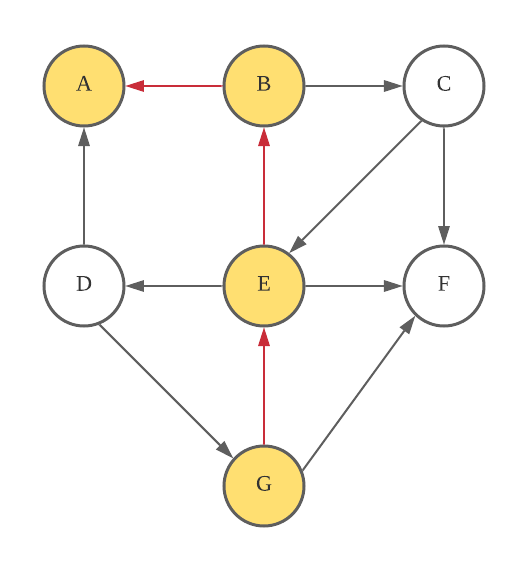
# Assignment 6

## Question 1

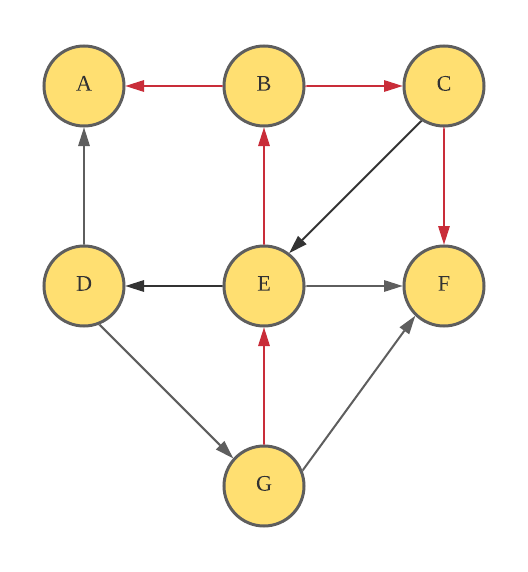
G->E->B->A

Backtrack to B



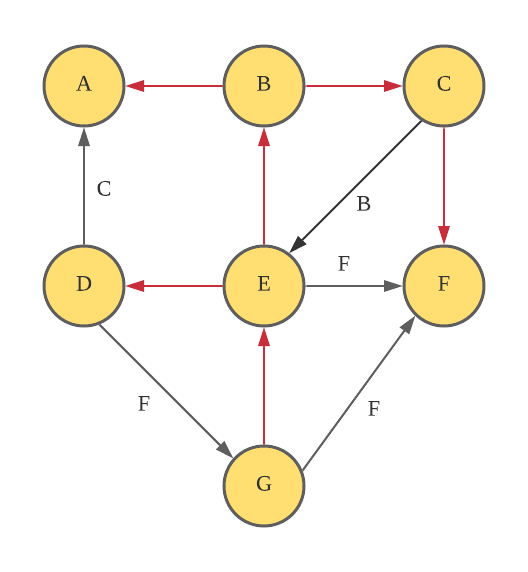
B->C->F

Backtrack to C -> B -> E



E->D

Finished



Tree edge: all edge represented by red color are tree edges

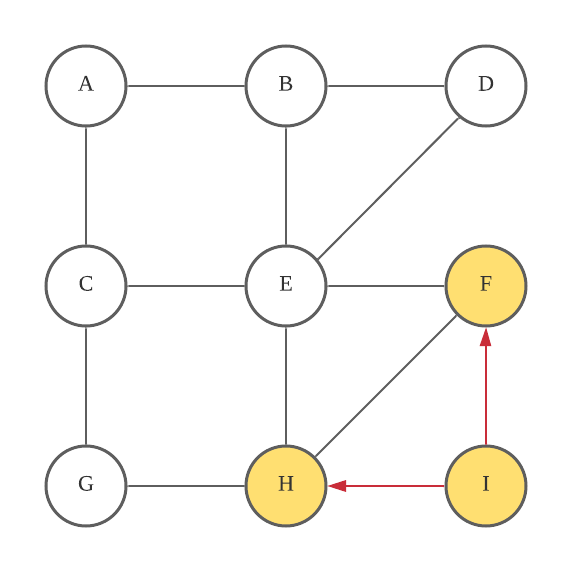
Back edge: a back edge connects a vertex to its ancestor in the DFS tree, they are labeled B.

