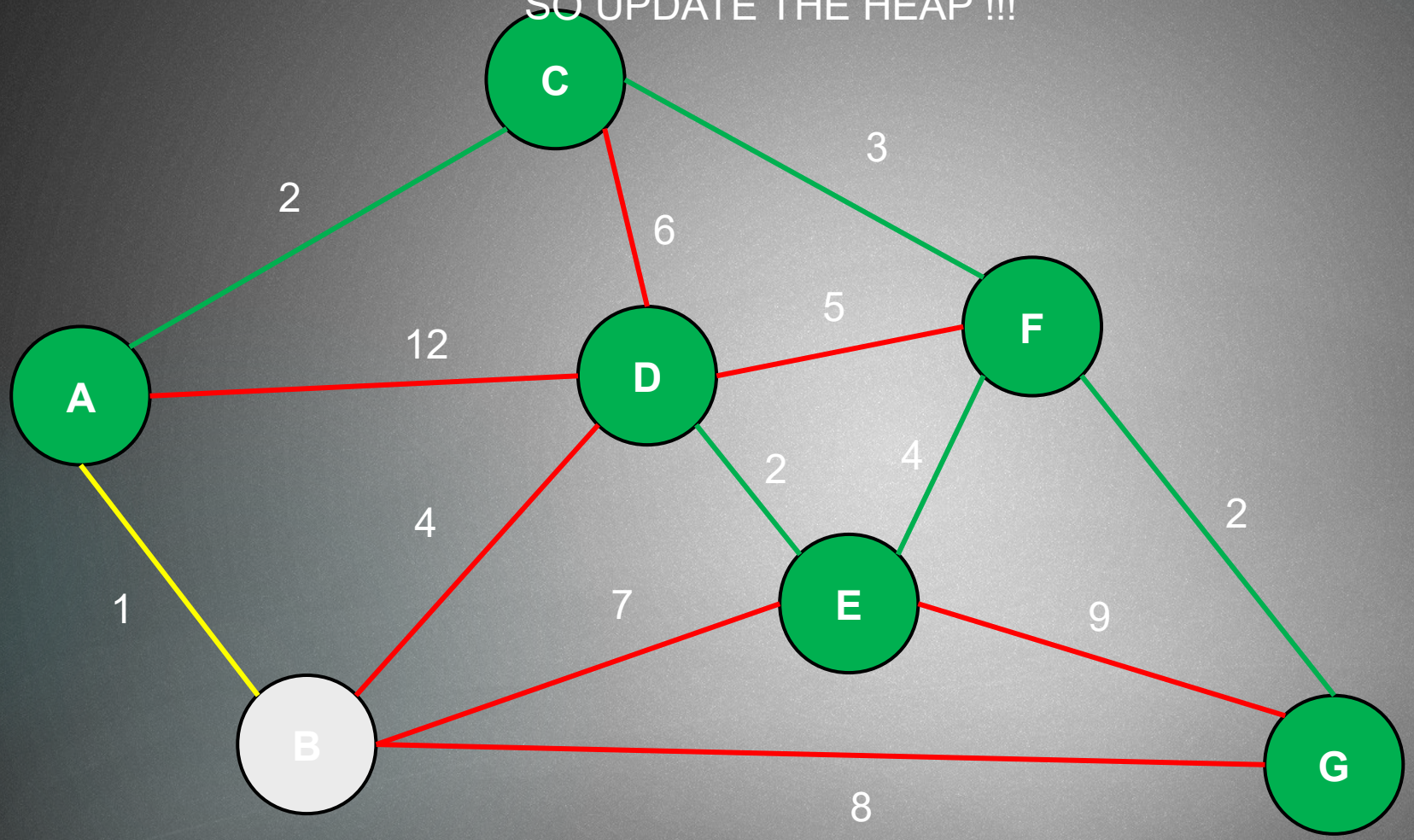
What about the A-B connection? We have considered a path to vertex B so far from vertex D. The cost from vertex D is 4, the cost from vertex A is 1
SO UPDATE THE HEAP !!!



