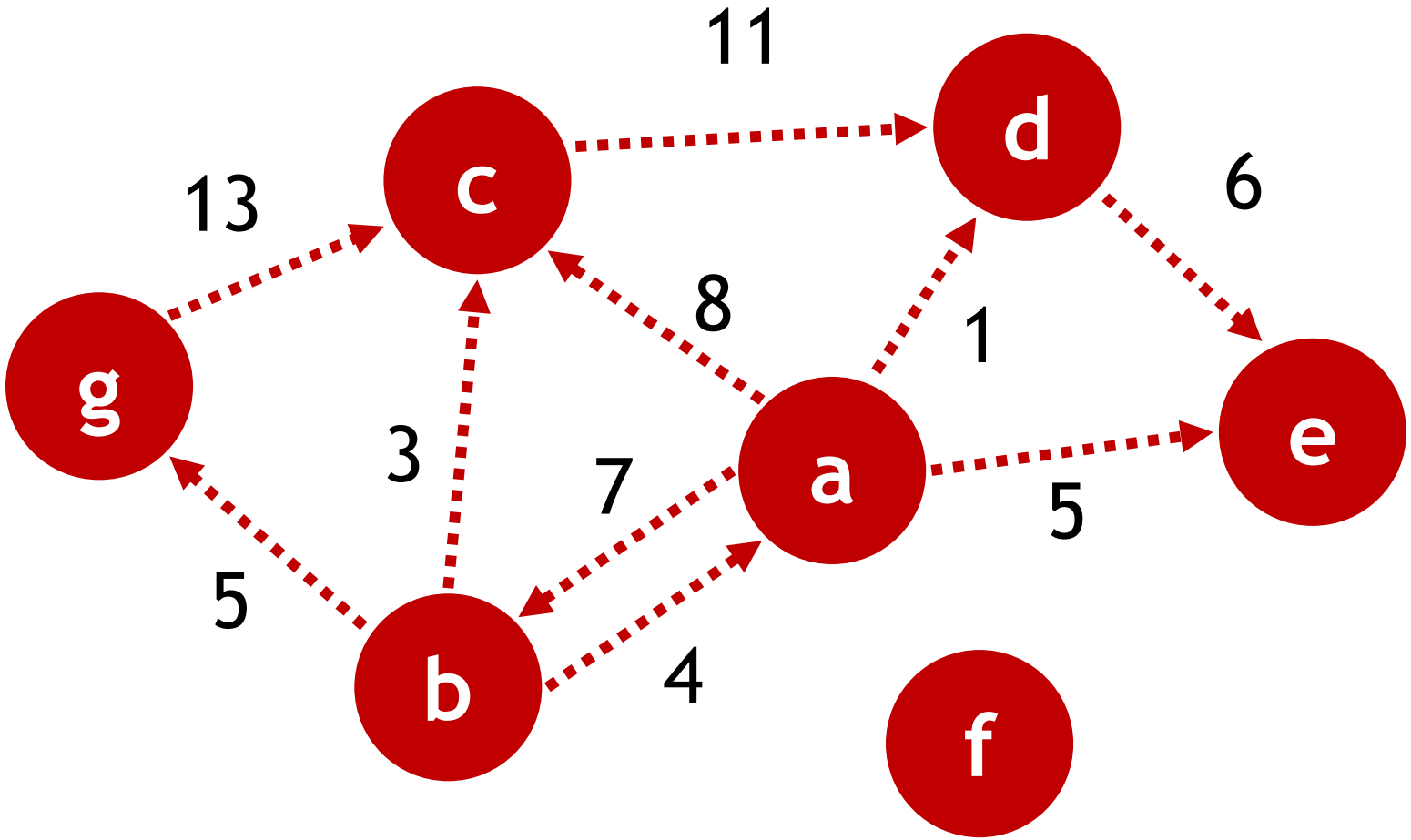
Graf

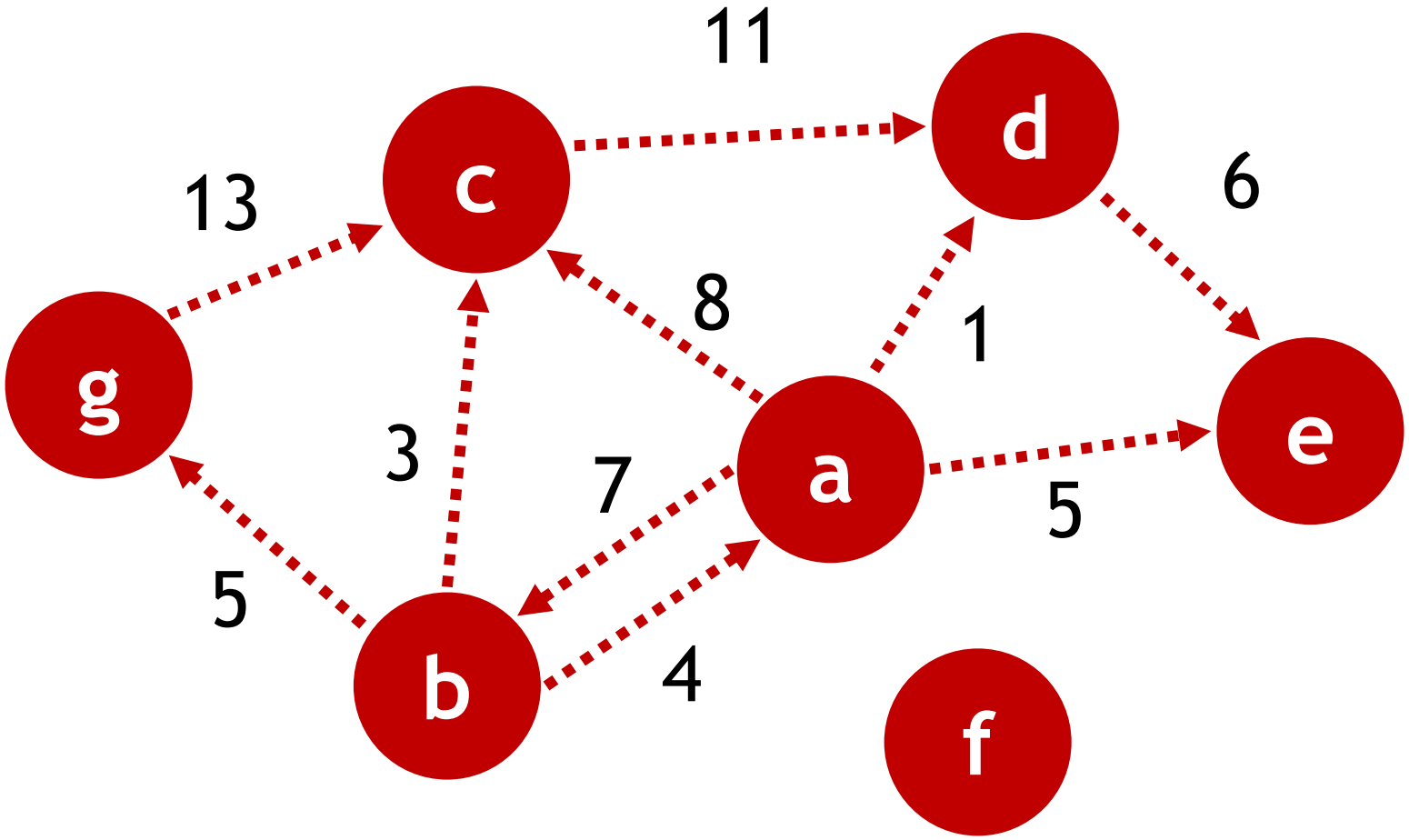


Programovacie techniky | Pavol Marák

OBSAH

- Reprezentácie grafu
- Zoznam susedov a jeho implementácia
- Prechod do hĺbky/šírky

Reprezentácia grafu

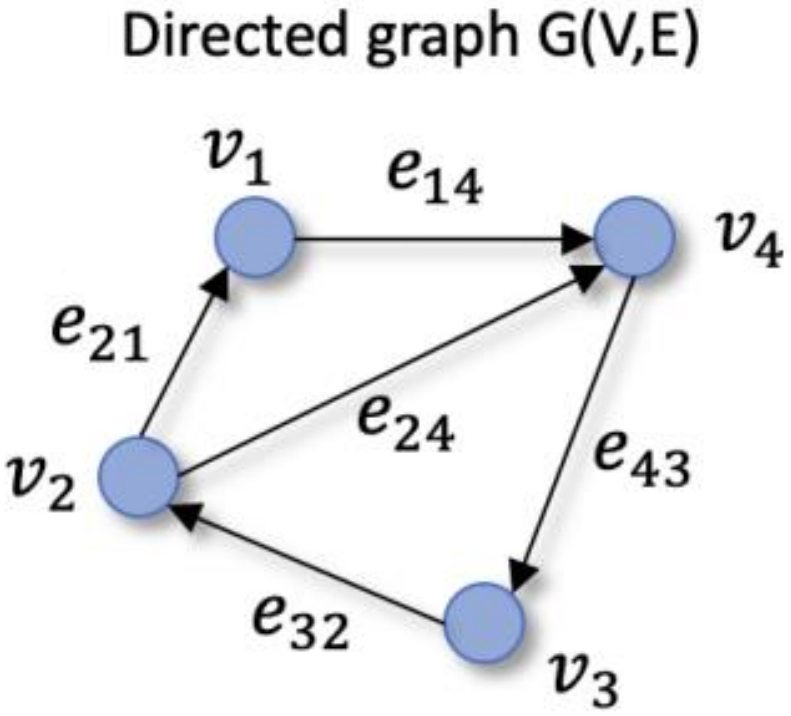


Reprezentácia grafu

Najznámejšie reprezentácie:

- Matica incidencie (incidence matrix)
- Matica susedov (adjacency matrix)
- **Zoznam susedov** (adjacency list)

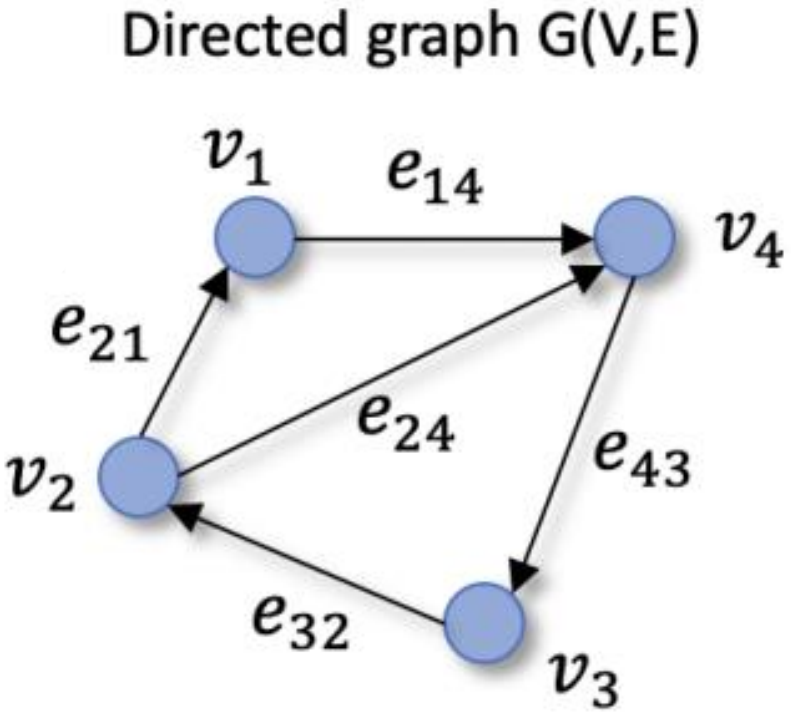
Matica incidencie



	e_{14}	e_{21}	e_{24}	e_{32}	e_{43}
v_1	1	-1	0	0	0
v_2	0	1	1	-1	0
v_3	0	0	0	1	-1
v_4	-1	0	-1	0	1

Zdroj: XU, M. Understanding graph embedding methods and their applications. arXiv:2012.08019v1 [cs.LG] 15 Dec 2020.

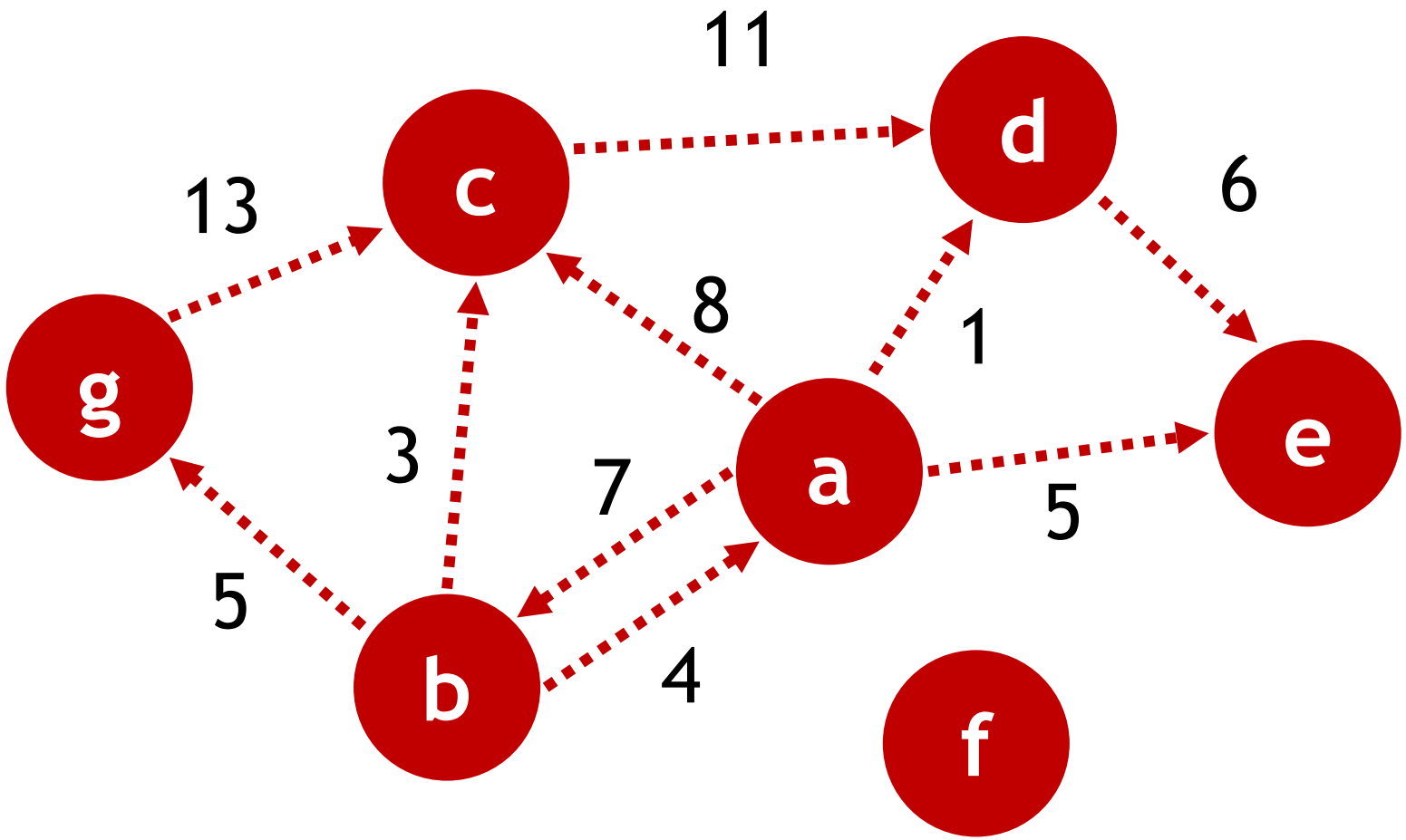
Matica susedov



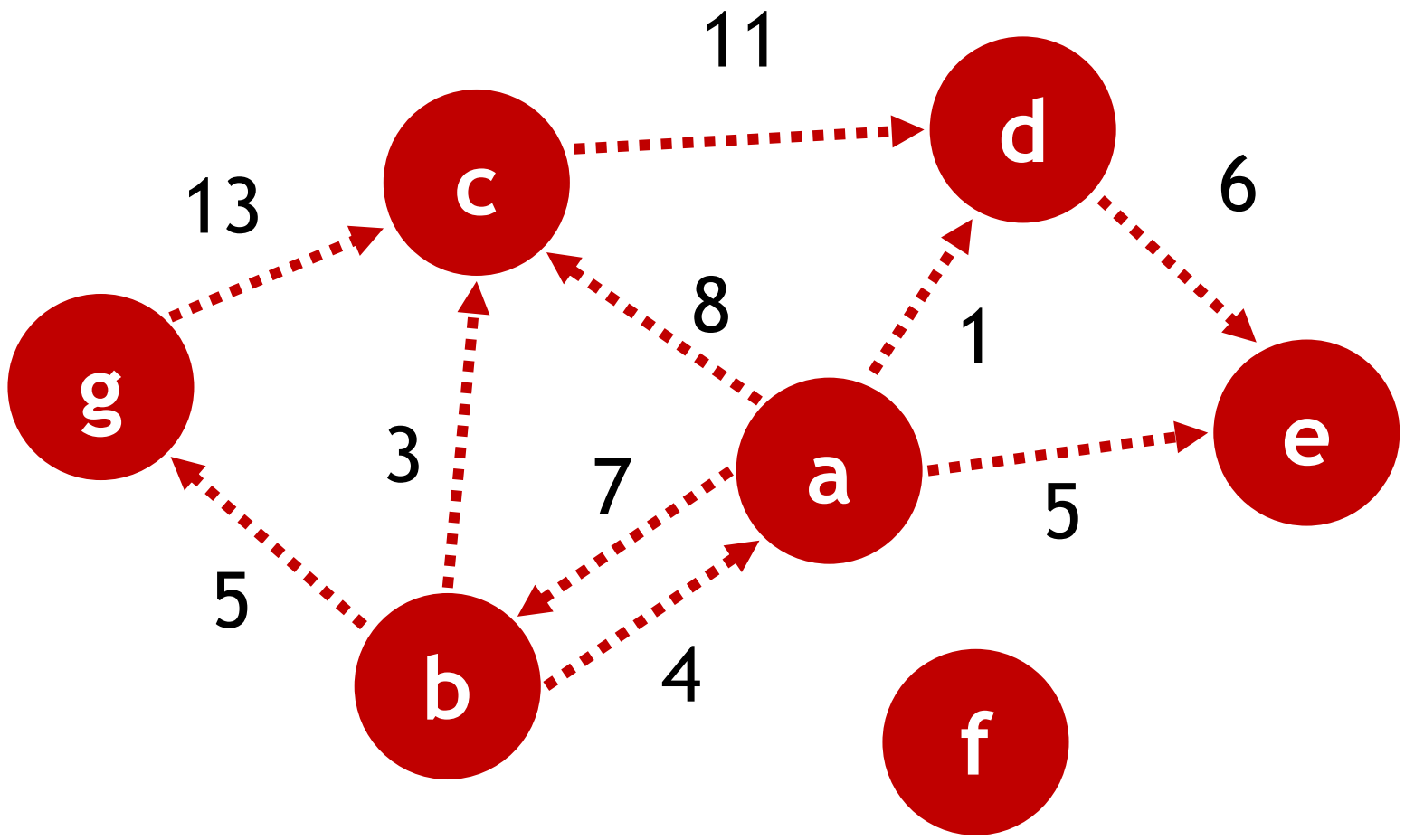
	v_1	v_2	v_3	v_4
v_1	0	0	0	1
v_2	1	0	0	1
v_3	0	1	0	0
v_4	0	0	1	0

Zdroj: XU, M. Understanding graph embedding methods and their applications. arXiv:2012.08019v1 [cs.LG] 15 Dec 2020.