Forward edge: a forward edge connects a vertex to its descendant in the DFS tree, they are labeled F.

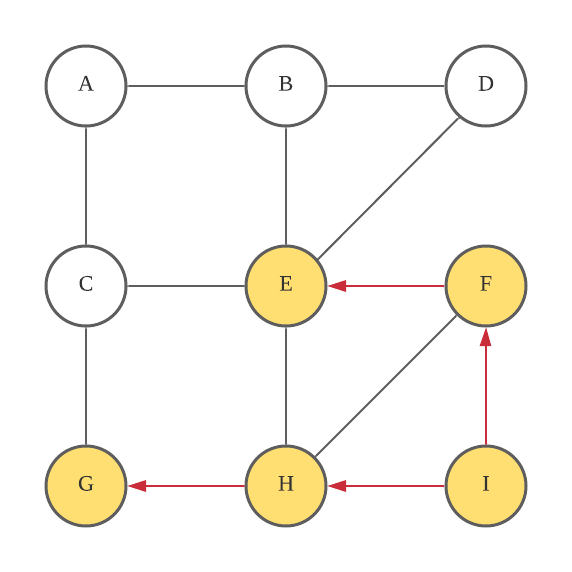
Cross edge: a cross edge connects a vertex to a vertex that is neither its ancestor nor its descendant, they are labeled C.

## Question 2

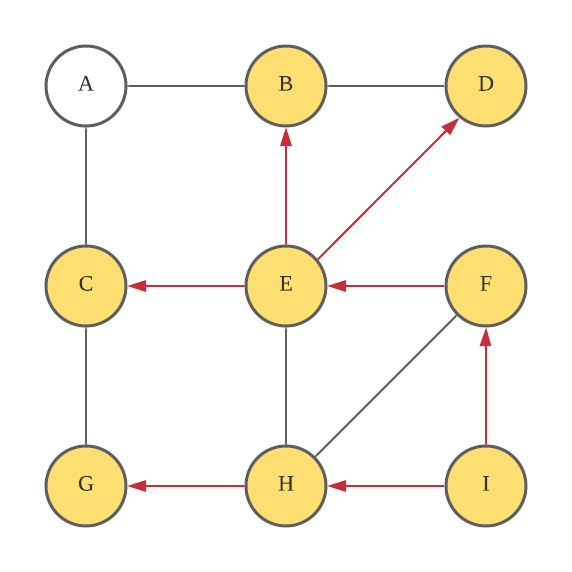
Explore vertices that are one-edge away from I



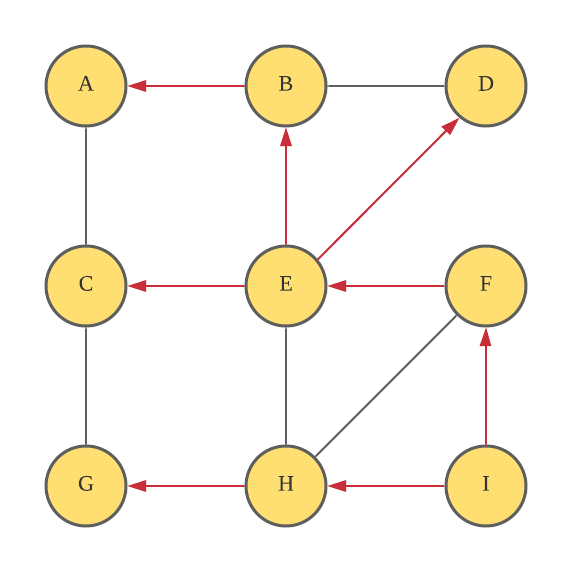
Explore vertices that are two-edge away from I