Heap

D-B 4

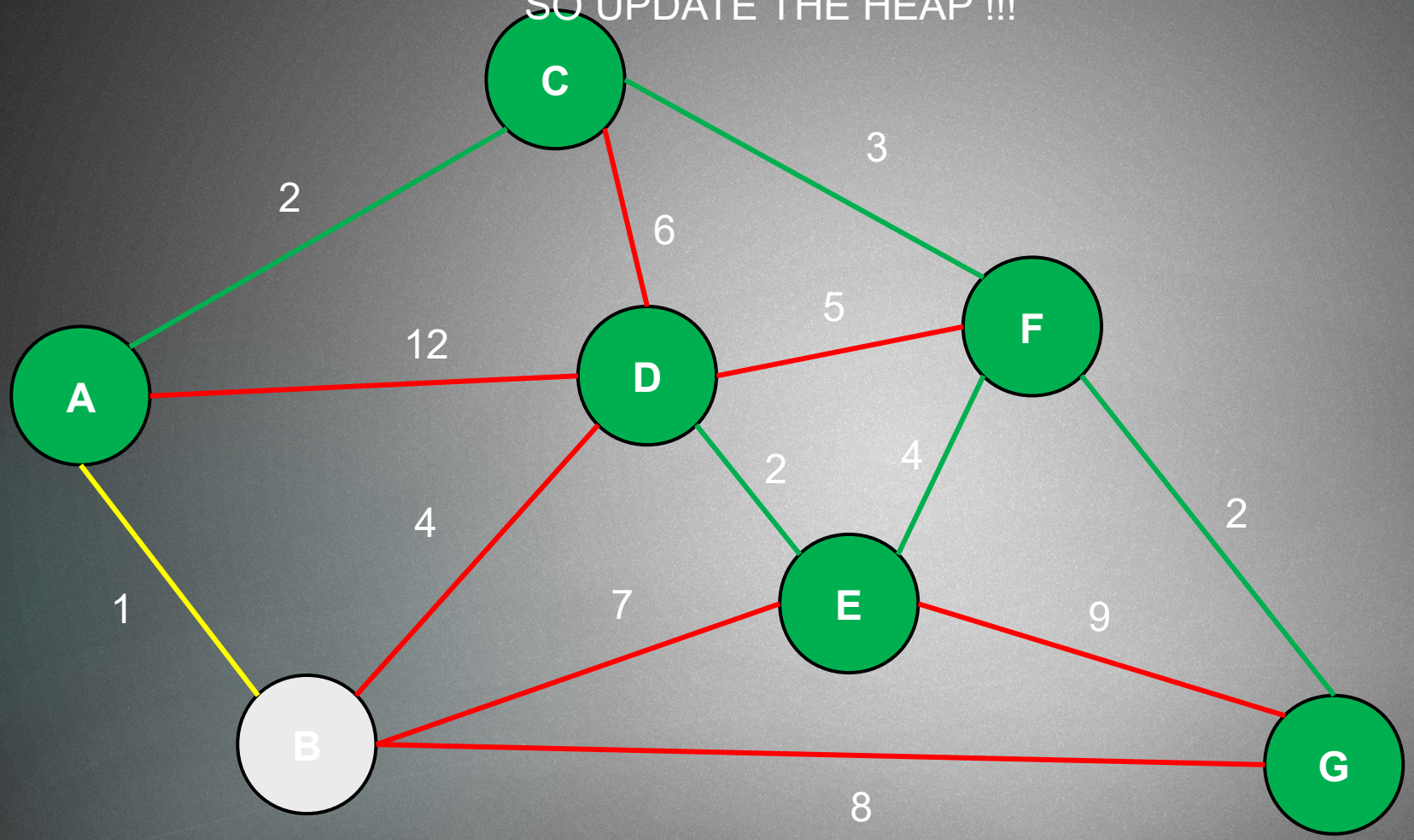
What about the A-B connection? We have considered a path to vertex B so far from vertex D. The cost from vertex D is 4, the cost from vertex A is 1
SO UPDATE THE HEAP !!!