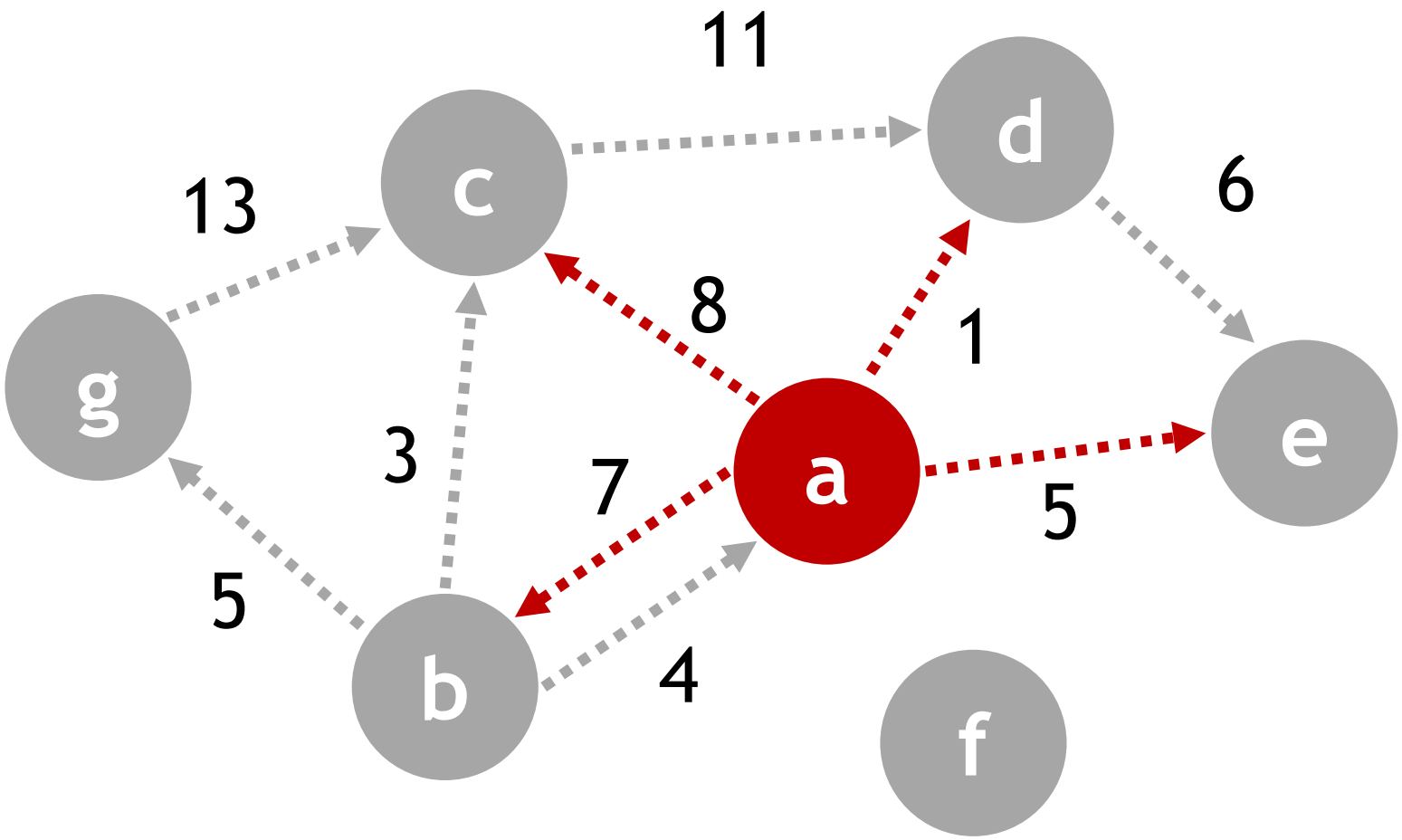
Zoznam susedov


11
c
d
13
6
8
1
g
e
3
7
a
5
5
b
4
f

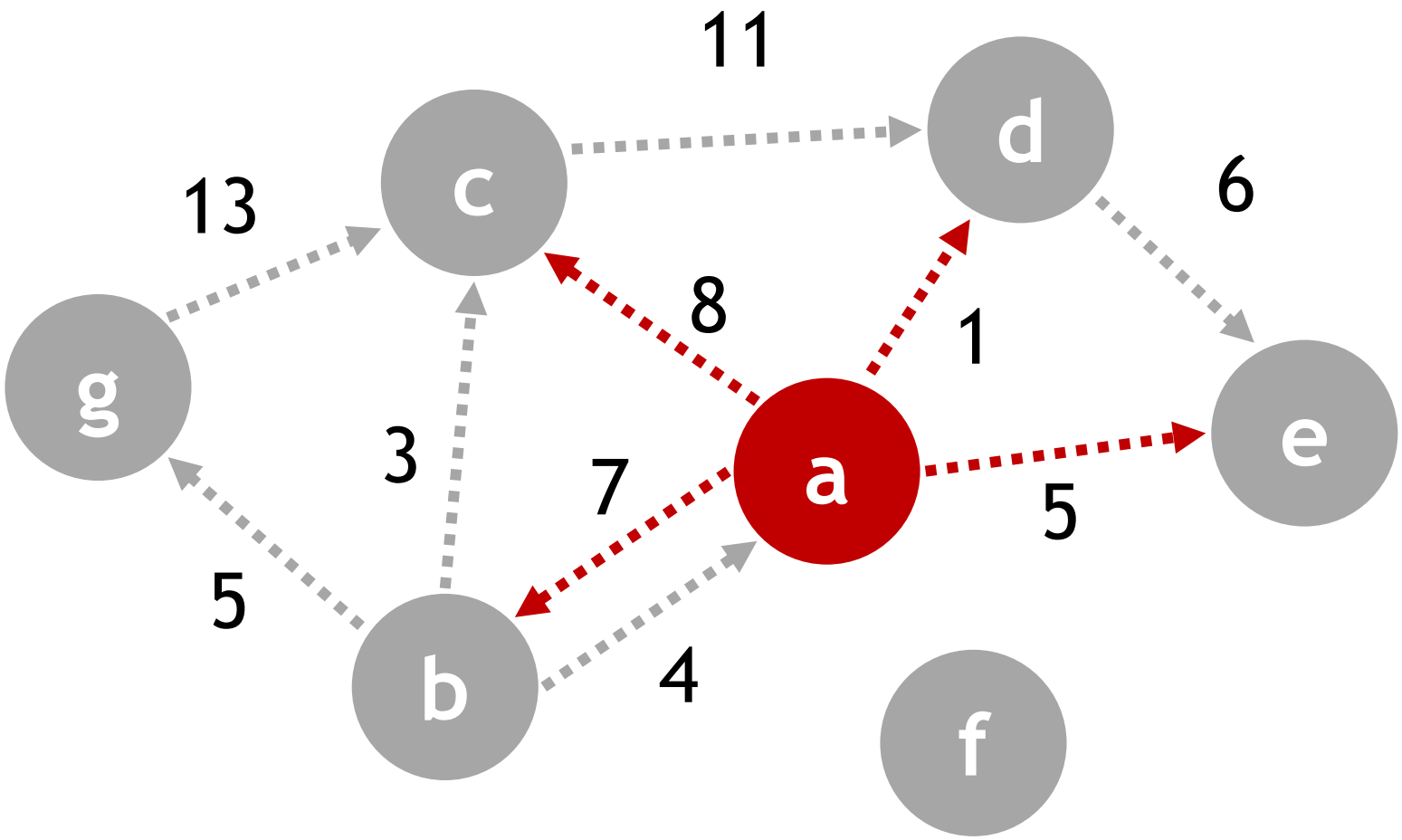
a	
b	
c	
d	
e	
f	
g	



a	
b	
c	
d	
e	
f	
g	

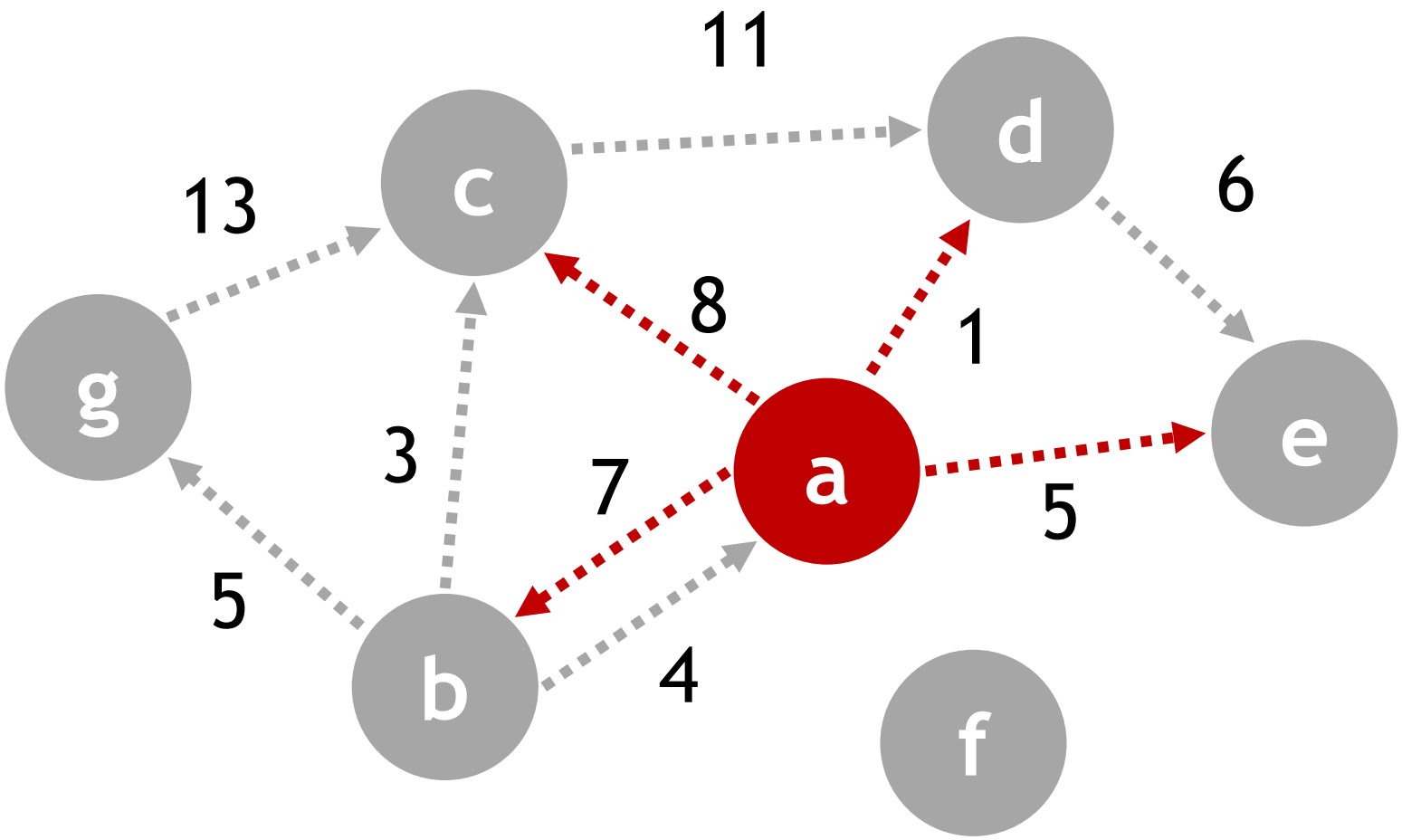


▶ a	b	c	d	e
b				
c				
d				
e				
f				
g				

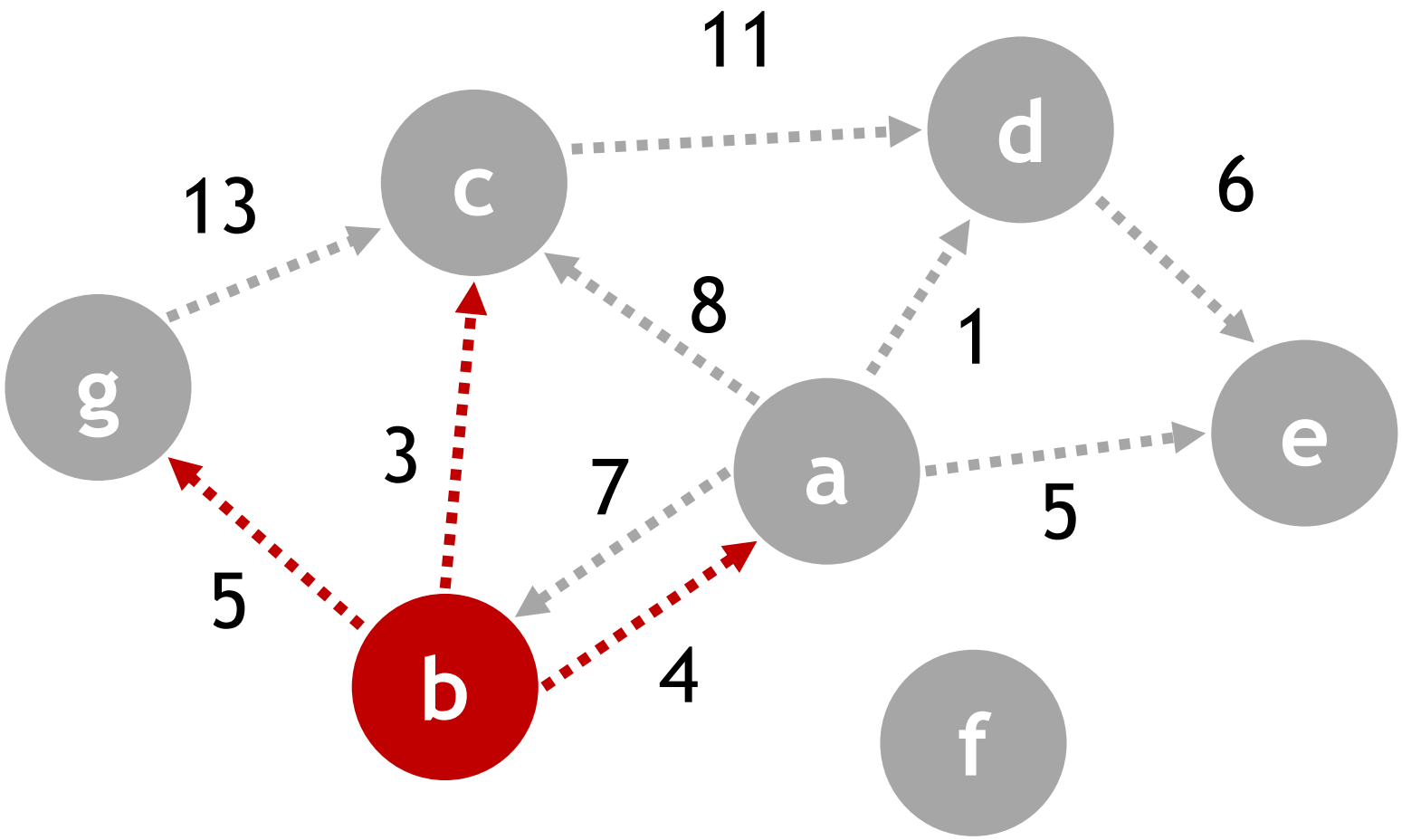


Zoznam susedov

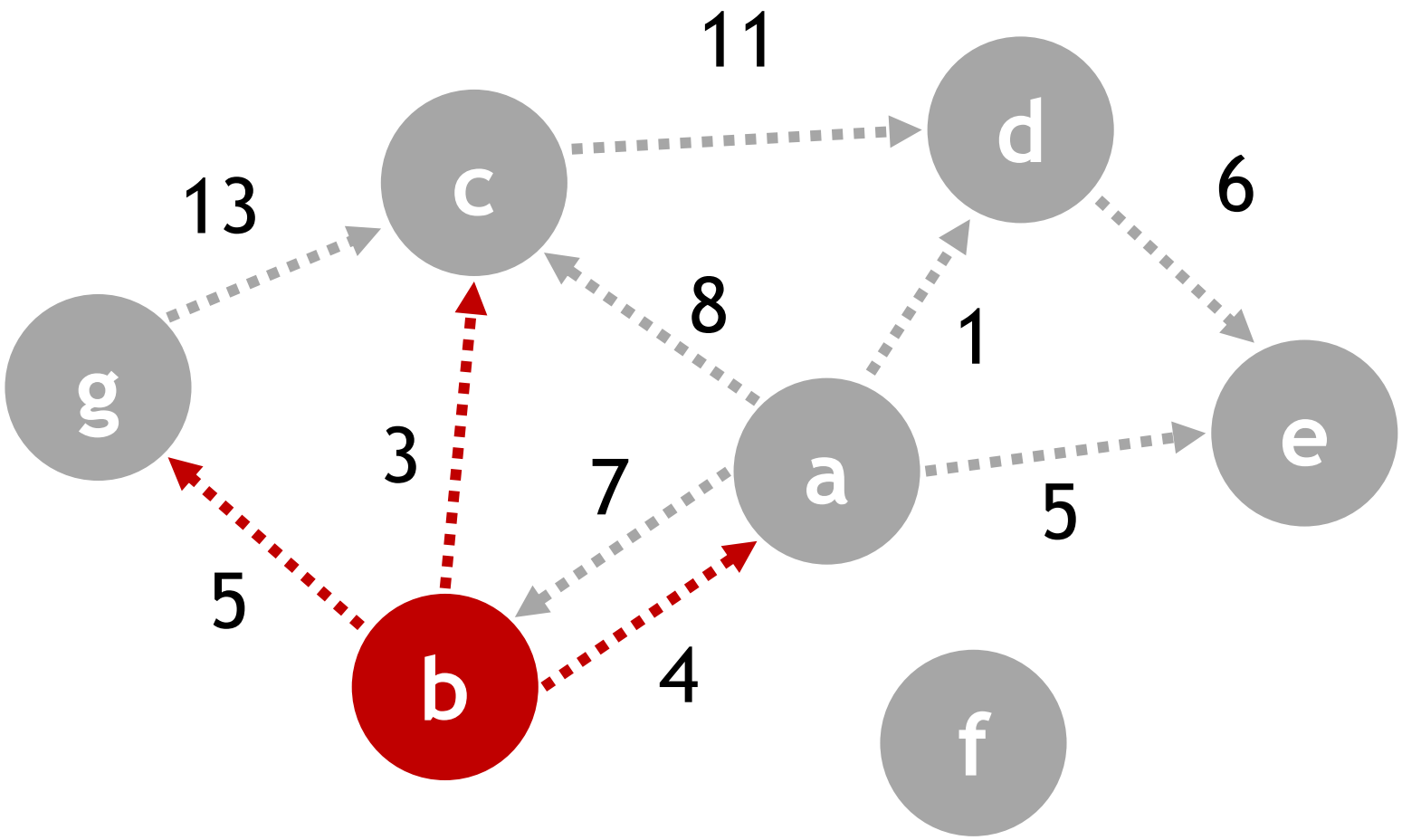
a	b c d e
b	
c	
d	
e	
f	
g	



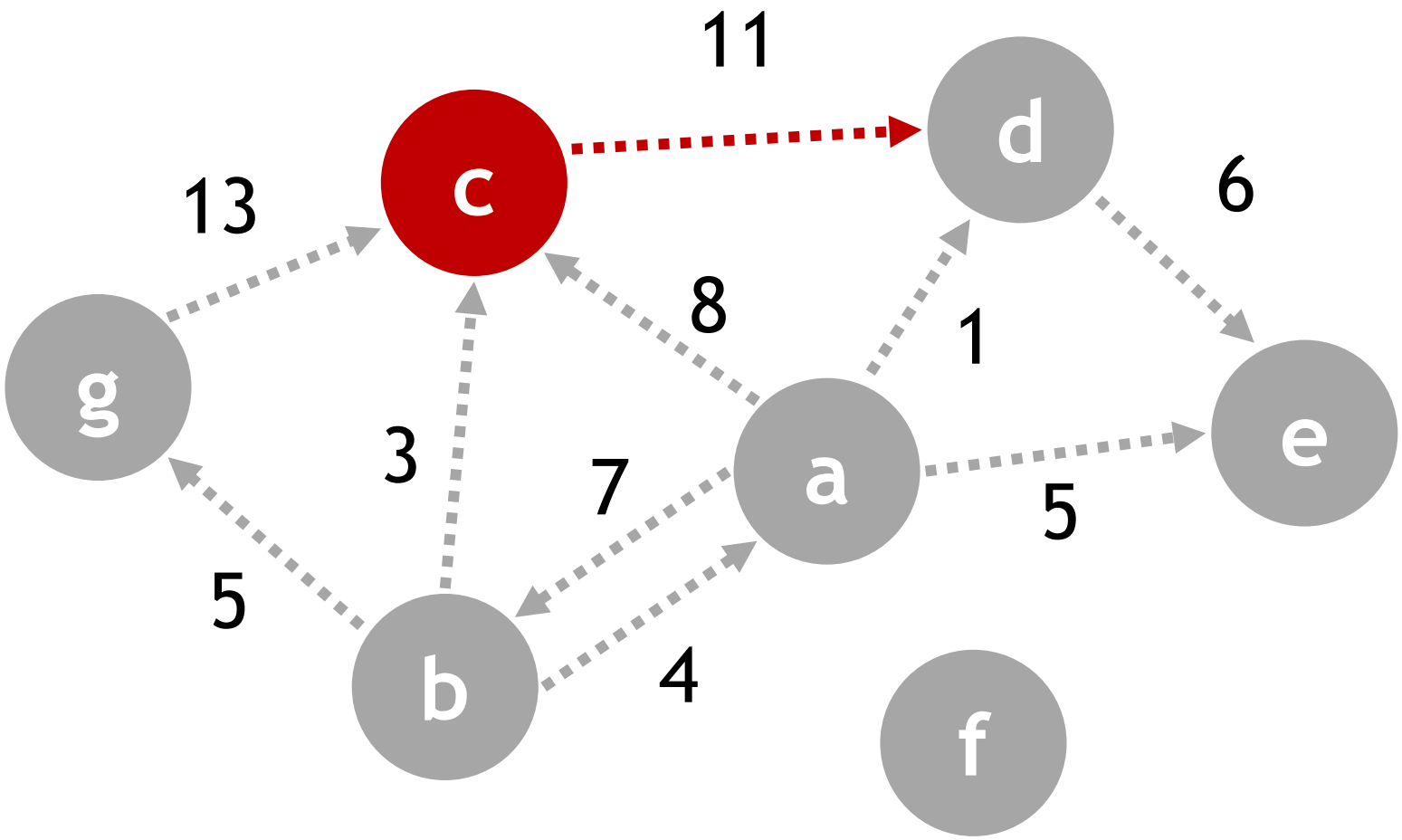
	a	b	c	d	e
▶	b				
	c				
	d				
	e				
	f				
	g				