Explore vertices that are three-edge away from I

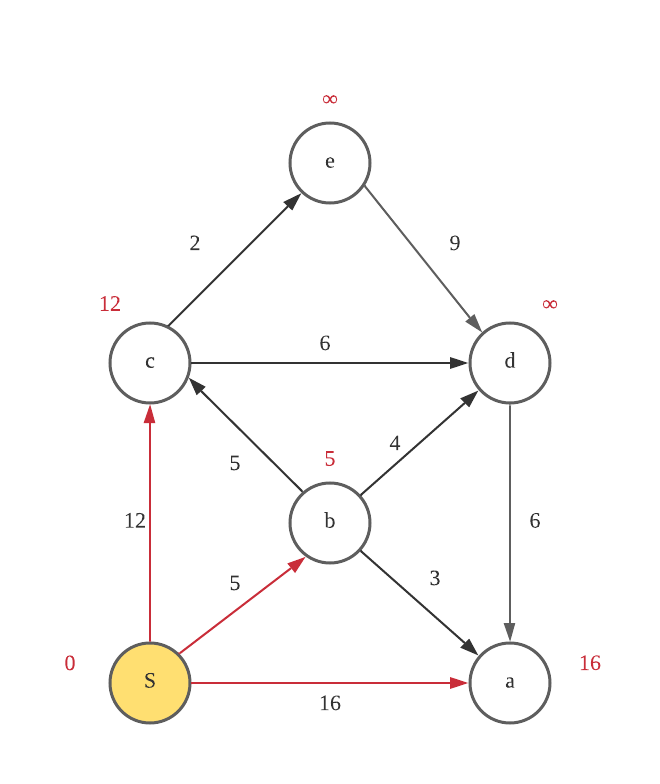


Explore vertices that are four-edge away from I

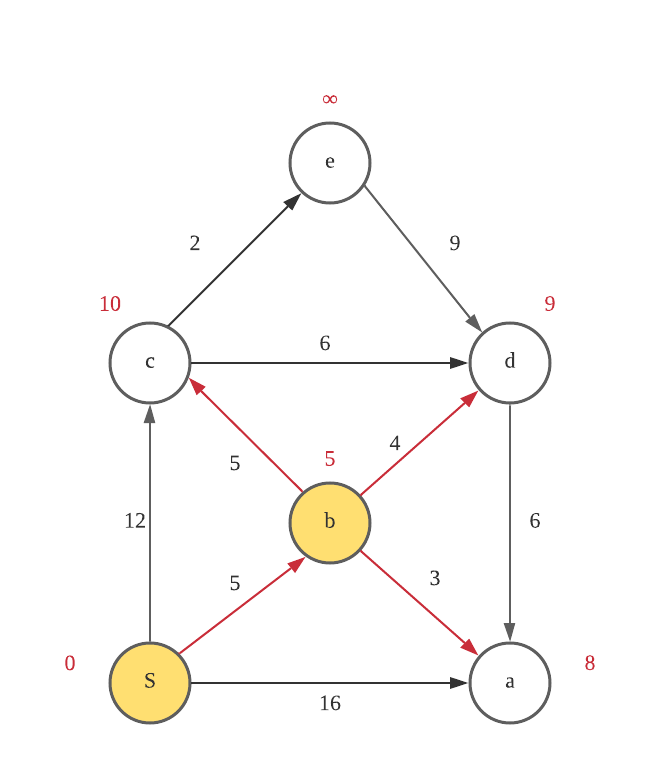