Heap

D-B 4

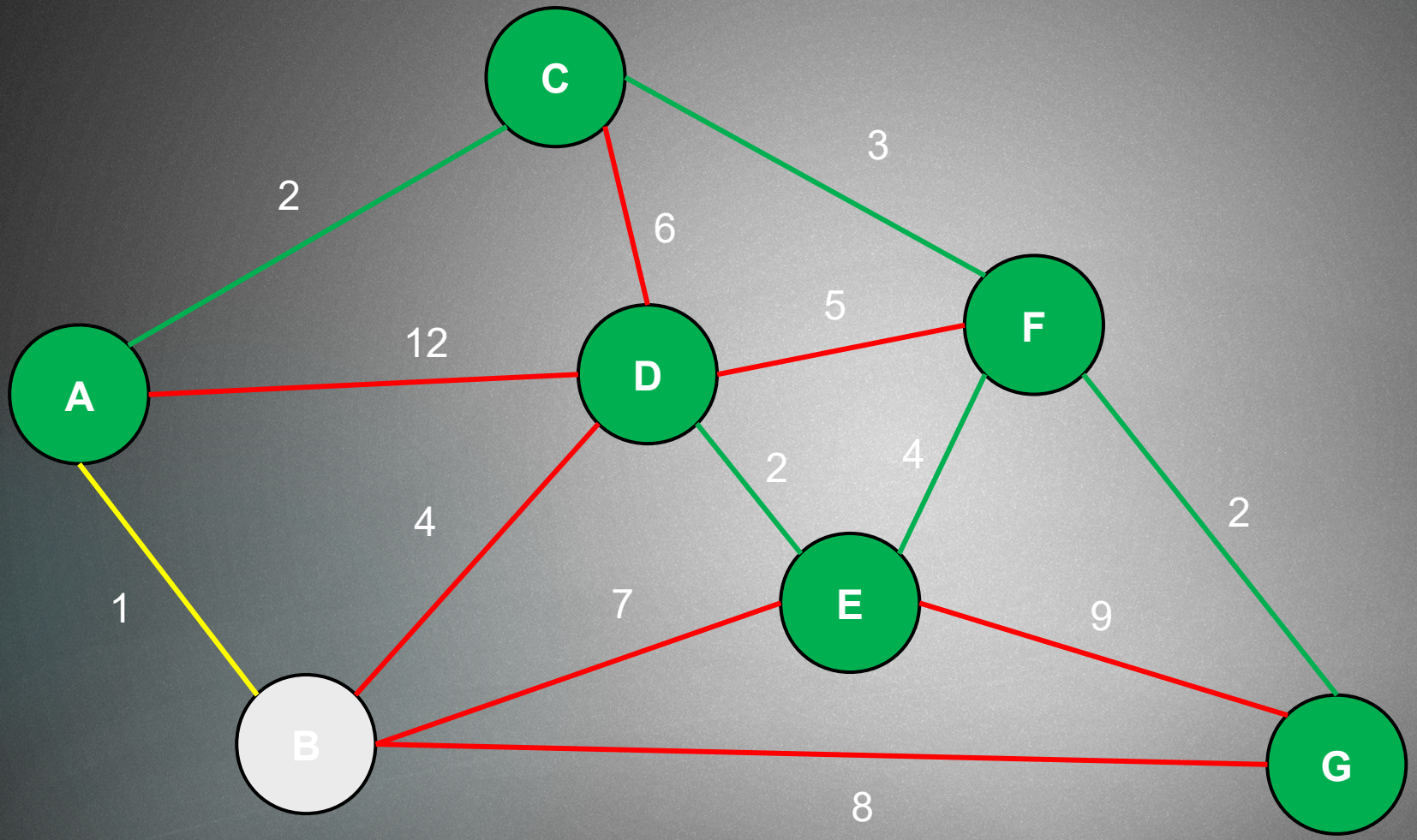
What about the A-B connection? We have considered a path to vertex B so far from vertex D. The cost from vertex D is 4, the cost from vertex A is 1
SO UPDATE THE HEAP !!!