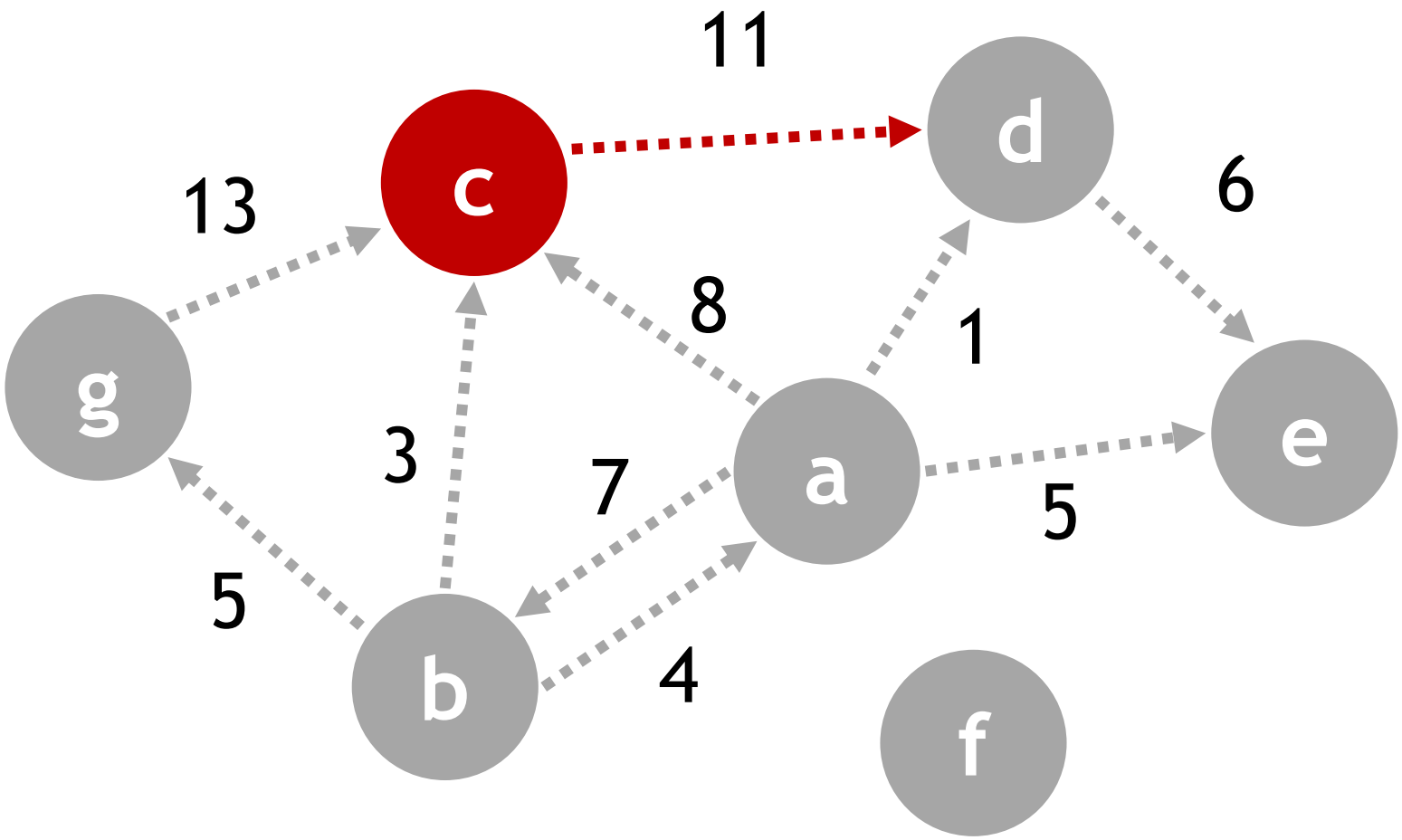
	a	b	c	d	e
▶	b	a	c	g	
	c				
	d				
	e				
	f				
	g				



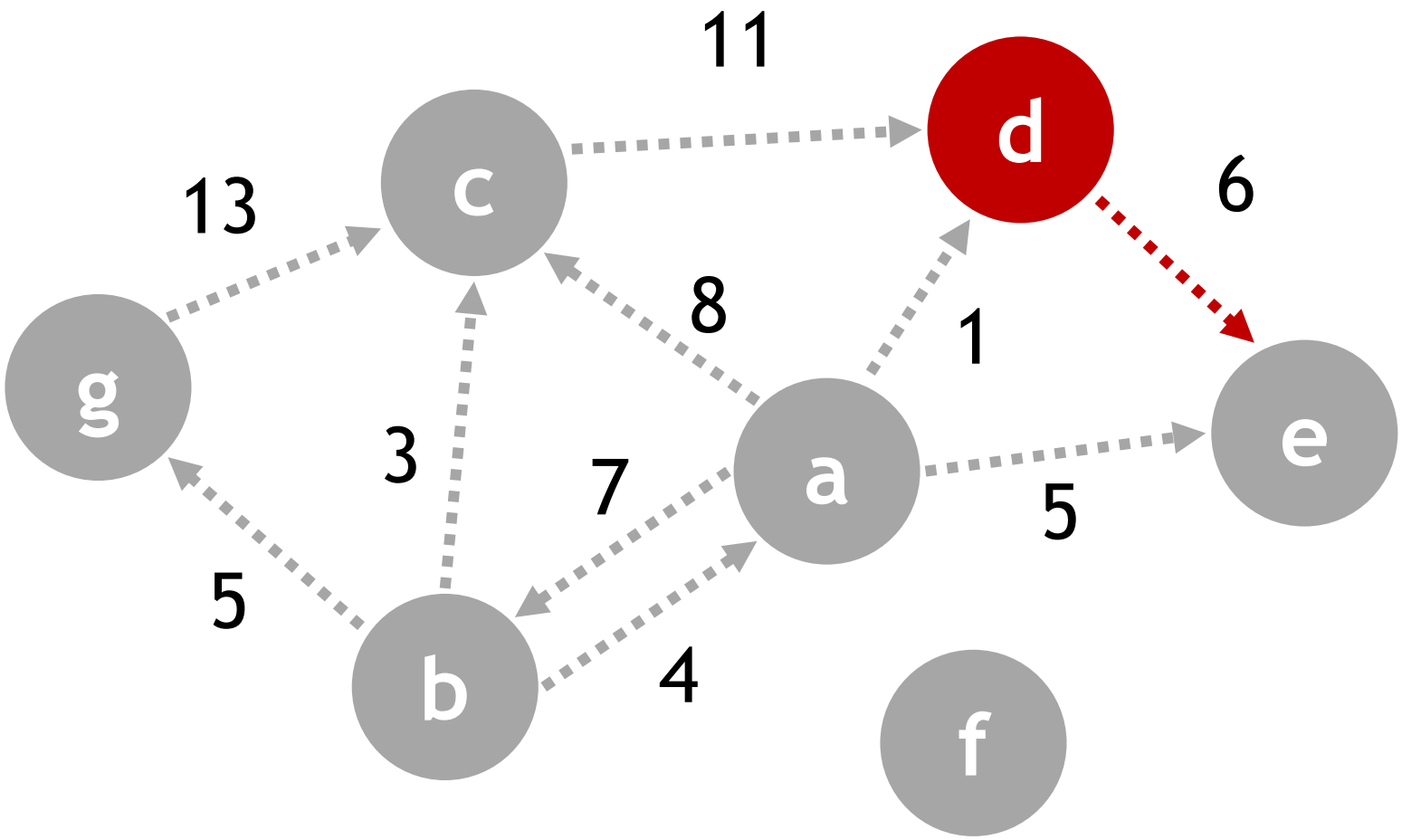
a	b c d e
b	a c g
▶ c	
d	
e	
f	
g	



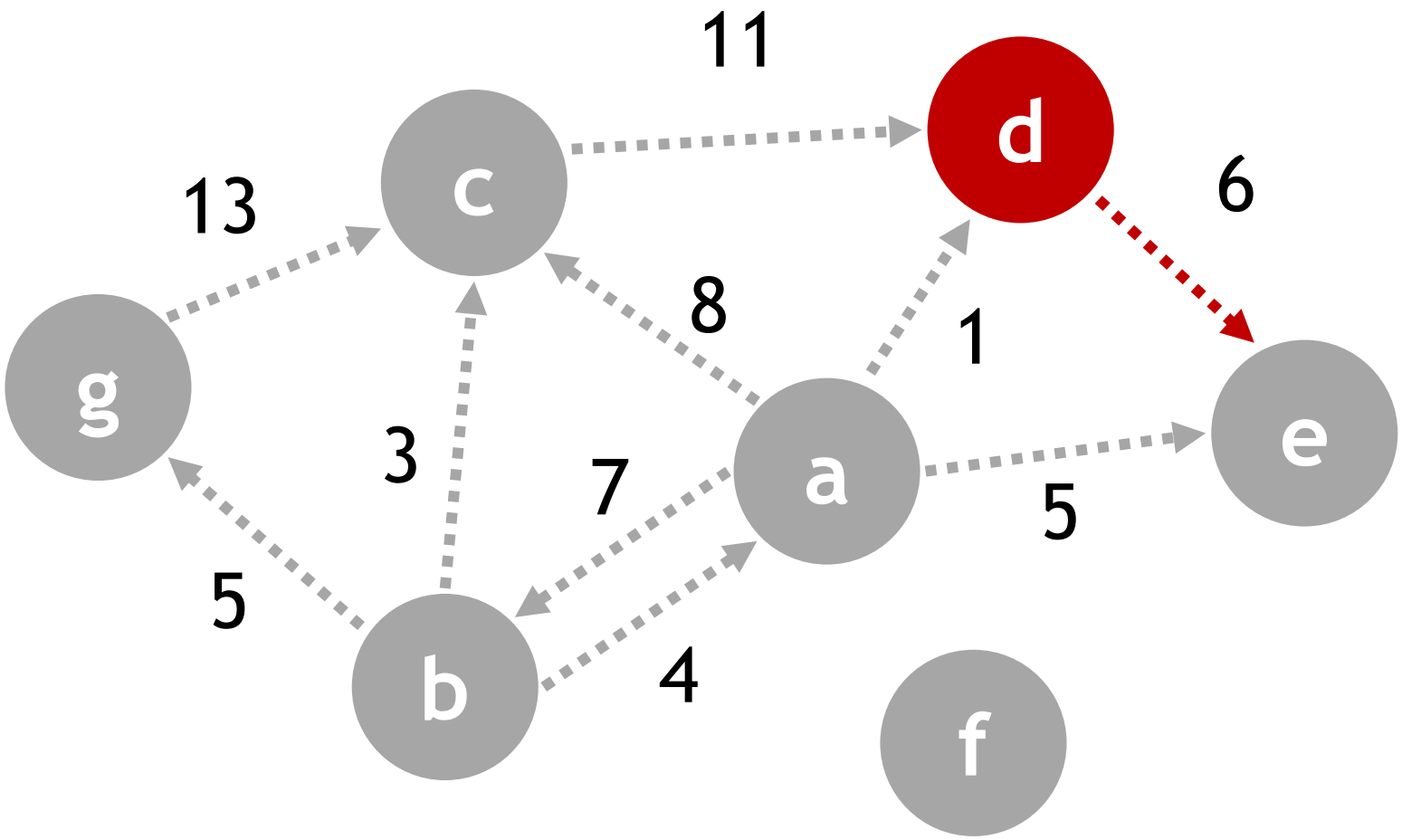
a	b c d e
b	a c g
c	d
d	
e	
f	
g	



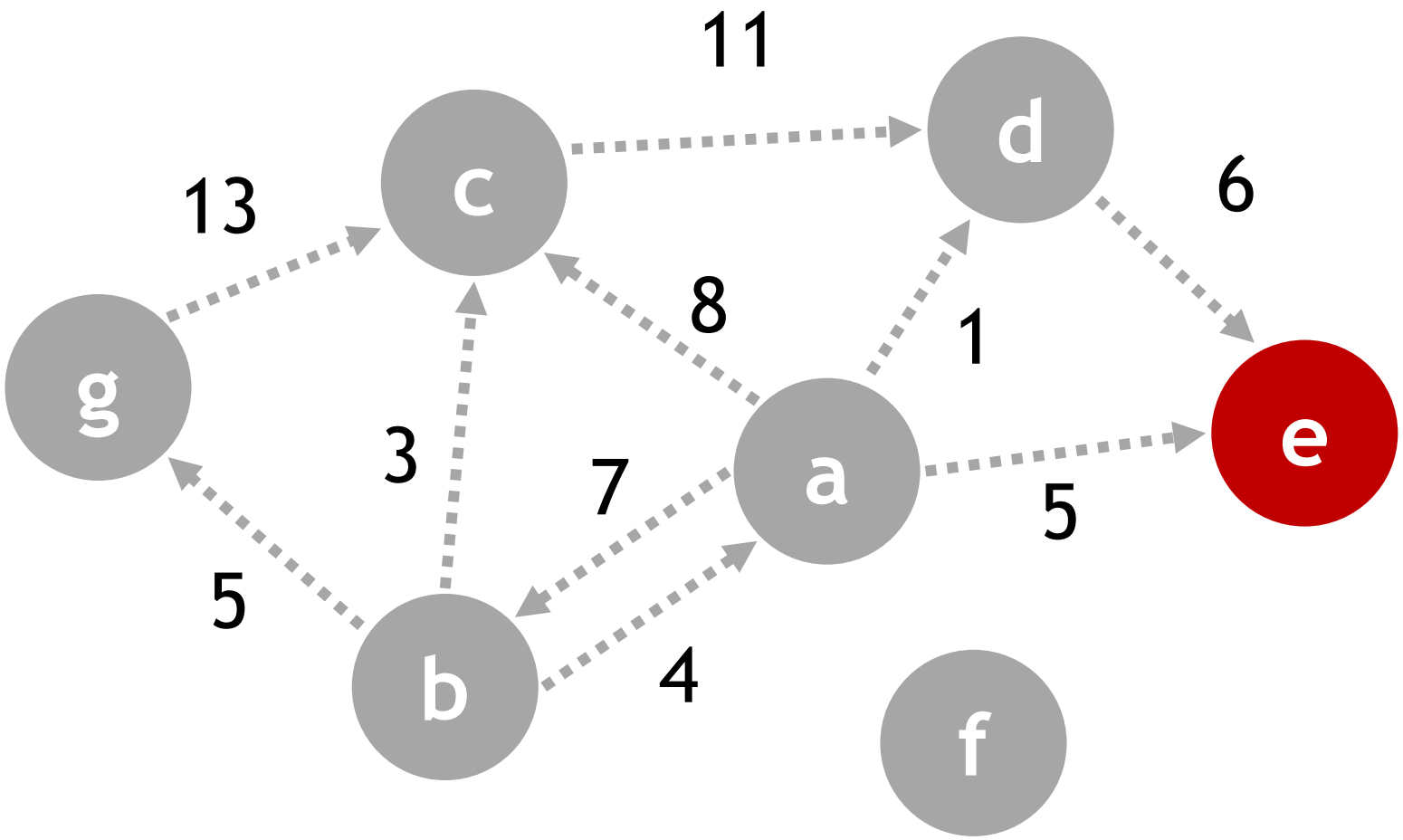
a	b c d e
b	a c g
c	d
d	
e	
f	
g	



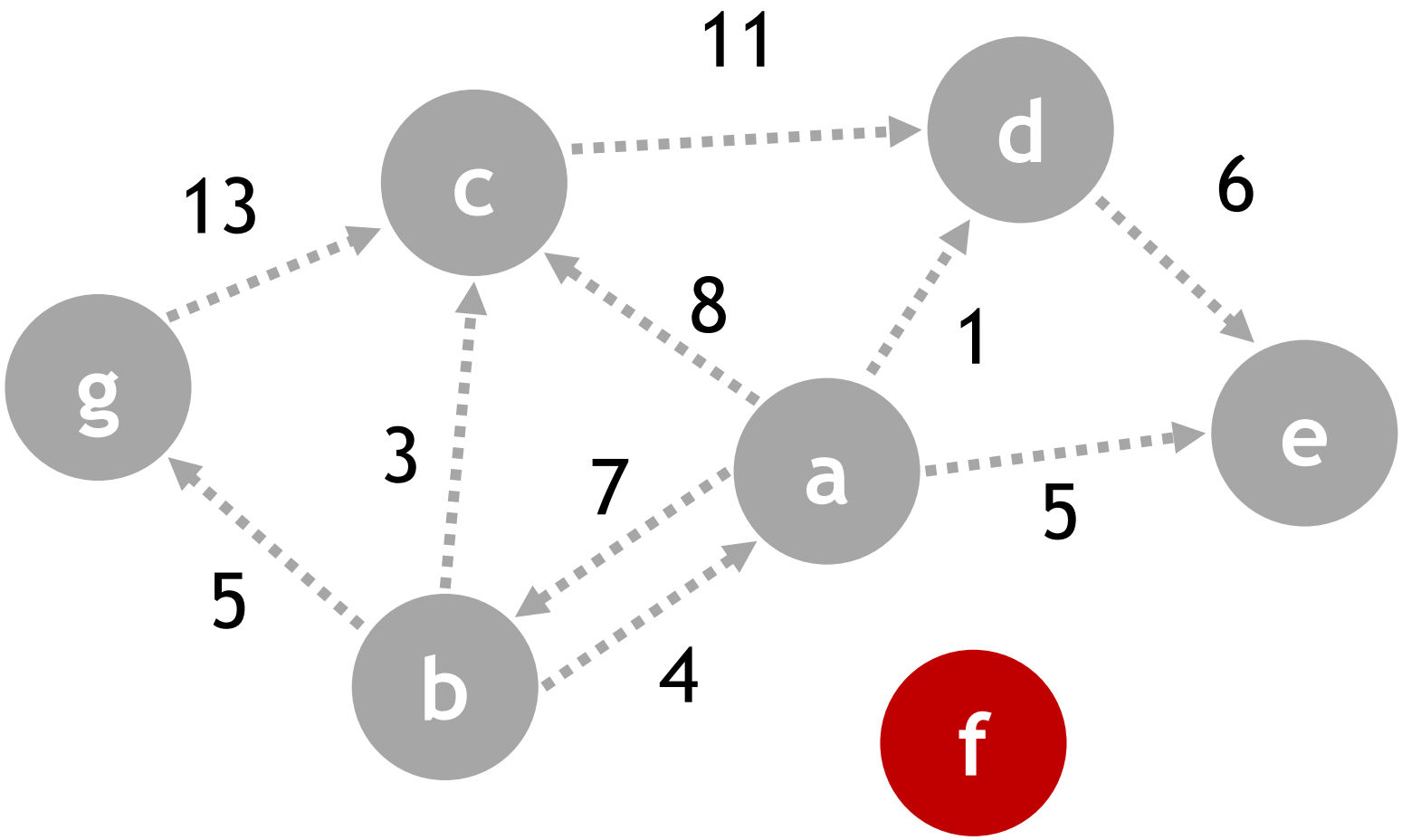
a	b	c	d	e	
b	a	c	g		
c	d				
▶ d	e				
e					
f					
g					