## Question 3

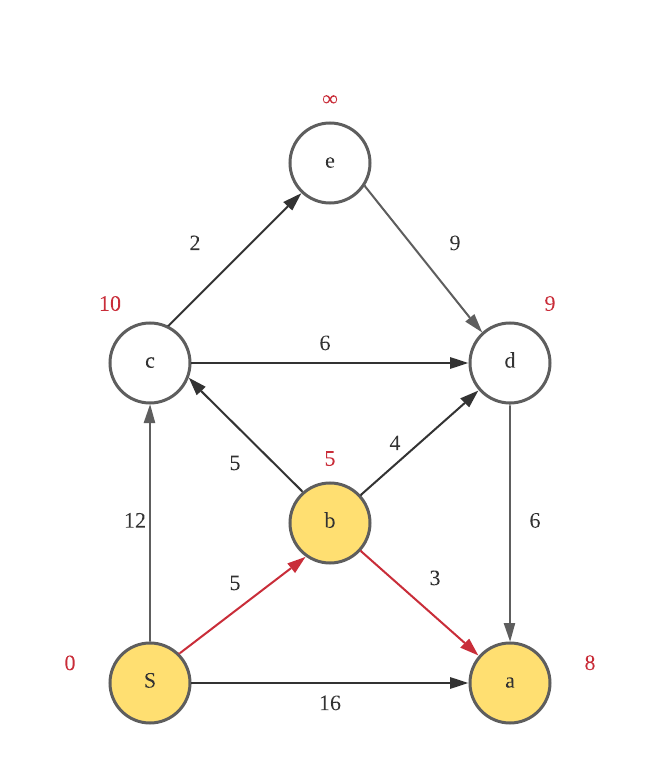
Iteration 1



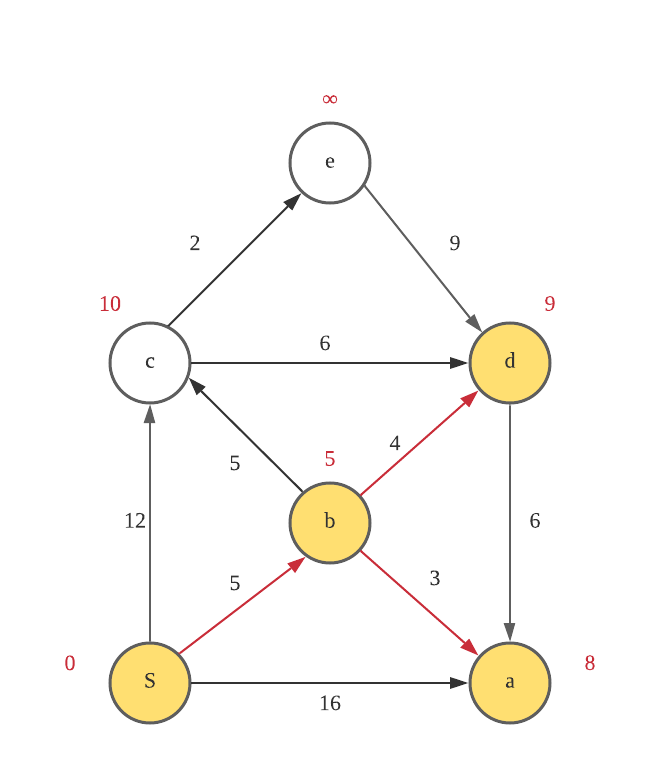
Iteration 2



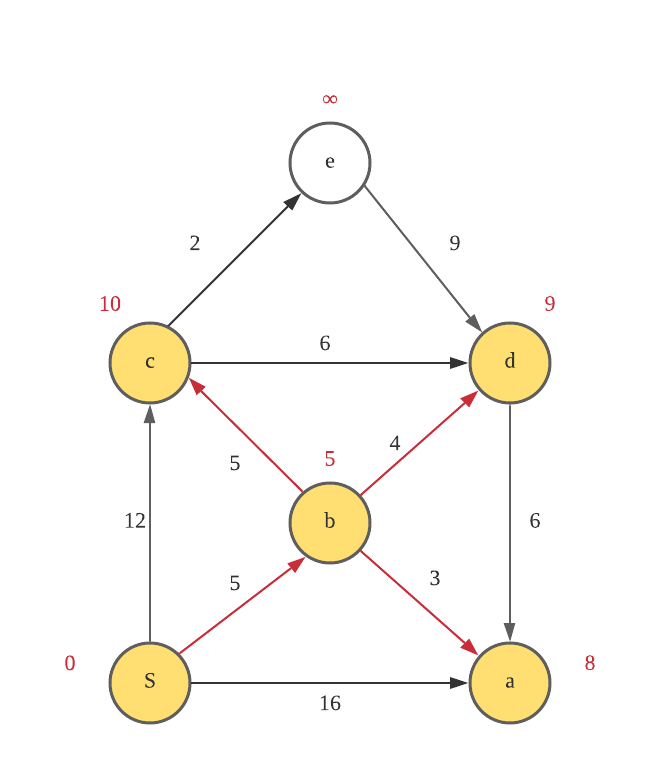
Iteration 3



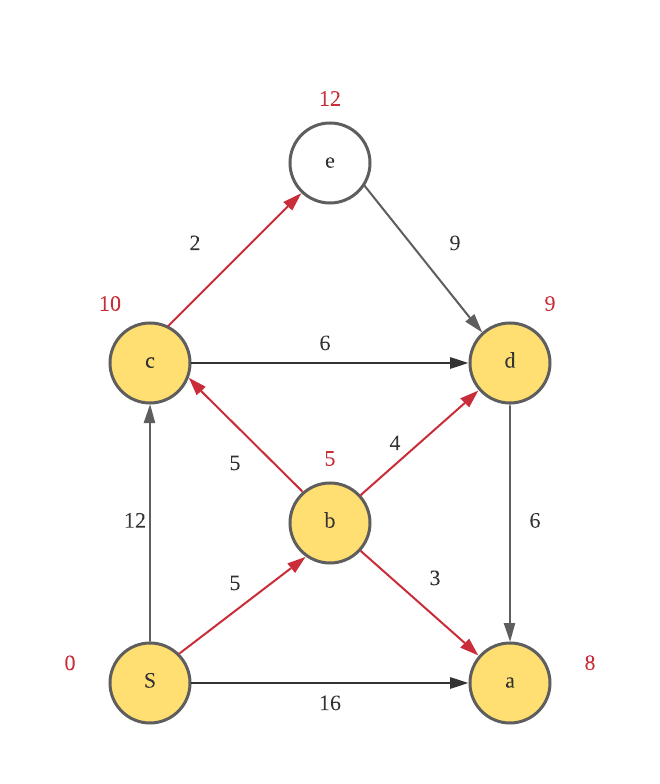