Heap

A-B 1

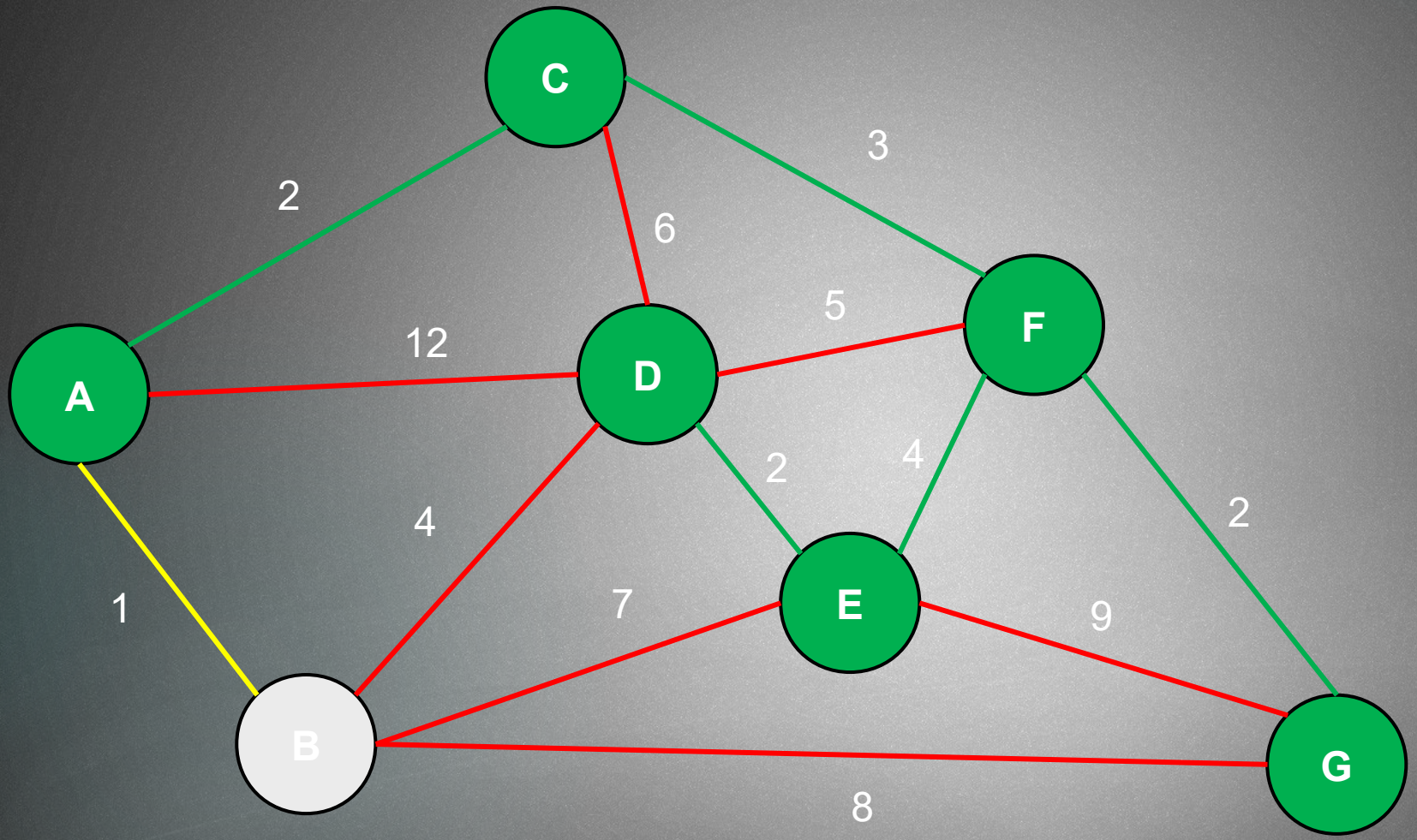
Keep going !!!