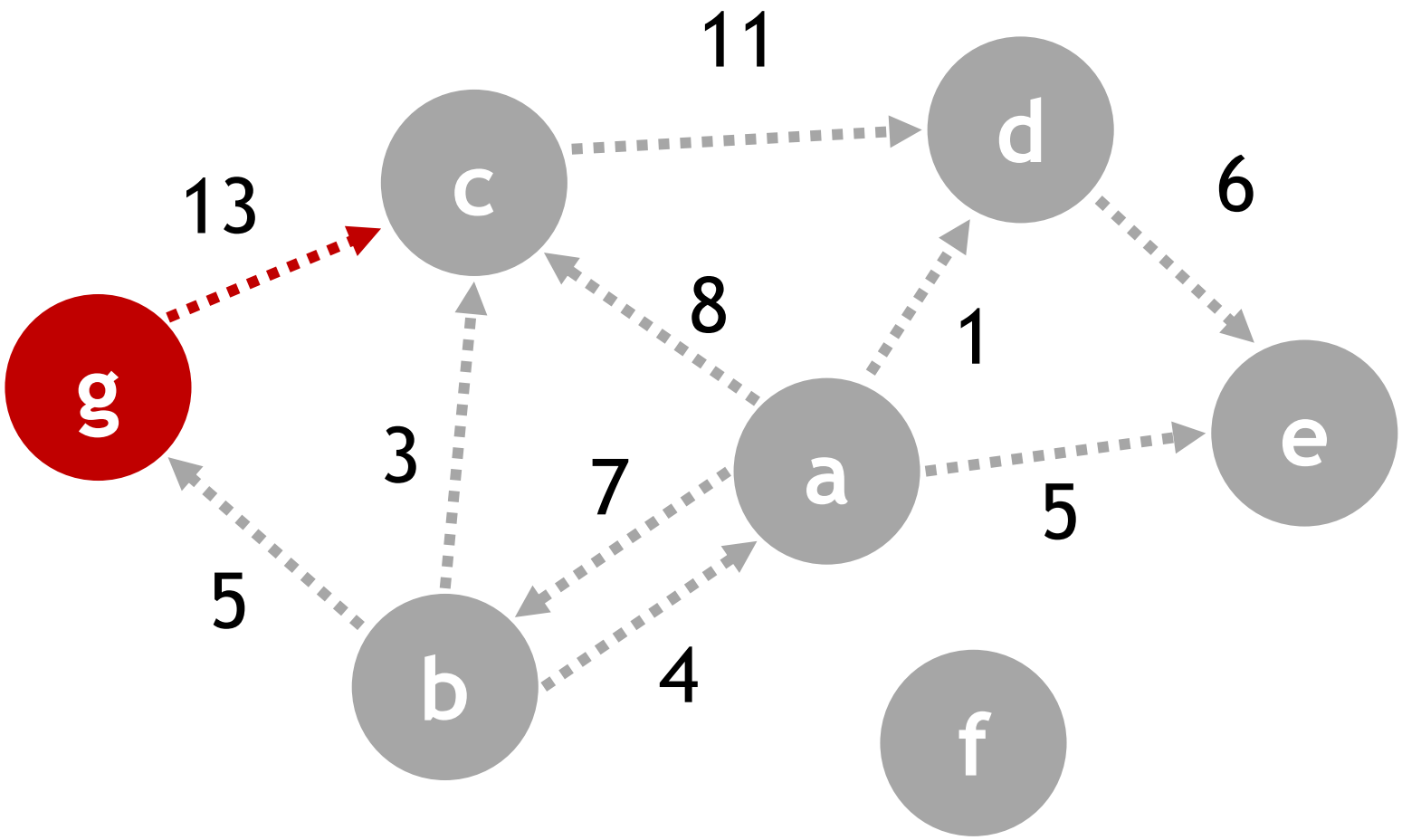
a	b c d e
b	a c g
c	d
d	e
e	
f	
g	



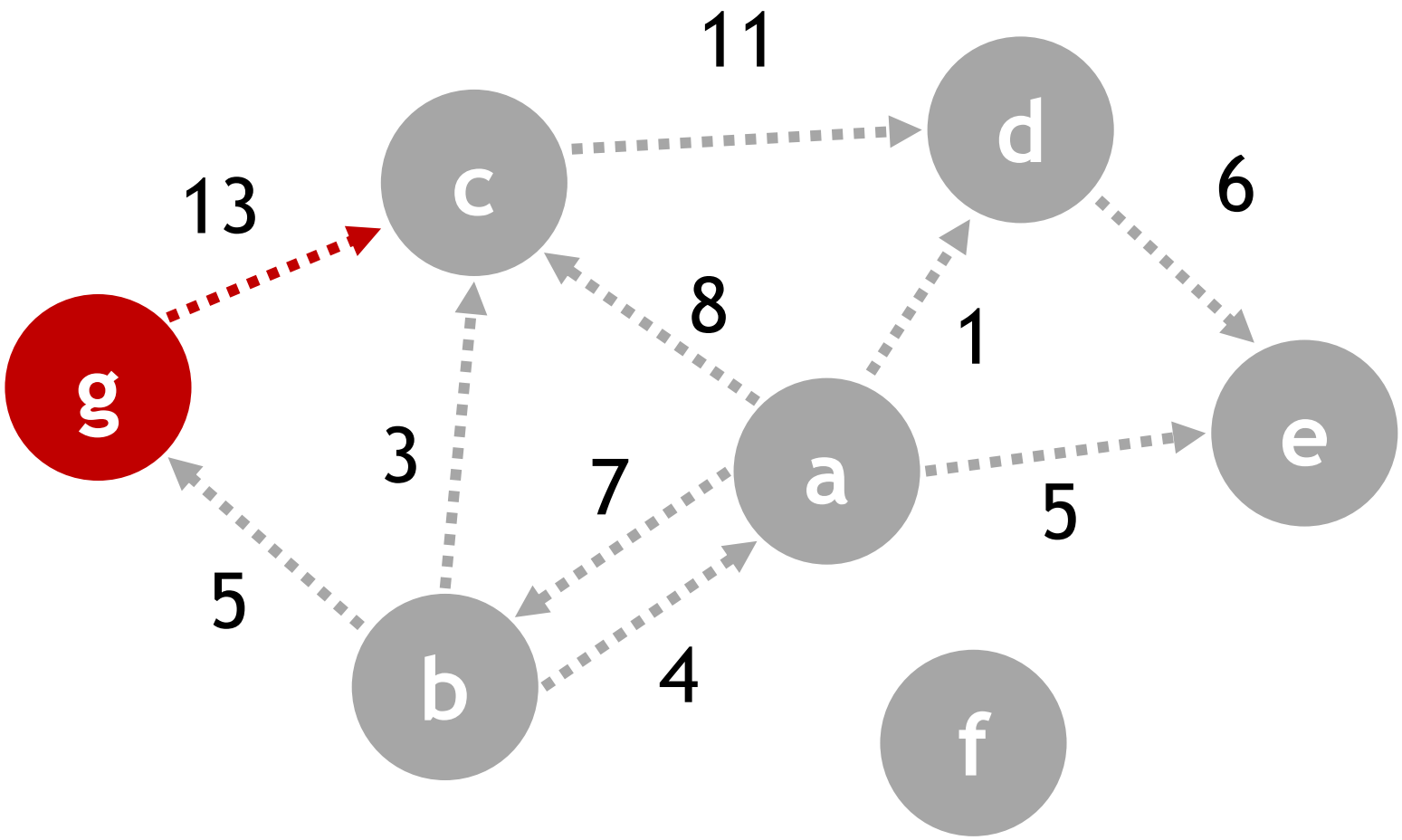
a	b c d e
b	a c g
c	d
d	e
e	
f	
g	



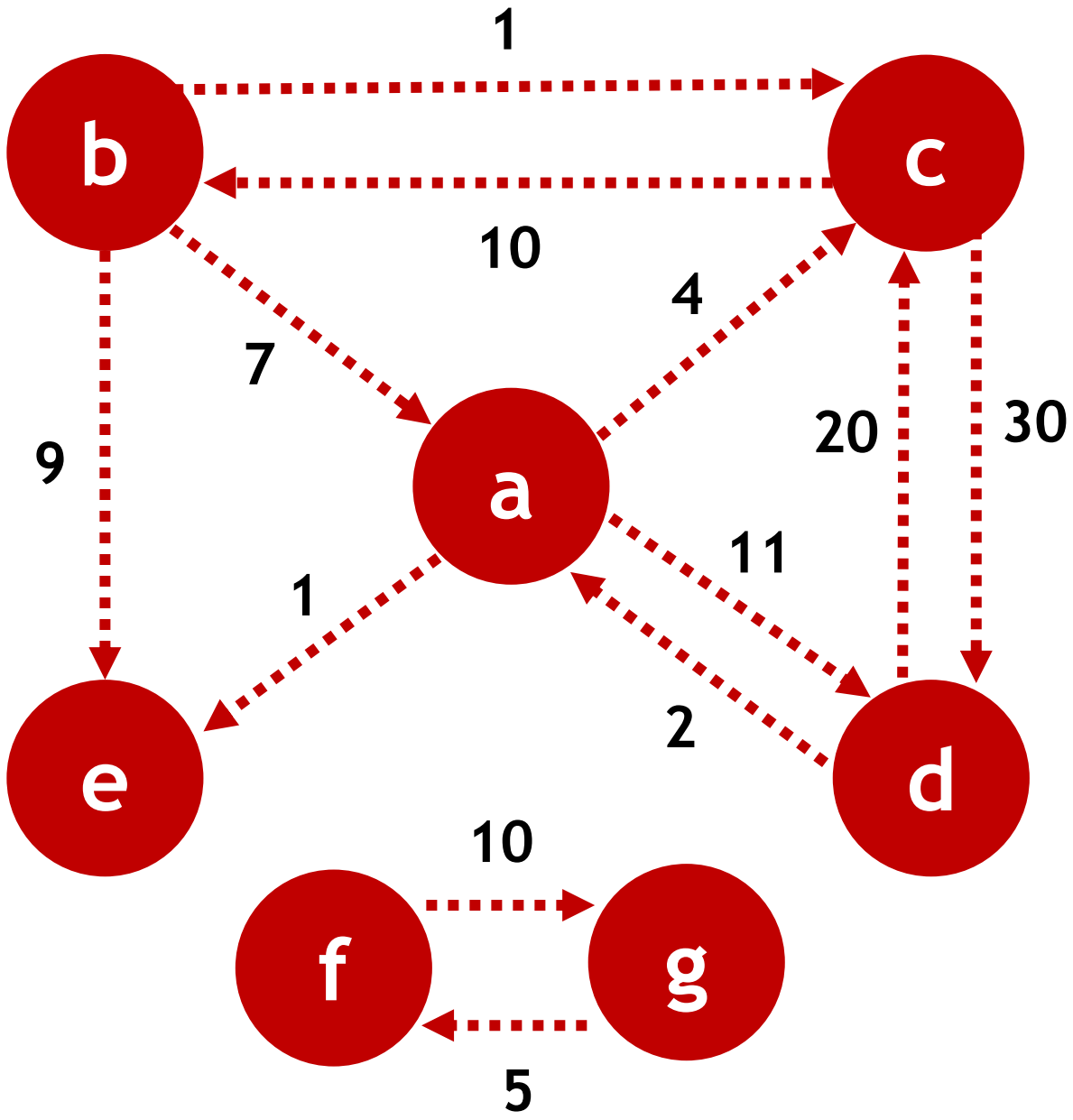
a	b c d e
b	a c g
c	d
d	e
e	
f	
g	

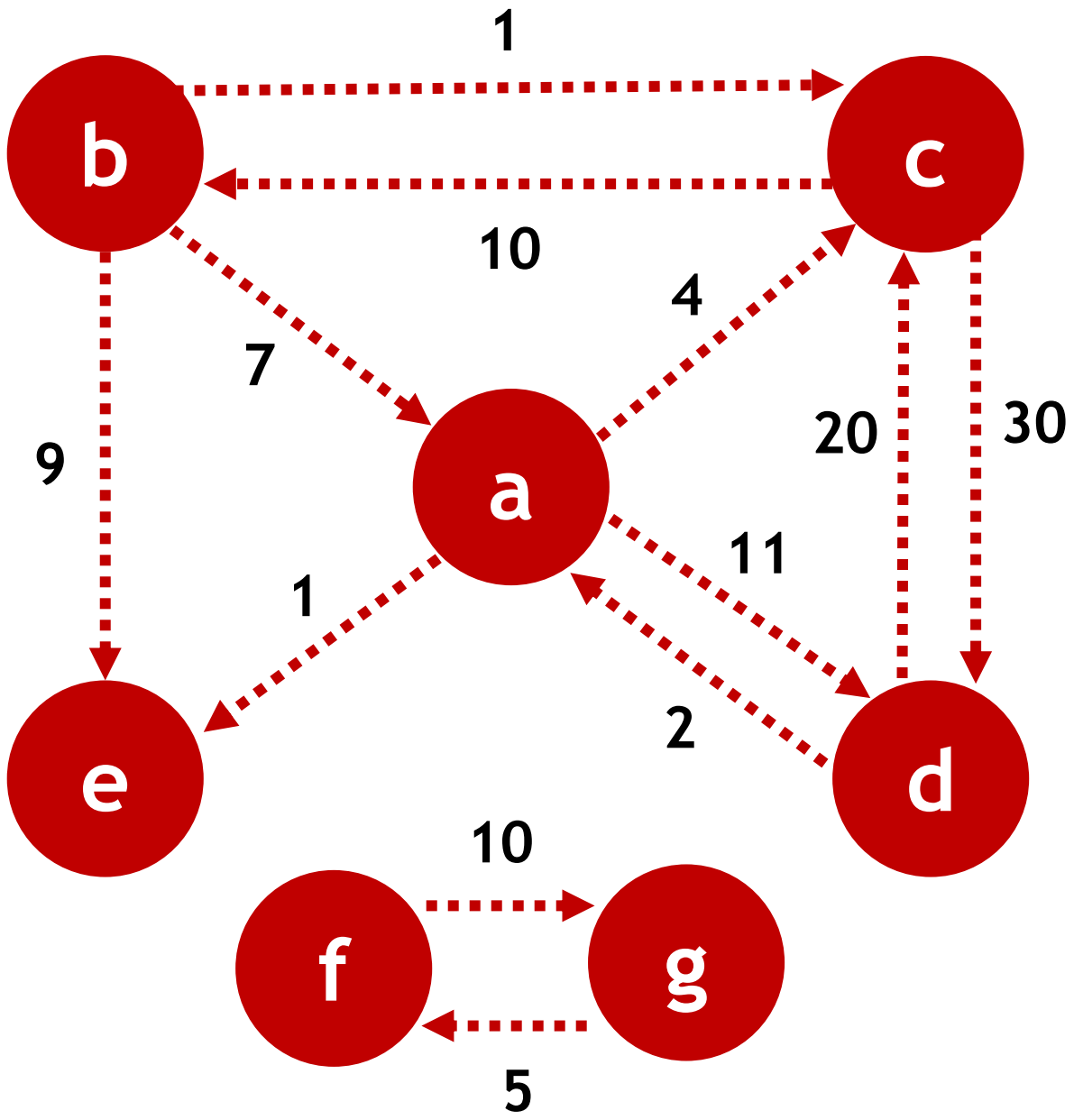


a	b c d e
b	a c g
c	d
d	e
e	
f	
g	c

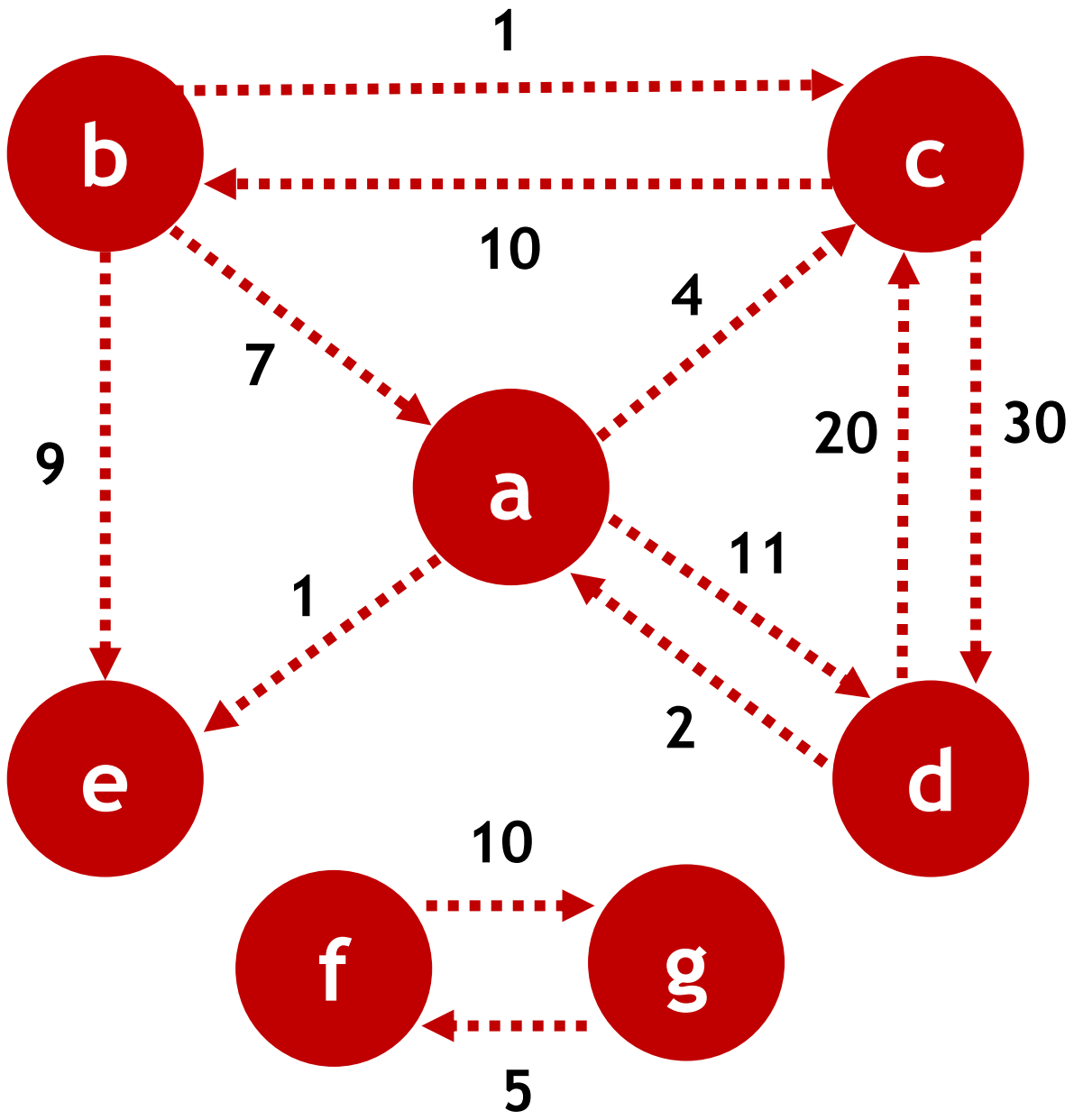


Implementácia zoznamu susedov

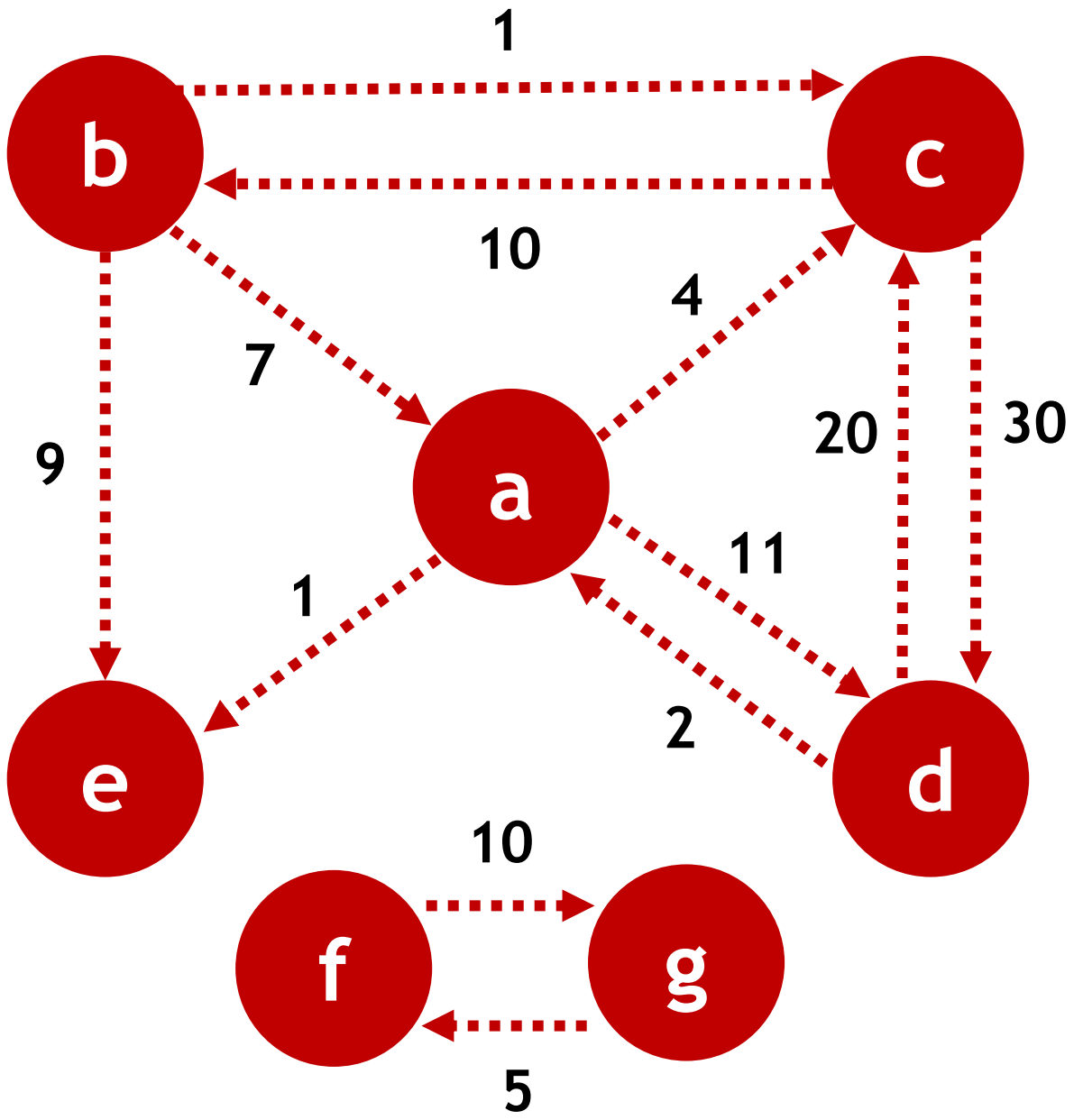

1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5



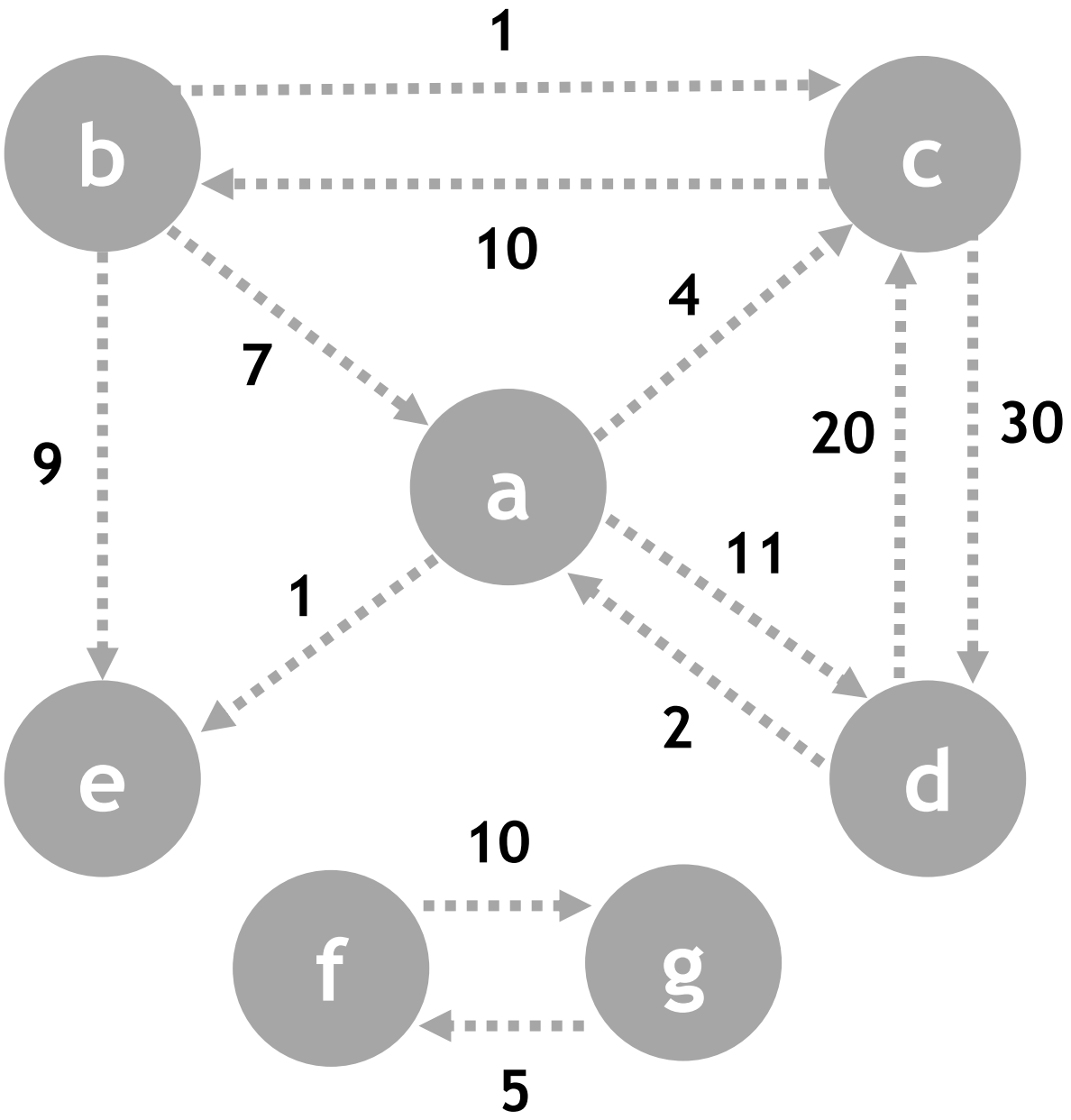

```
// reprezentacia vrcholu grafu
struct Vertex {
    string name; // nazov vrcholu
    list<EdgeTo> edges; // zoznam susedov
};
```

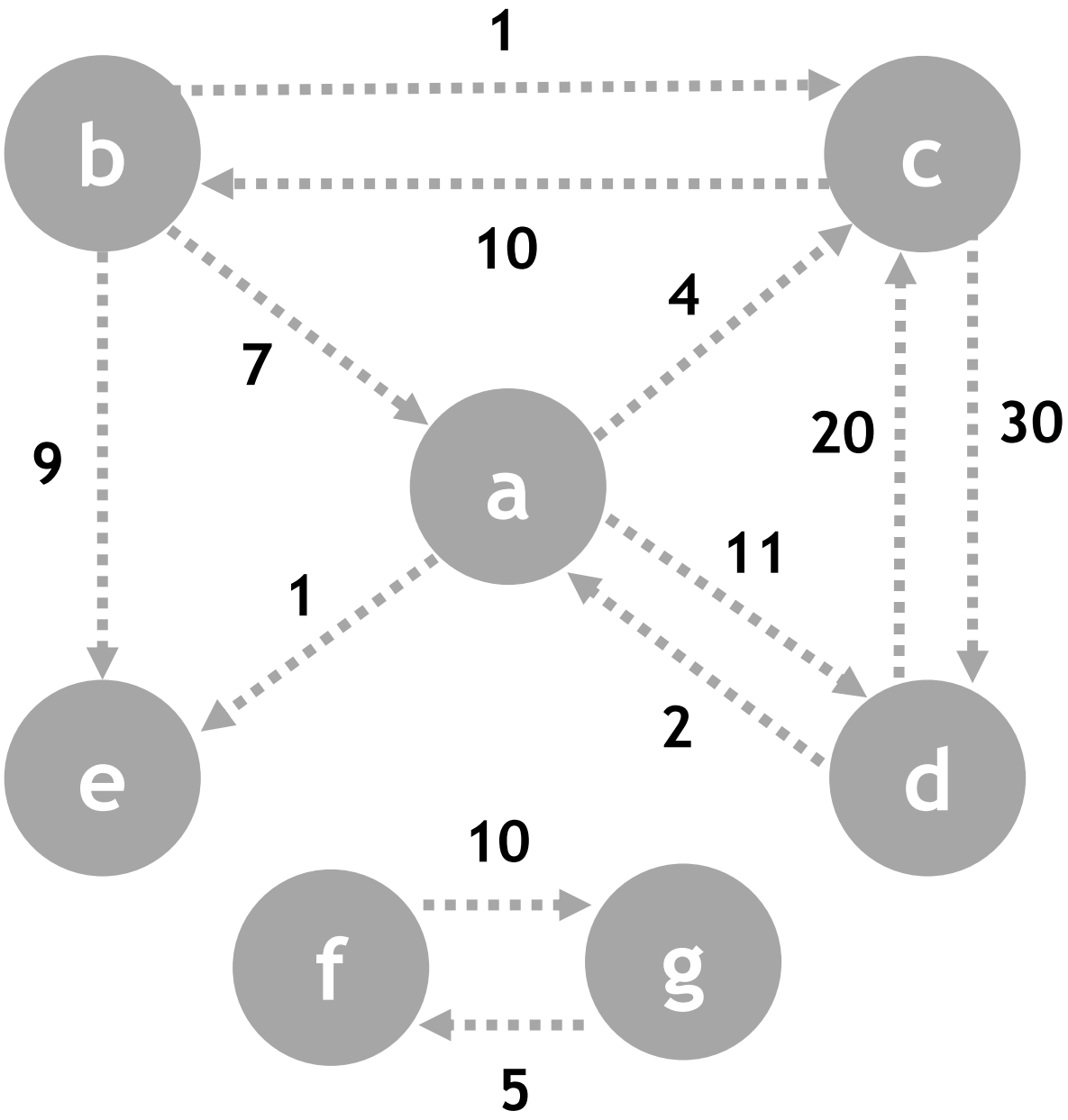