Iteration 4



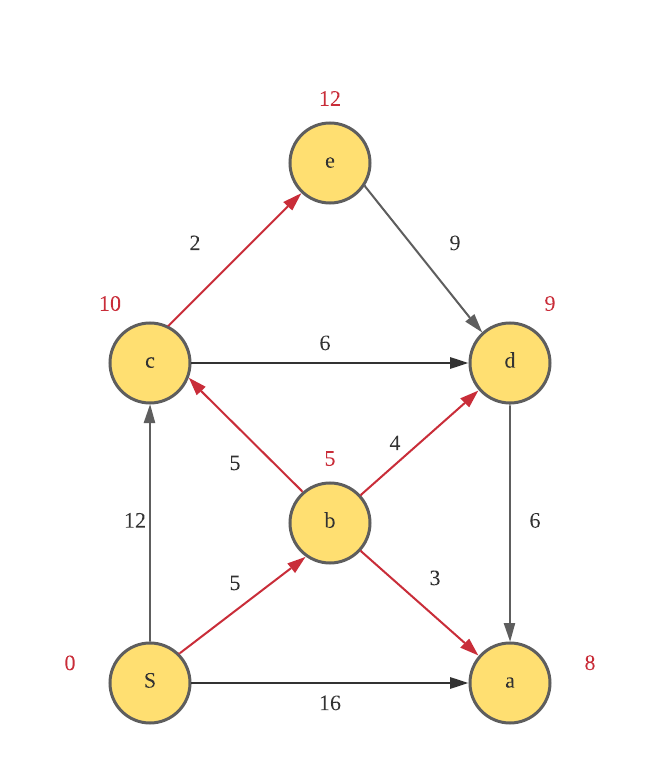
Iteration 5



Iteration 6



Iteration 7



A: S->b->a distance: 8

B: S->b distance: 5

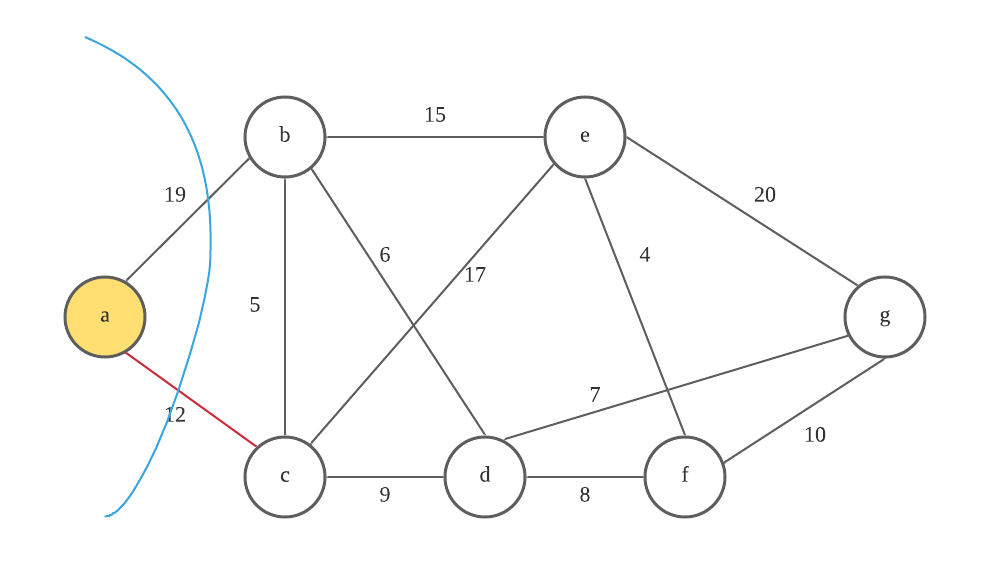
C: S->b->c distance: 10