Heap

A-B 1

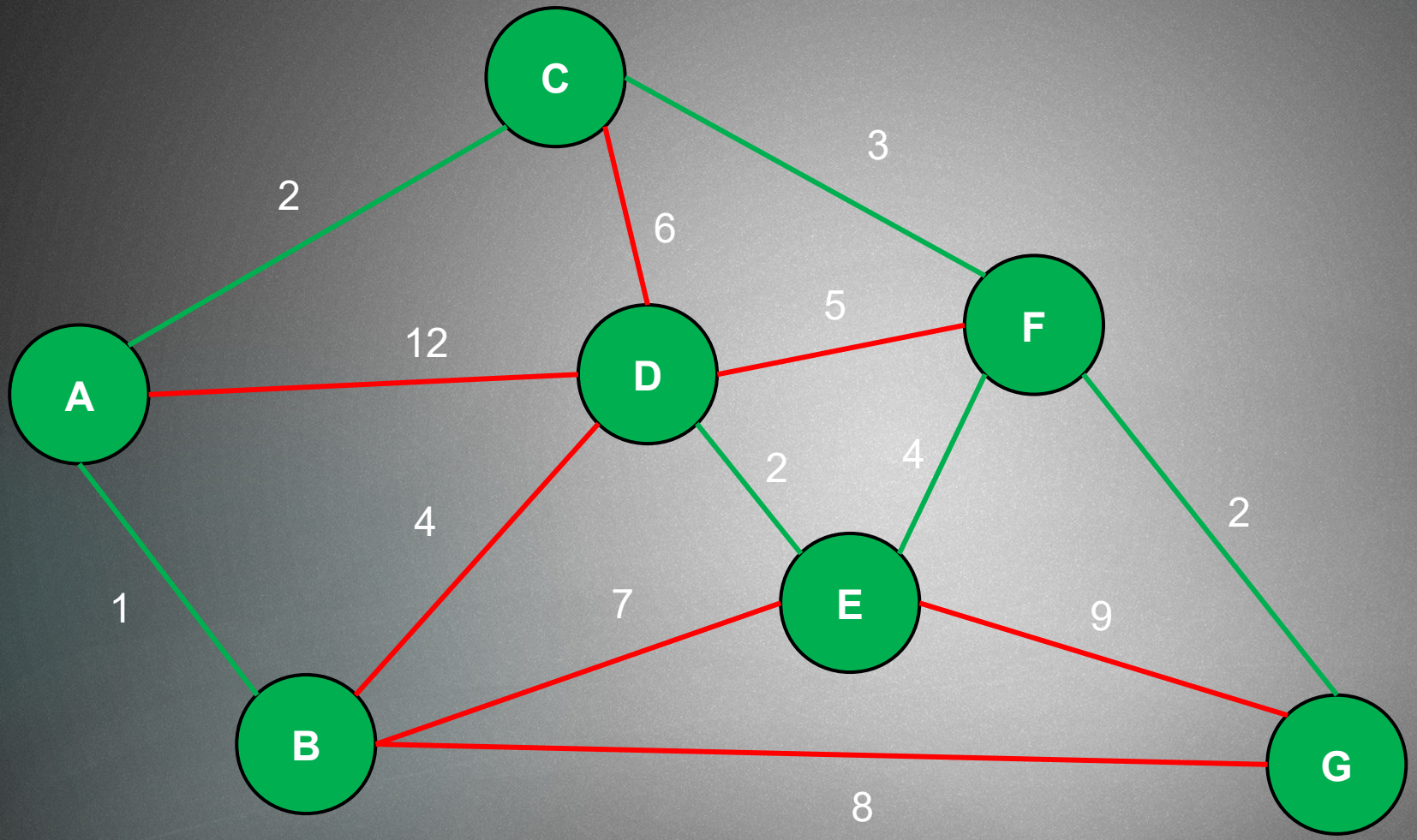
Keep going !!!