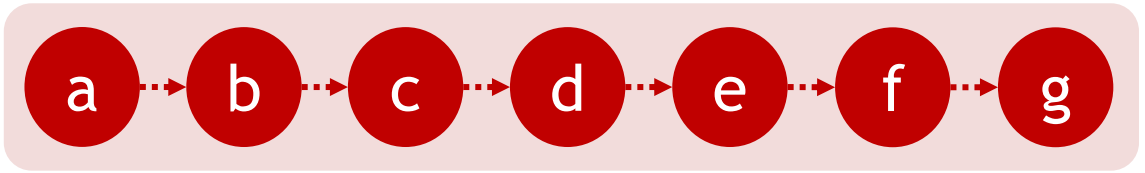

```cpp
// reprezentacia vrcholu grafu
struct Vertex {
    string name; // nazov vrcholu
    list<EdgeTo> edges; // zoznam susedov
};

// reprezentacia hrany grafu
struct EdgeTo {
    Vertex *endVertex; // koncovy vrchol
    unsigned length; // dlzka hrany
};
```

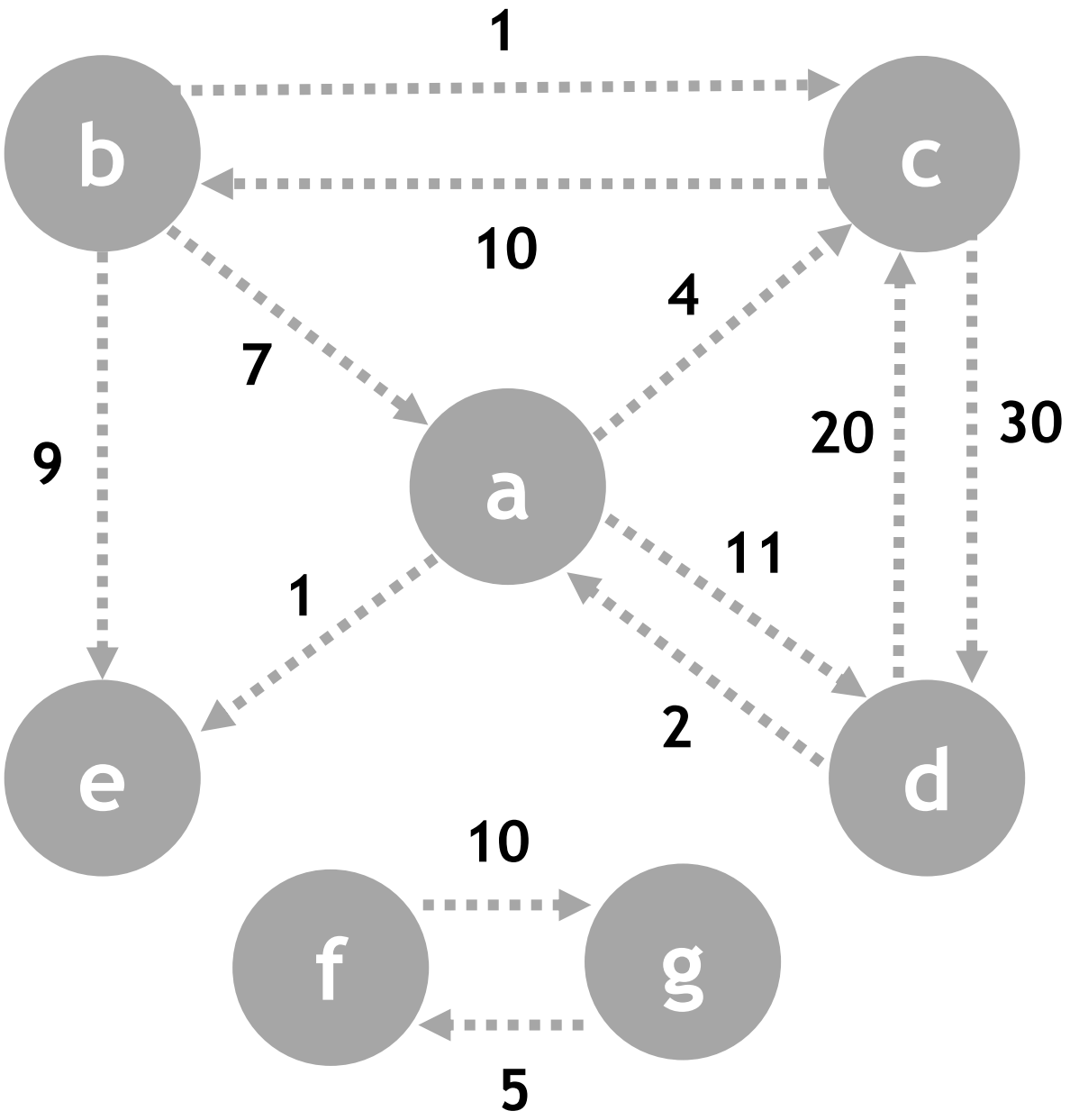



```cpp
// reprezentacia vrcholu grafu
struct Vertex {
    string name; // nazov vrcholu
    list<EdgeTo> edges; // zoznam susedov
};

// reprezentacia hrany grafu
struct EdgeTo {
    Vertex *endVertex; // koncovy vrchol
    unsigned length; // dlzka hrany
};

// reprezentacia grafu
struct Graph {
    list<Vertex> vertices;
};
```