D: S->b->d distance: 9

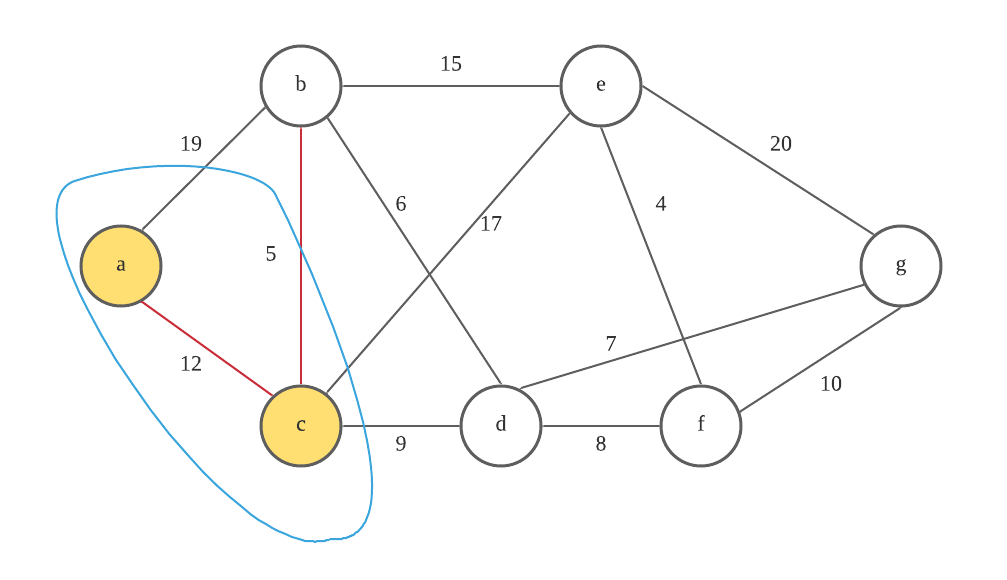
E: S->b->c->e distance: 12

## Question 4

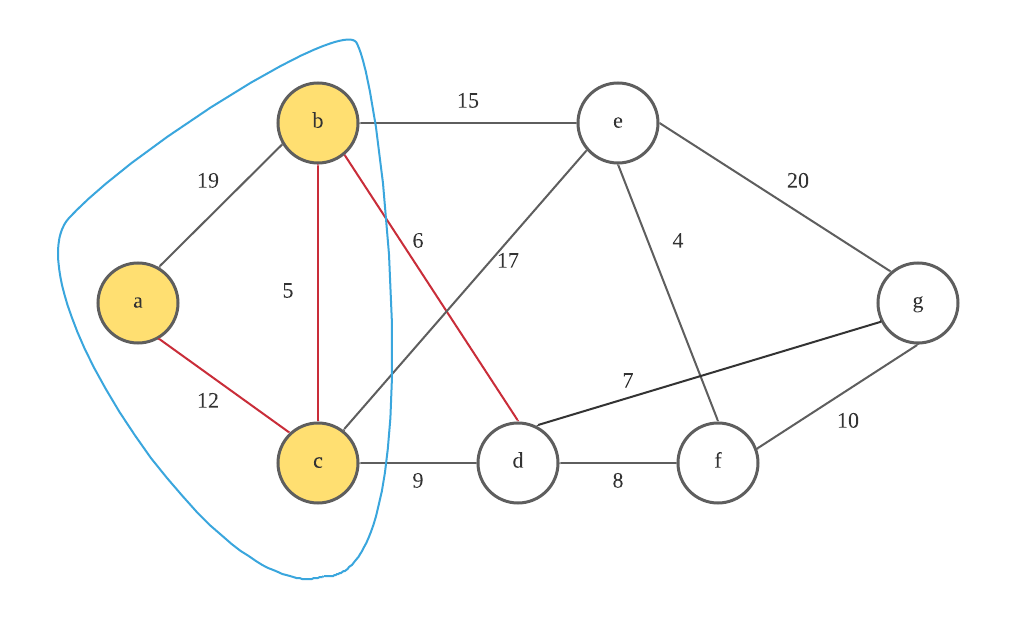
Initial tree, begin at a, (a, c) is minimum-weight edge