Heap

A-B 1

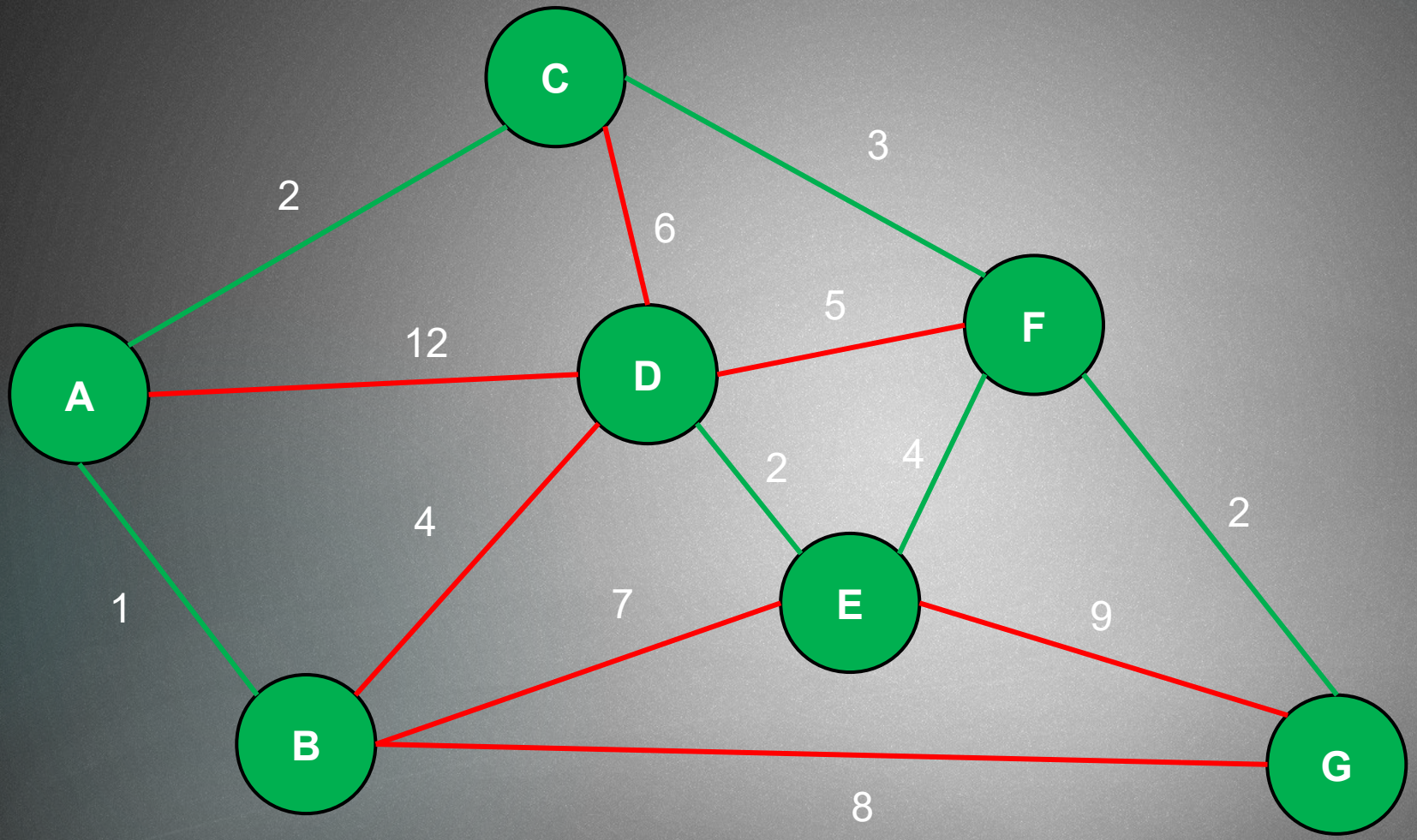
Keep going !!!