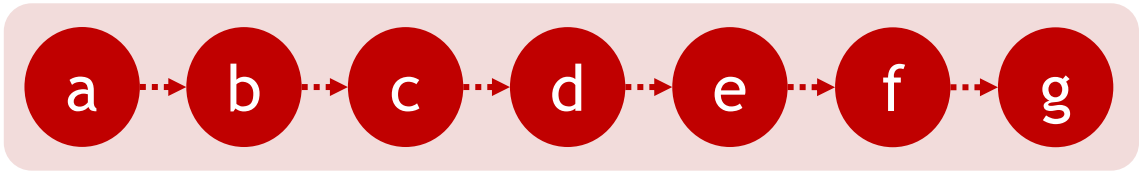

1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5

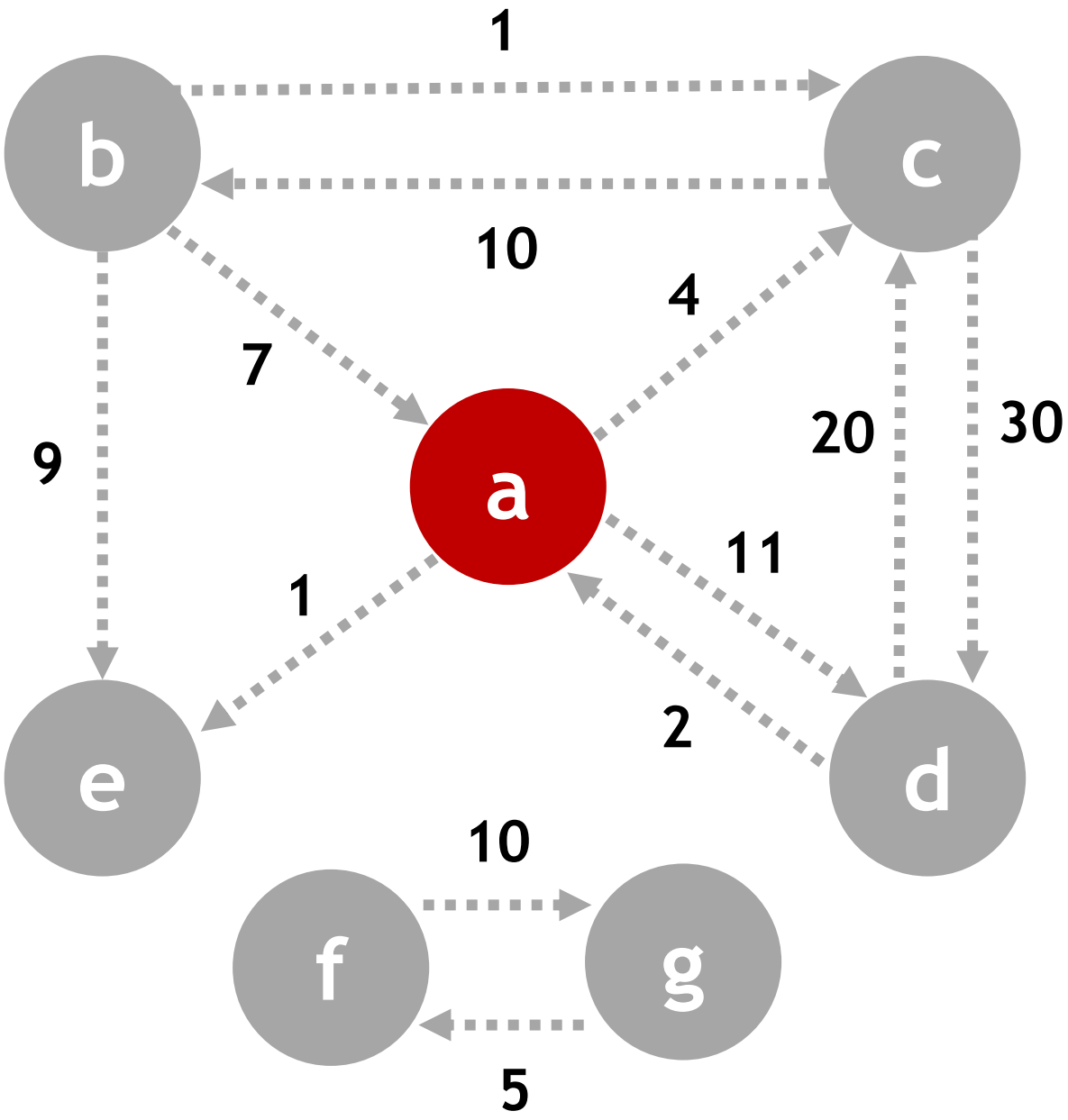

1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5


a
b
c
d
e
f
g

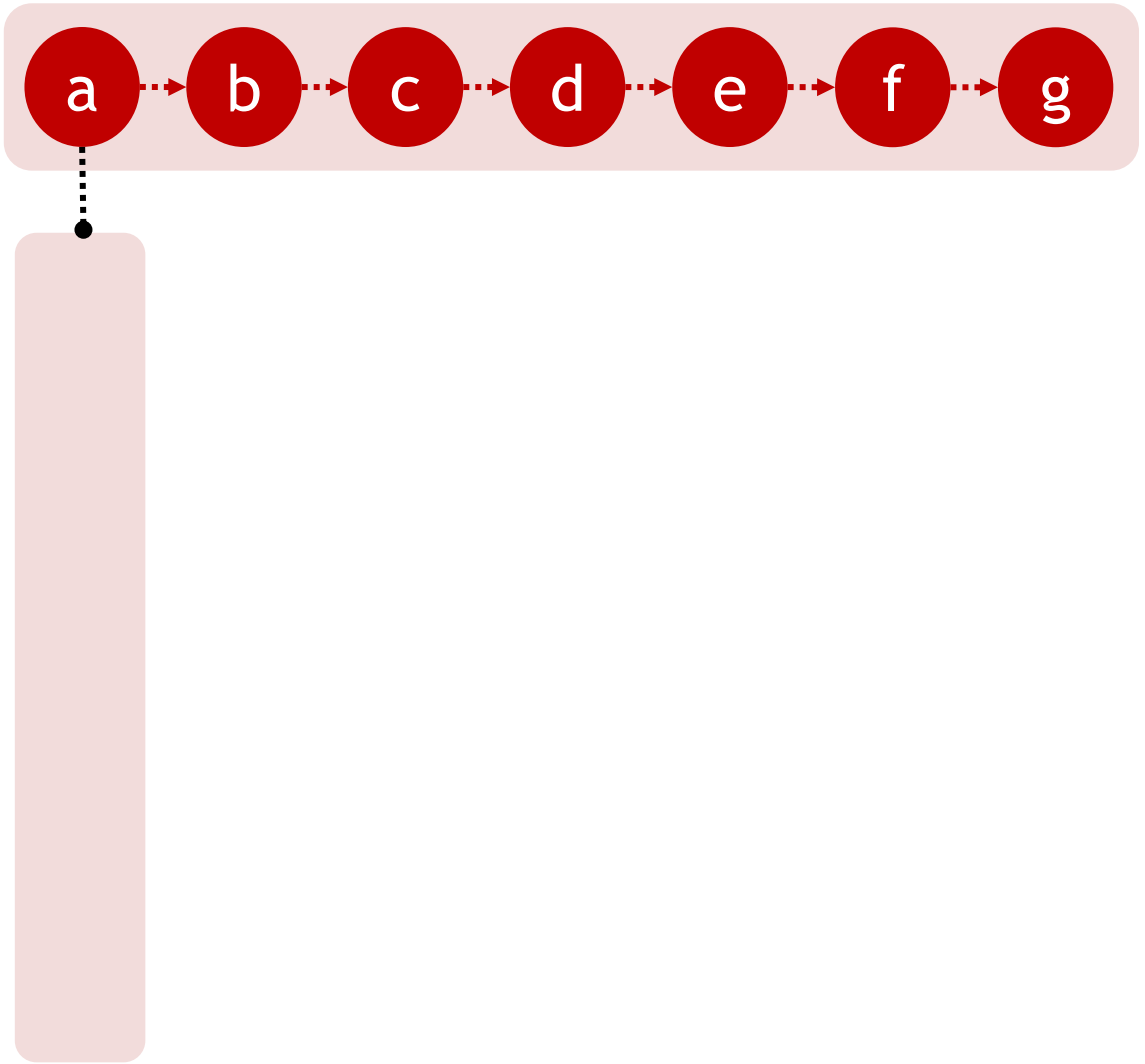


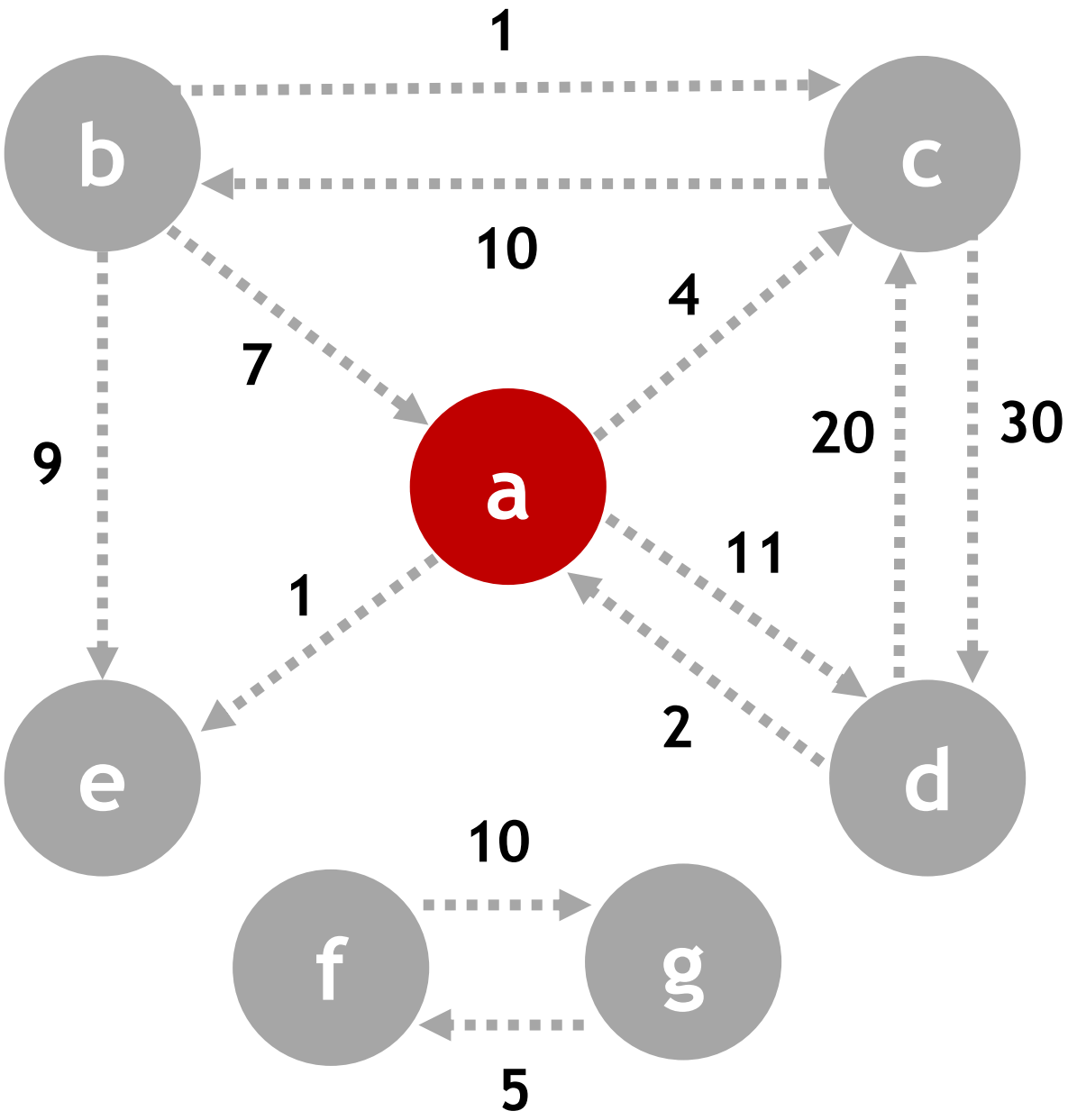

```
list<Vertex> vertices;
```

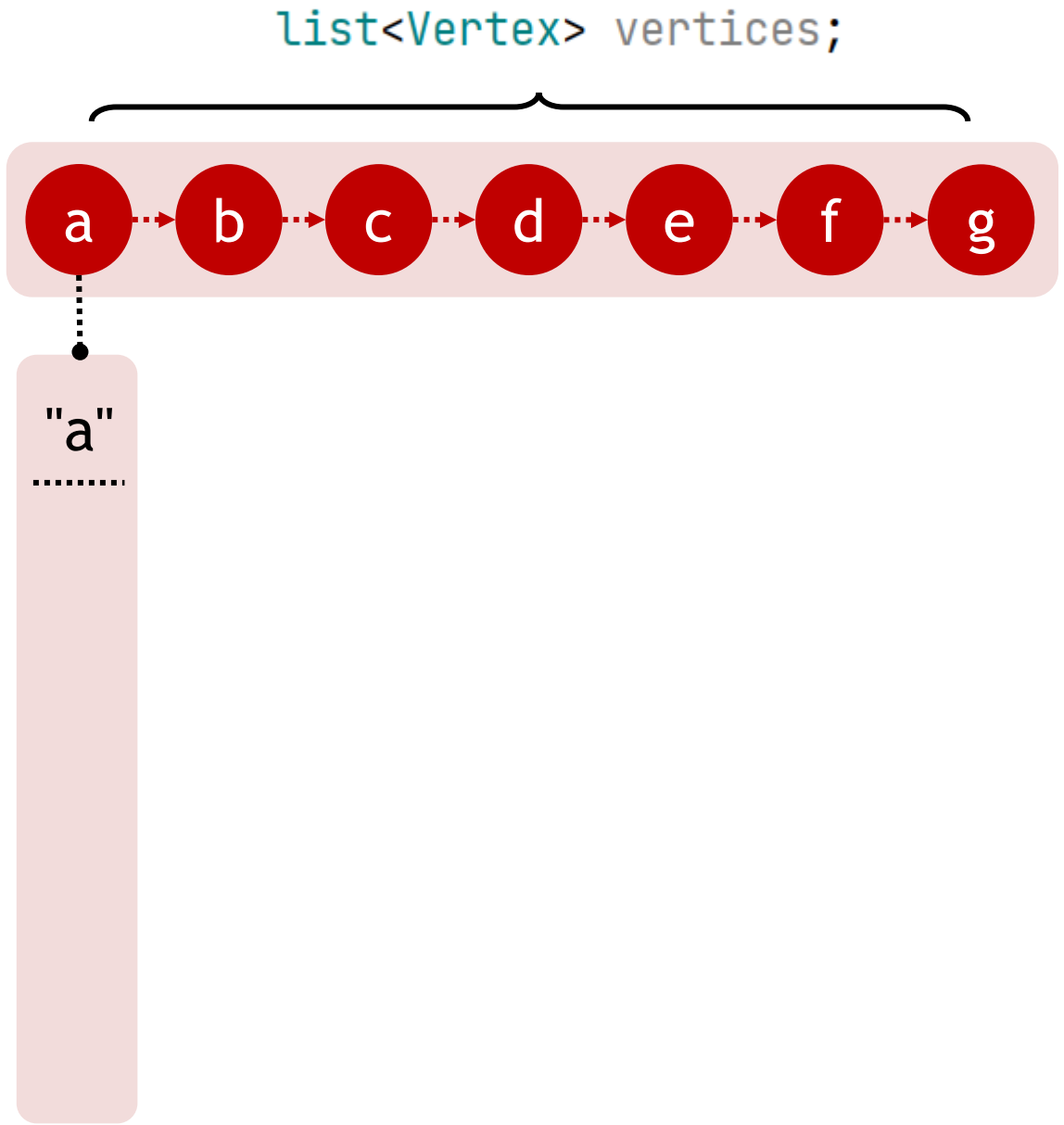


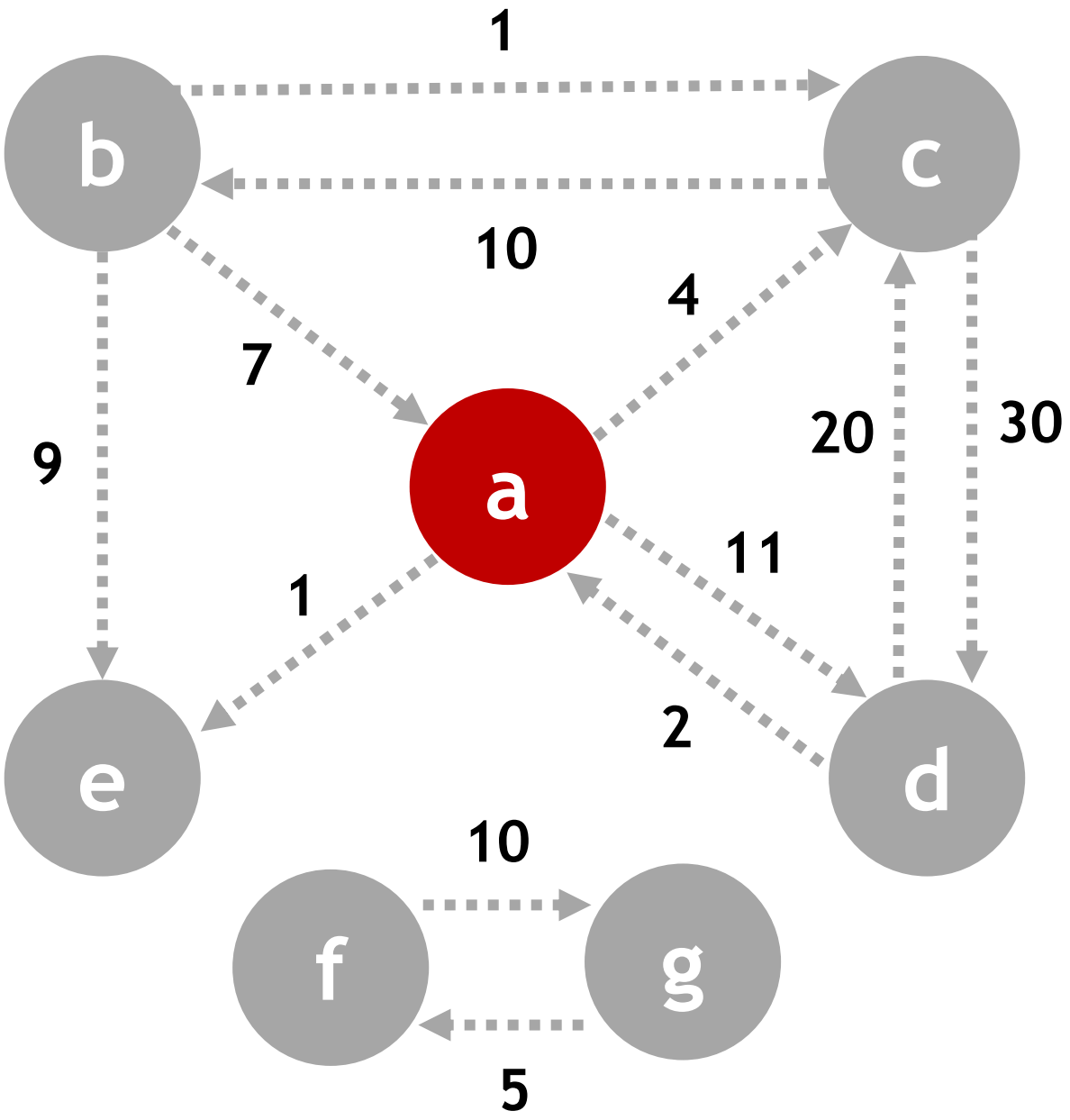



```
list<Vertex> vertices;
```

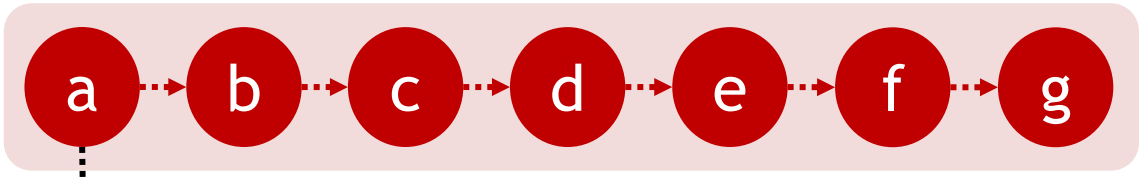


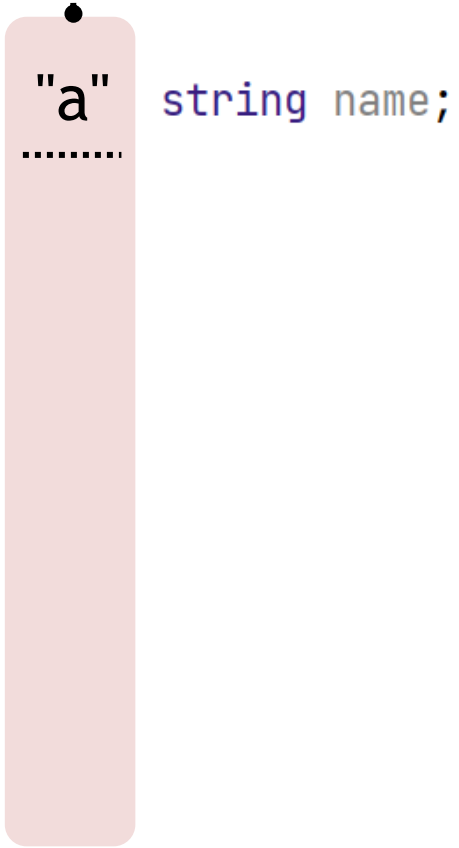

1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5


list<Vertex> vertices;
a
b
c
d
e
f
g
"a"