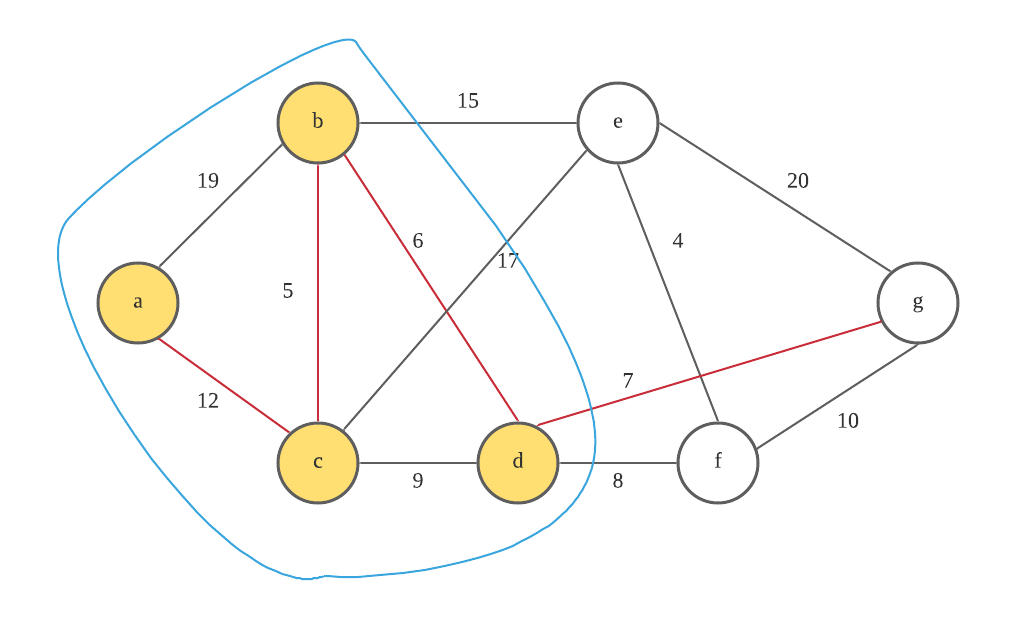
(c, b) is minimum-weight edge



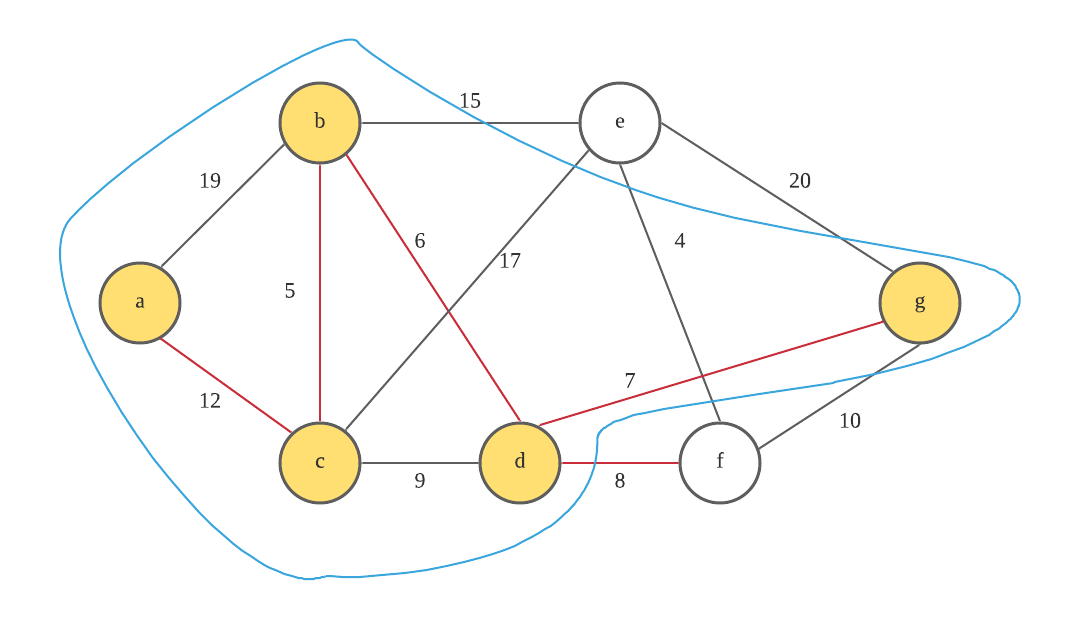
(b, d) is minimum-weight edge



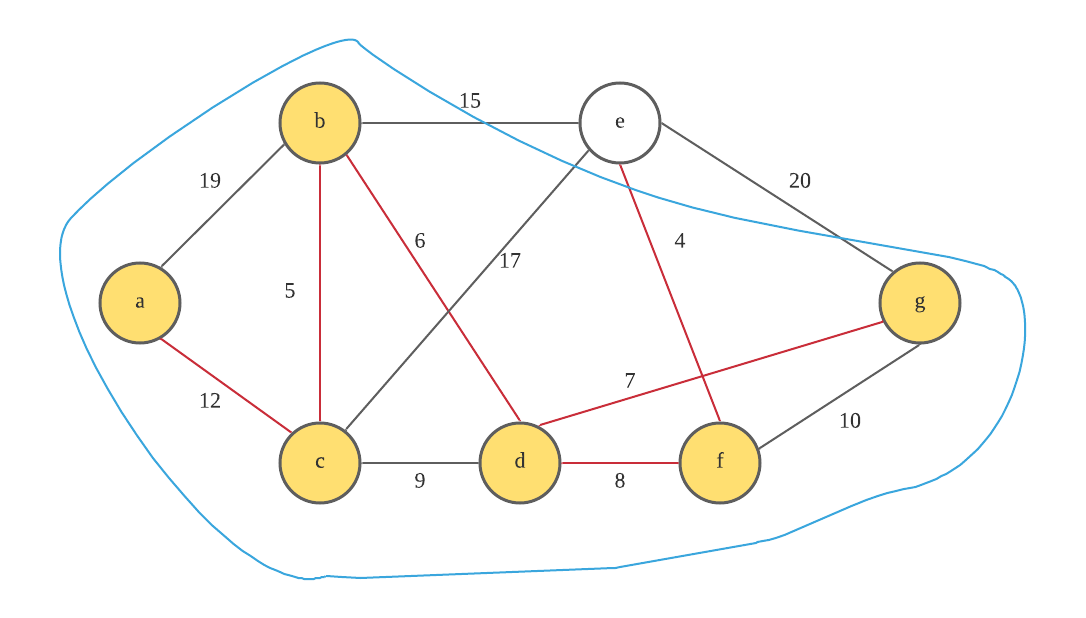