Heap

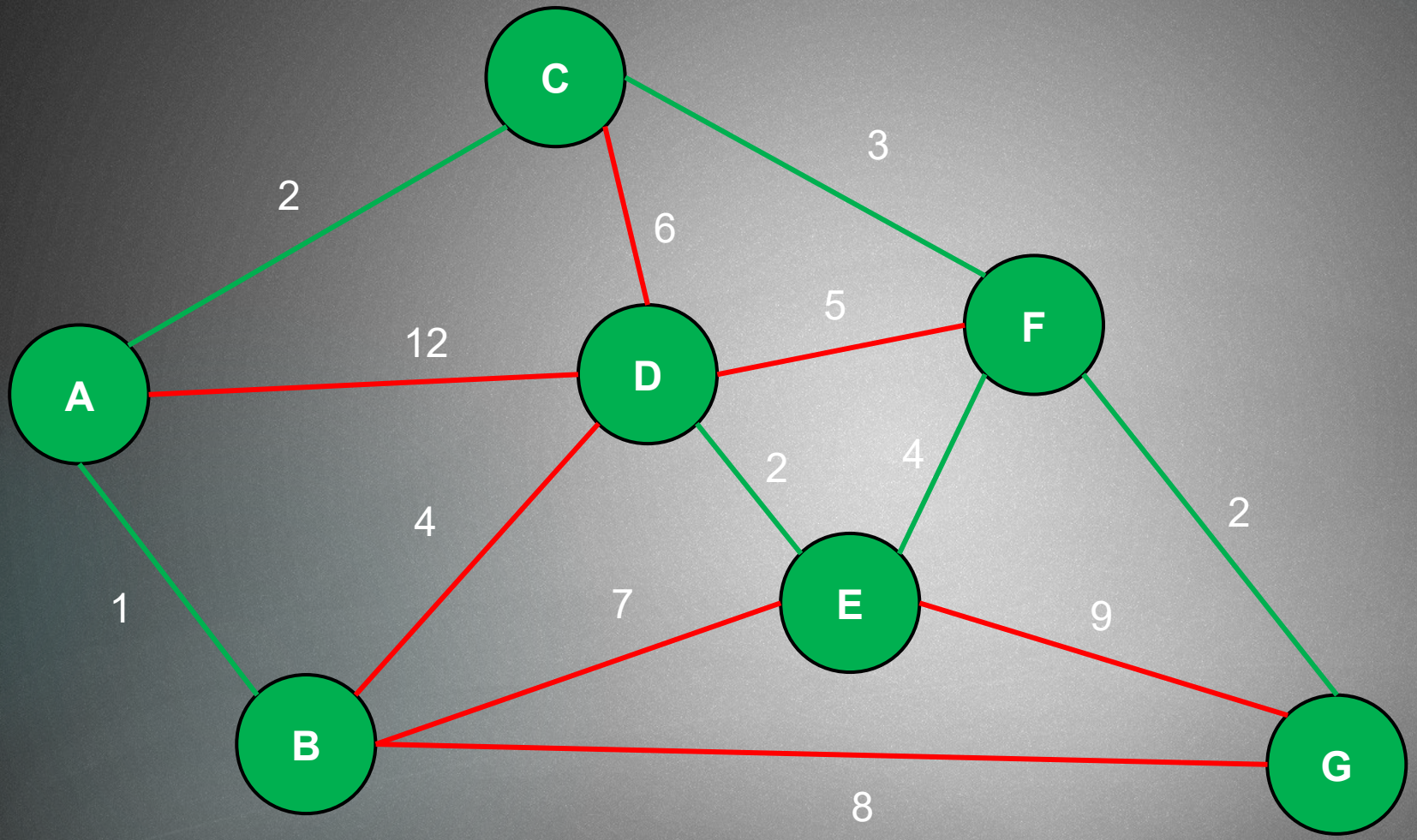
A-B 1

Keep going !!!



Heap

We have managed to include all the vertexes in the MST
So the algorithm terminates → overall cost = 14



Heap

We have managed to include all the vertexes in the MST
So the algorithm terminates → overall cost = 14