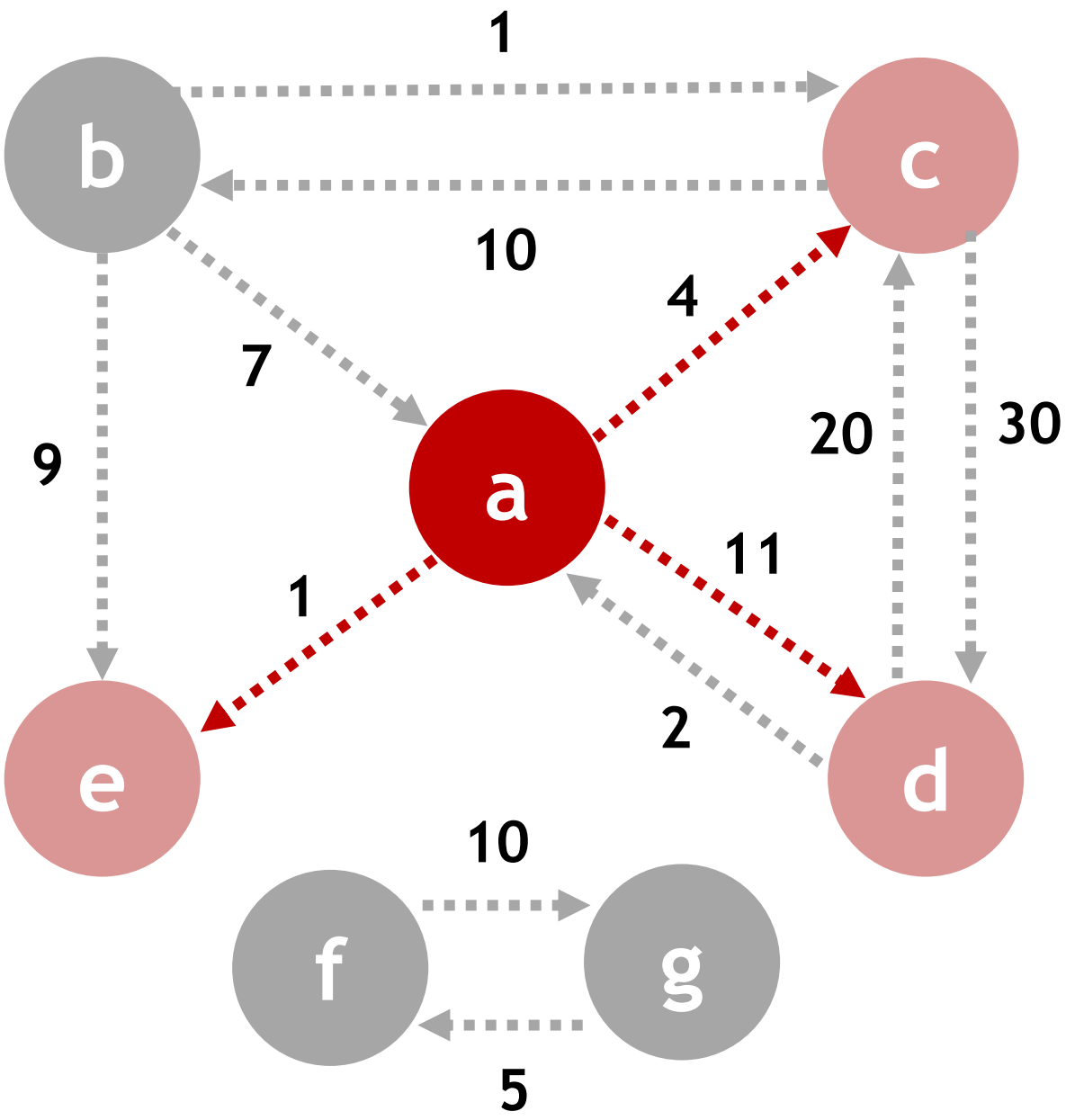



```
list<Vertex> vertices;
```

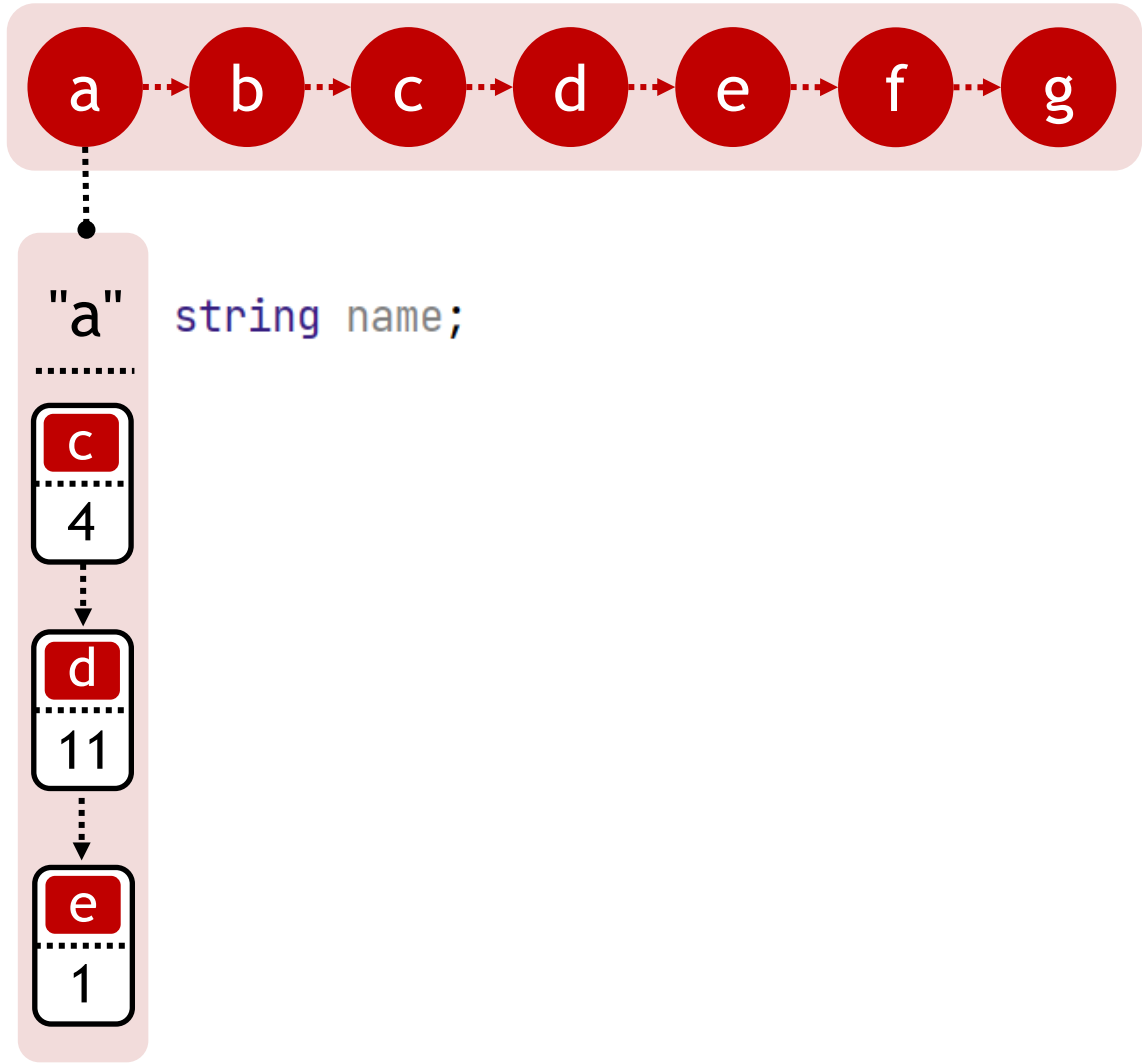


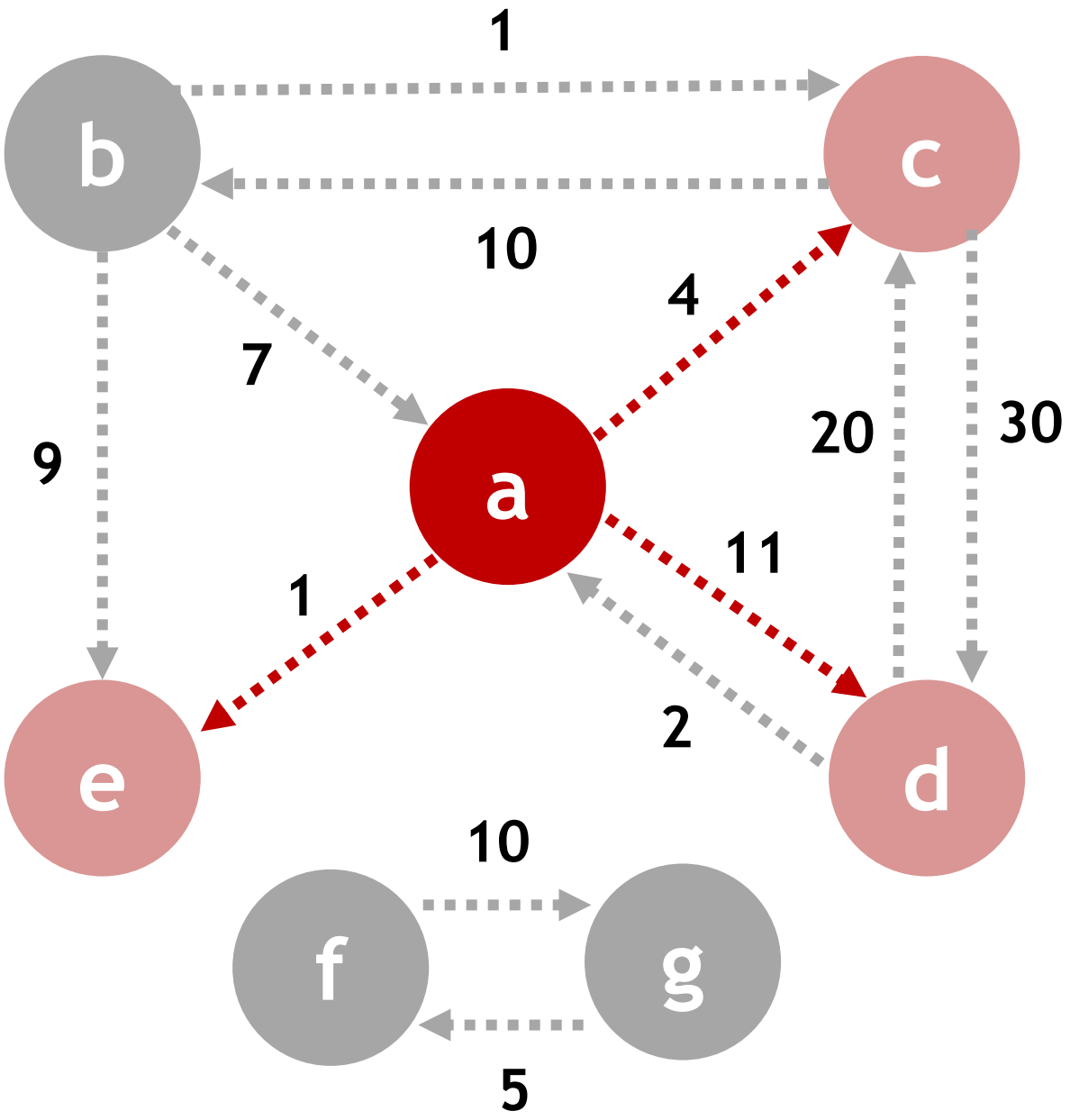



```
string name;
```

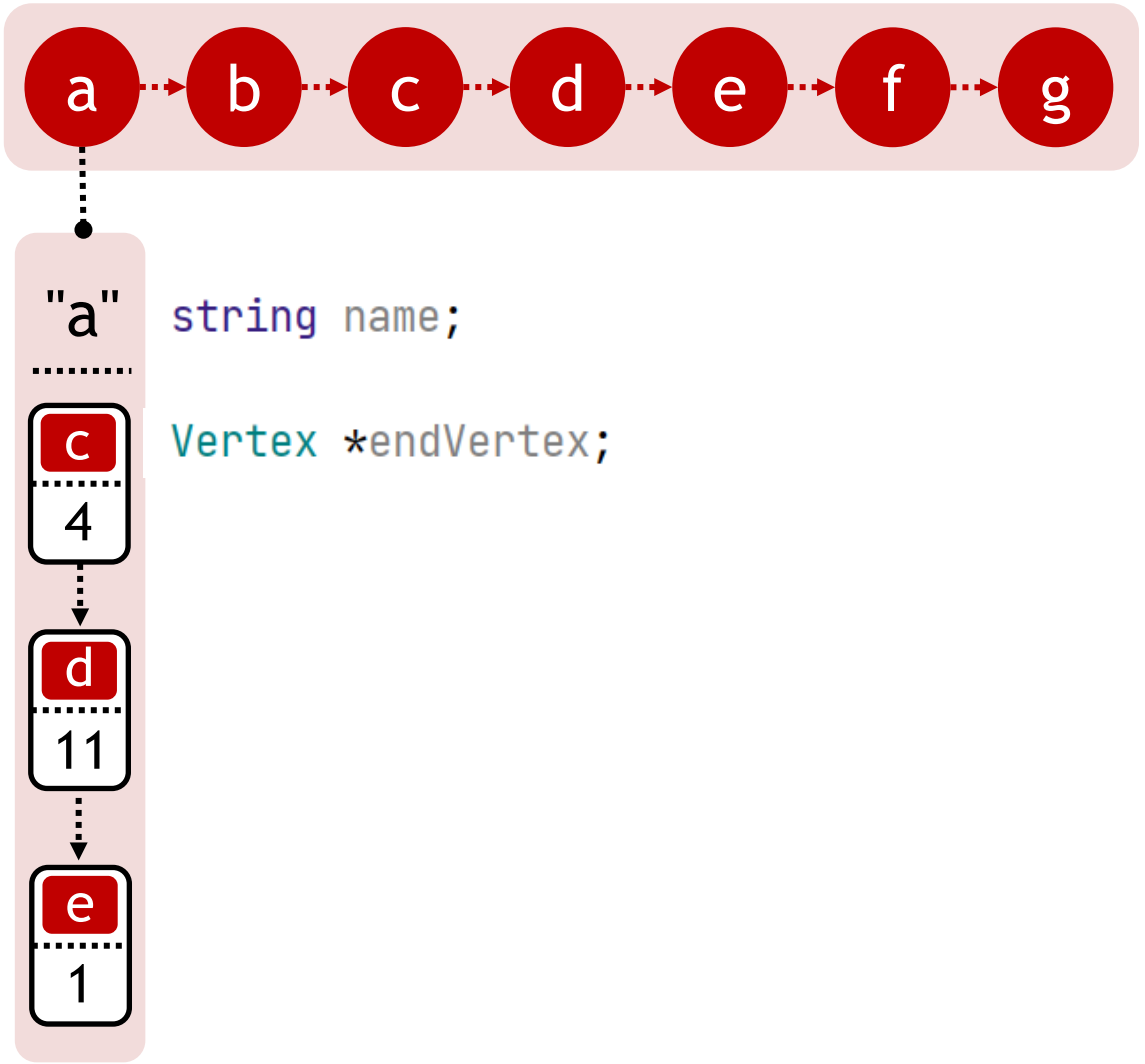


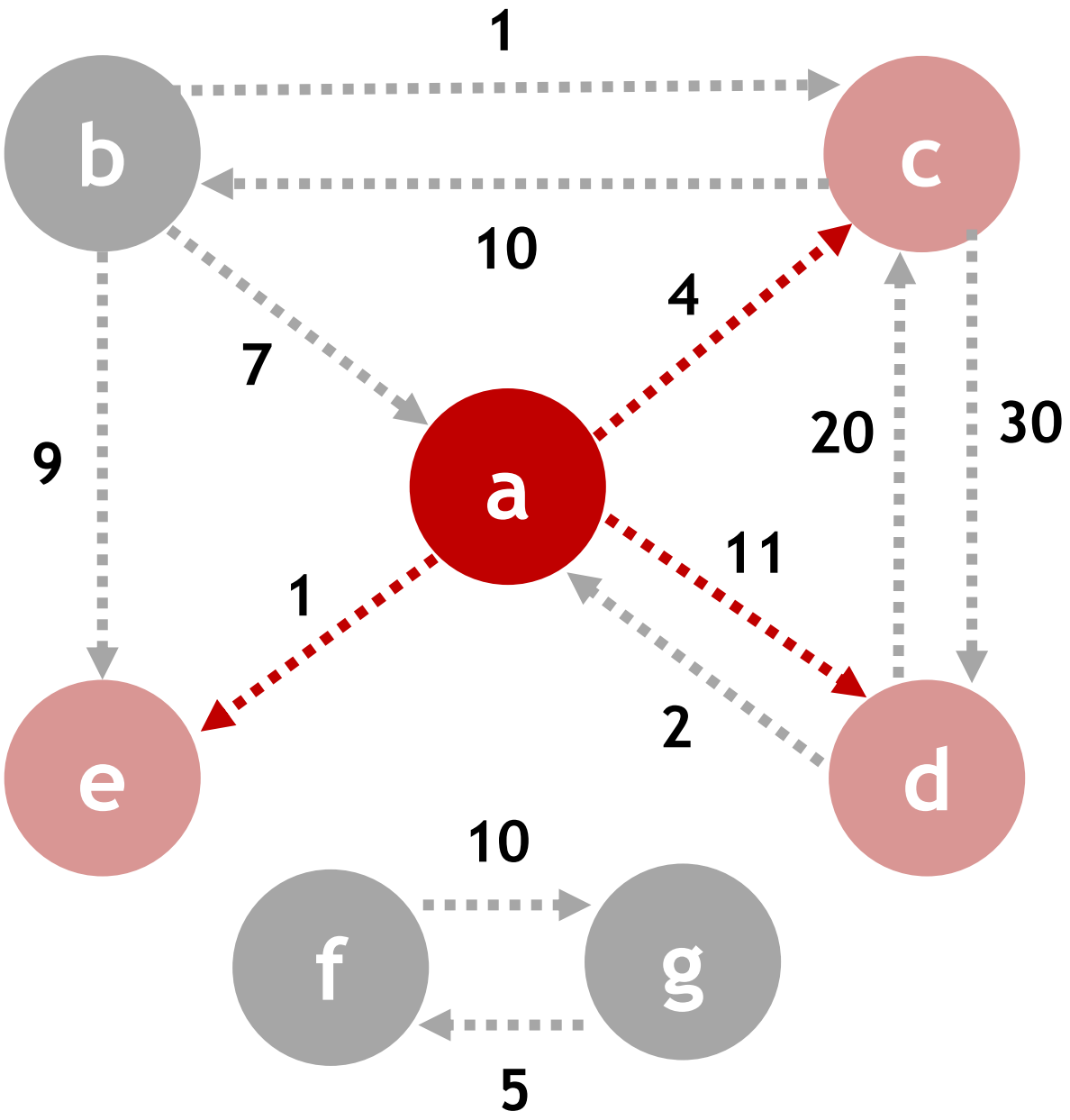

```
list<Vertex> vertices;
```



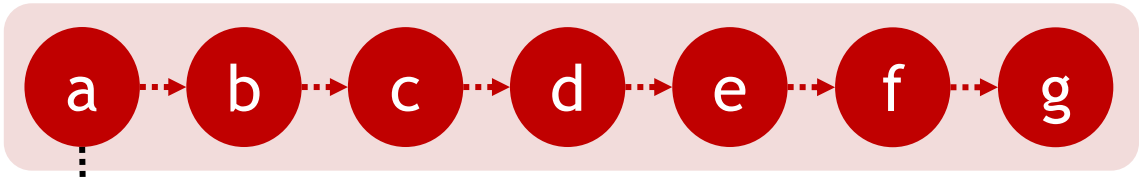


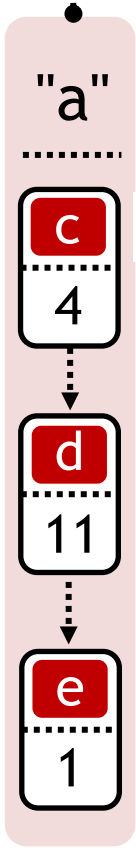

```
list<Vertex> vertices;
```



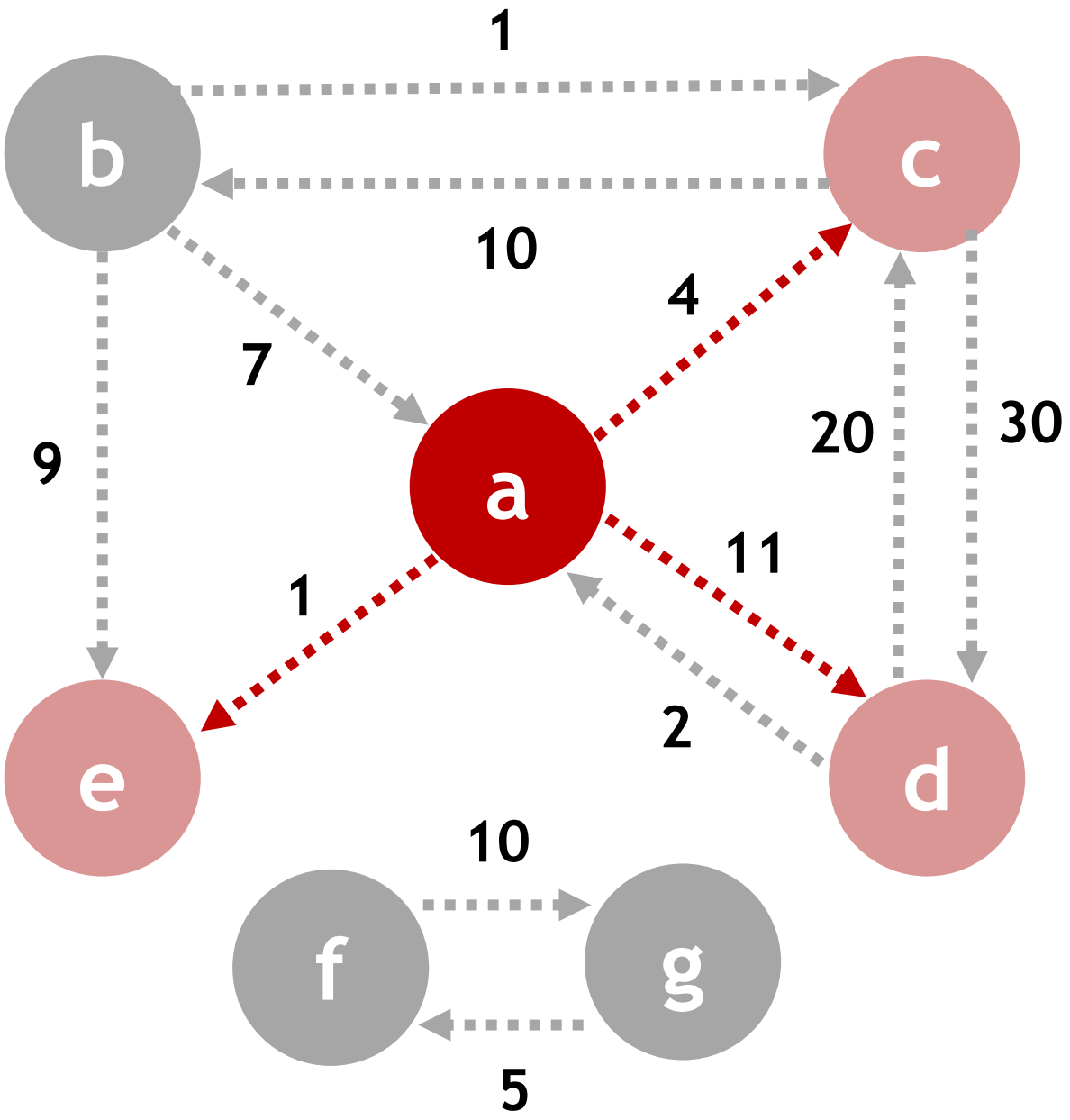



```
list<Vertex> vertices;
```

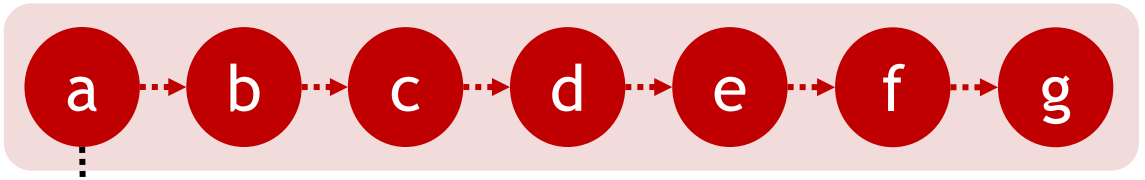


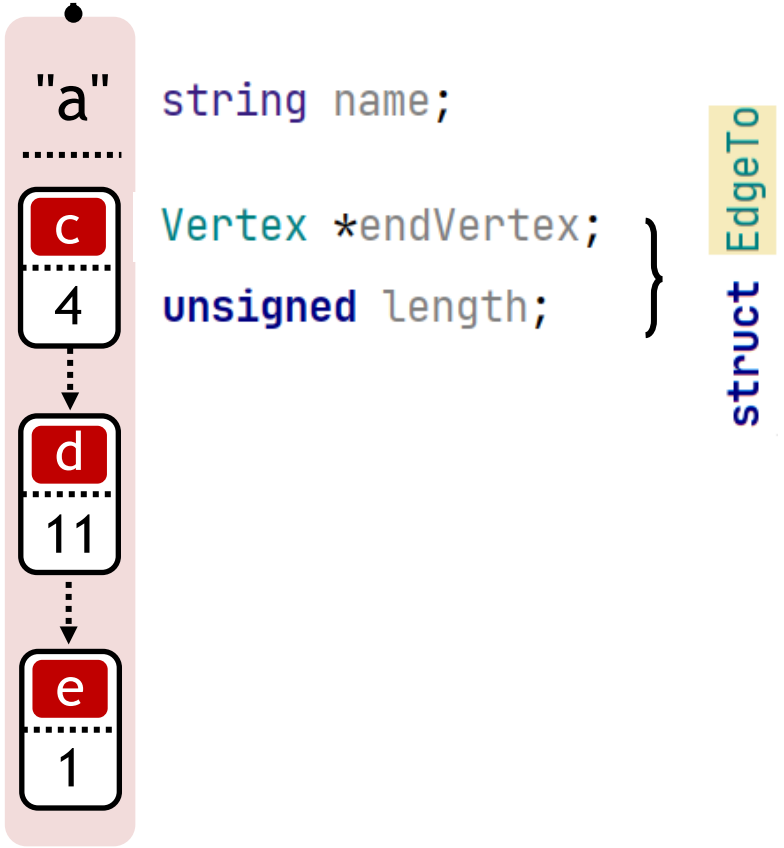


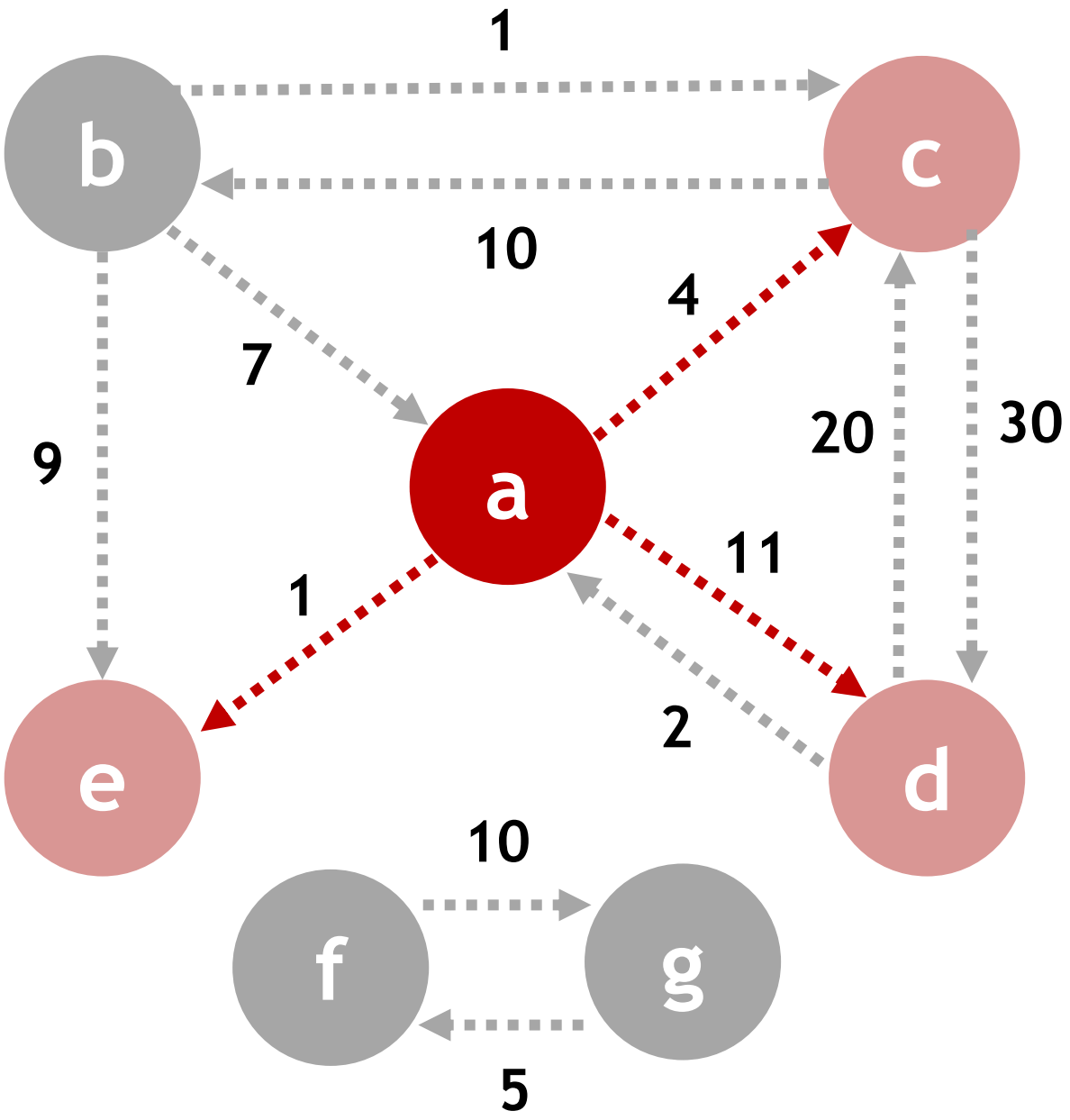

```
string name;

Vertex *endVertex;
unsigned length;
```