(d, g) is minimum-weight edge



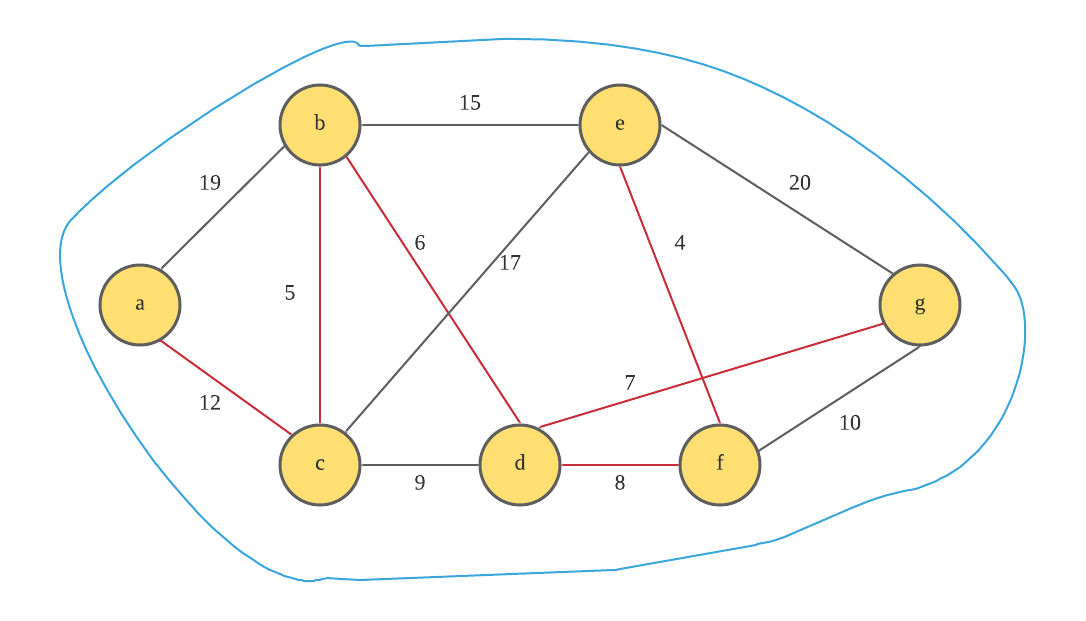
(d, f) is minimum-weight edge



(f, e) is minimum-weight edge



Finished



Sequence of nodes: a->c->b->d->g->f->e