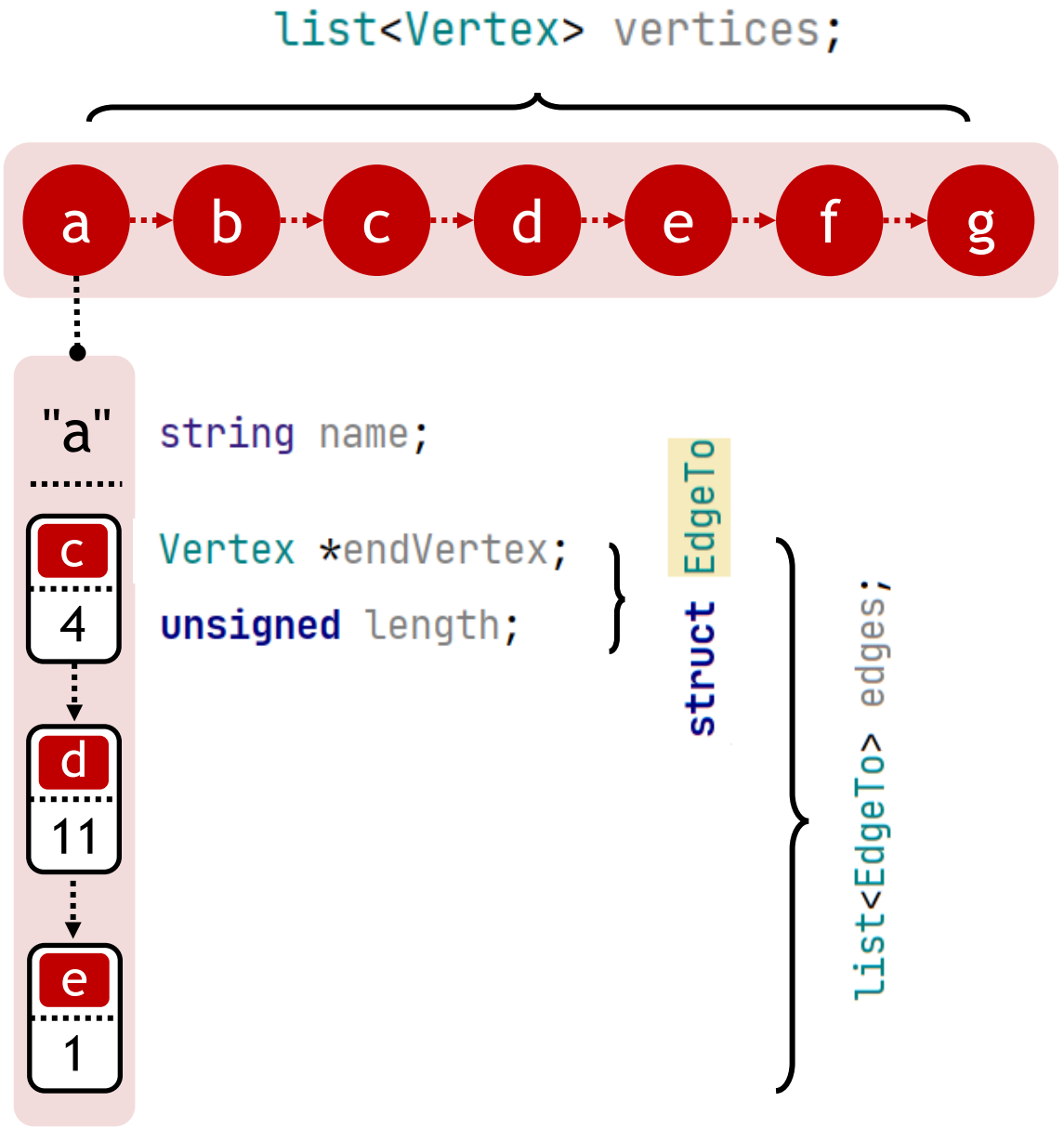


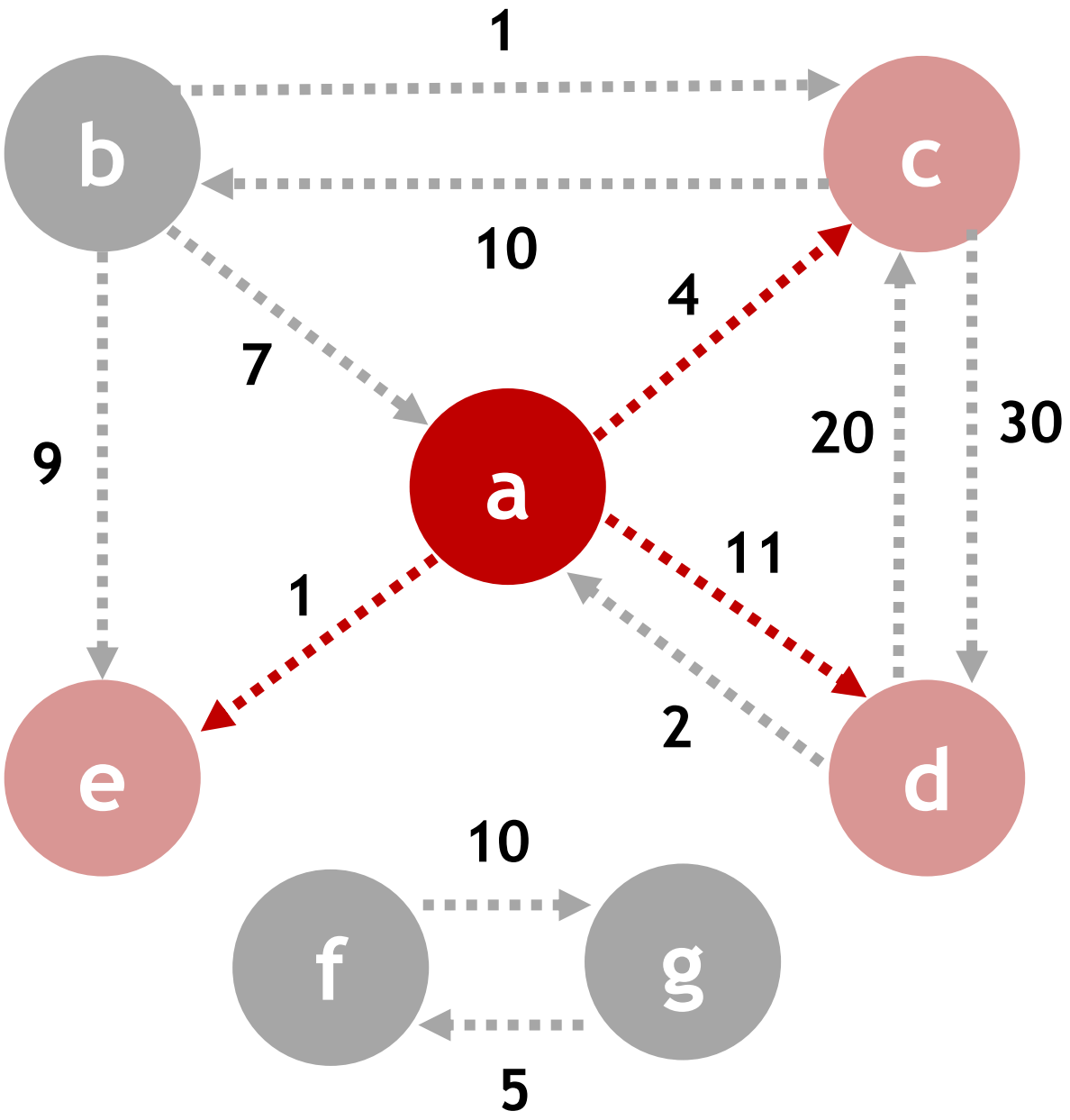

```
list<Vertex> vertices;
```

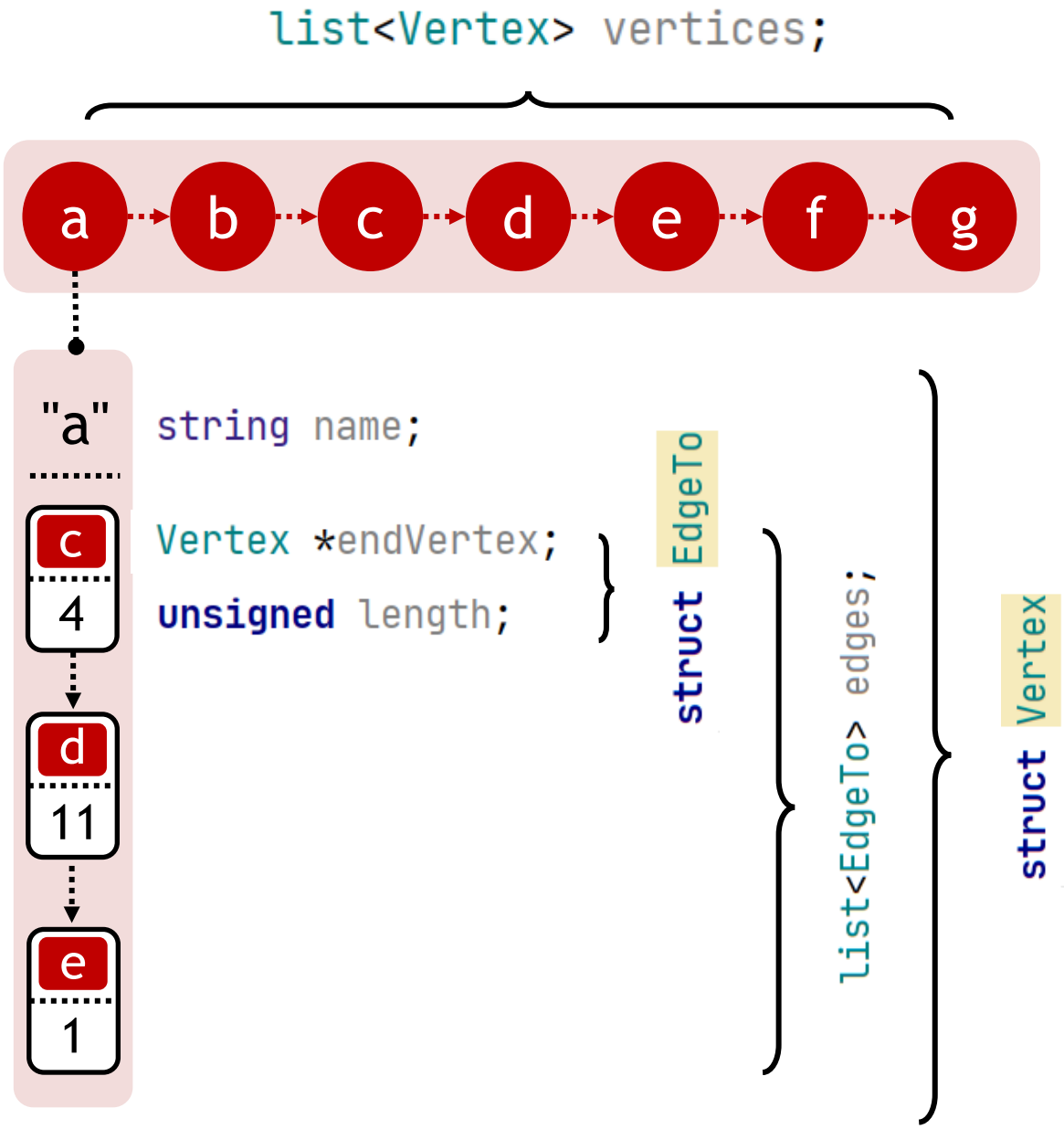


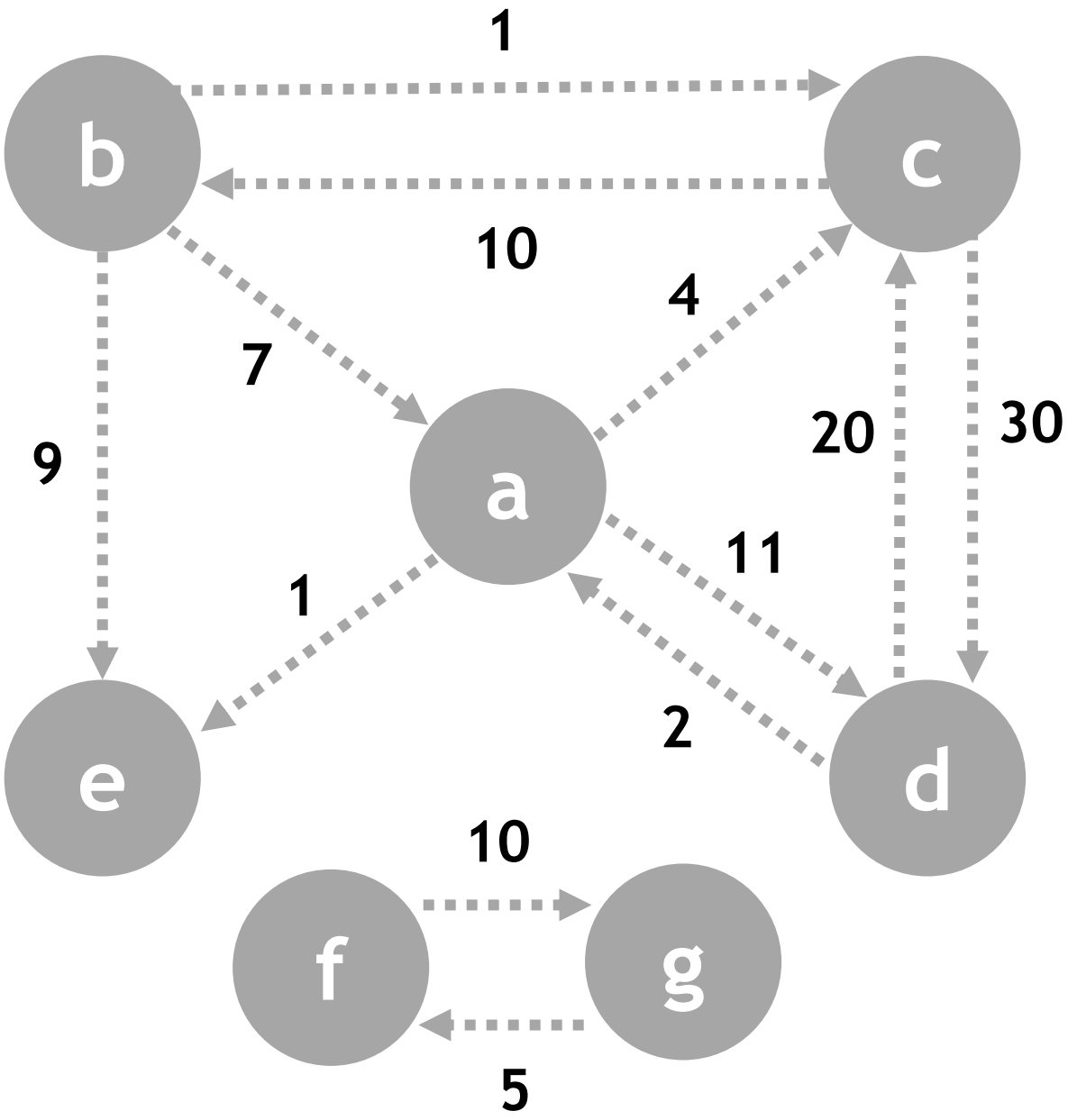



1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5

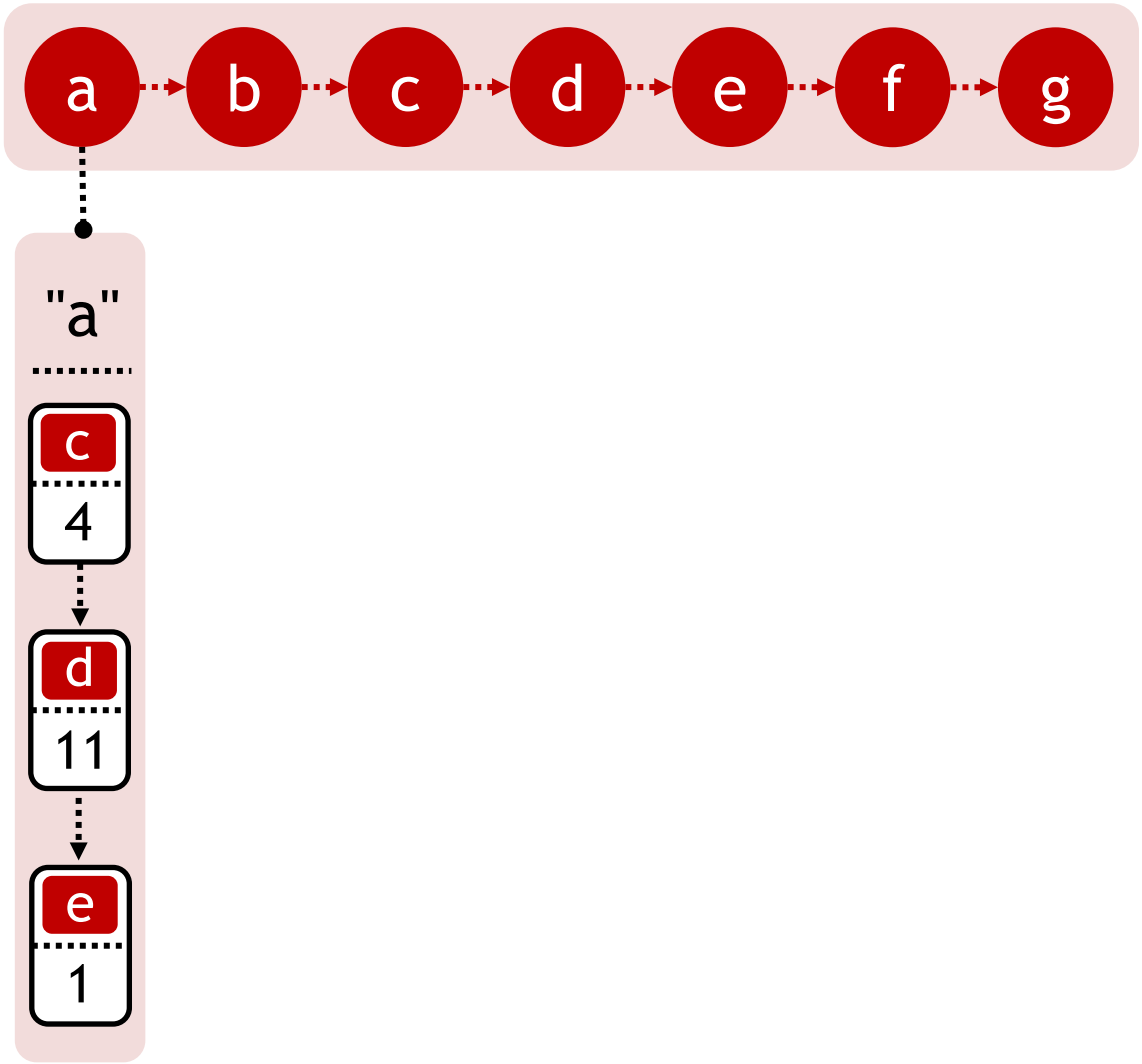

list<Vertex> vertices;
a
b
c
d
e
f
g
"a"
string name;
c
4
Vertex *endVertex;
unsigned length;
struct EdgeTo
d
11
e
1
list<EdgeTo> edges;

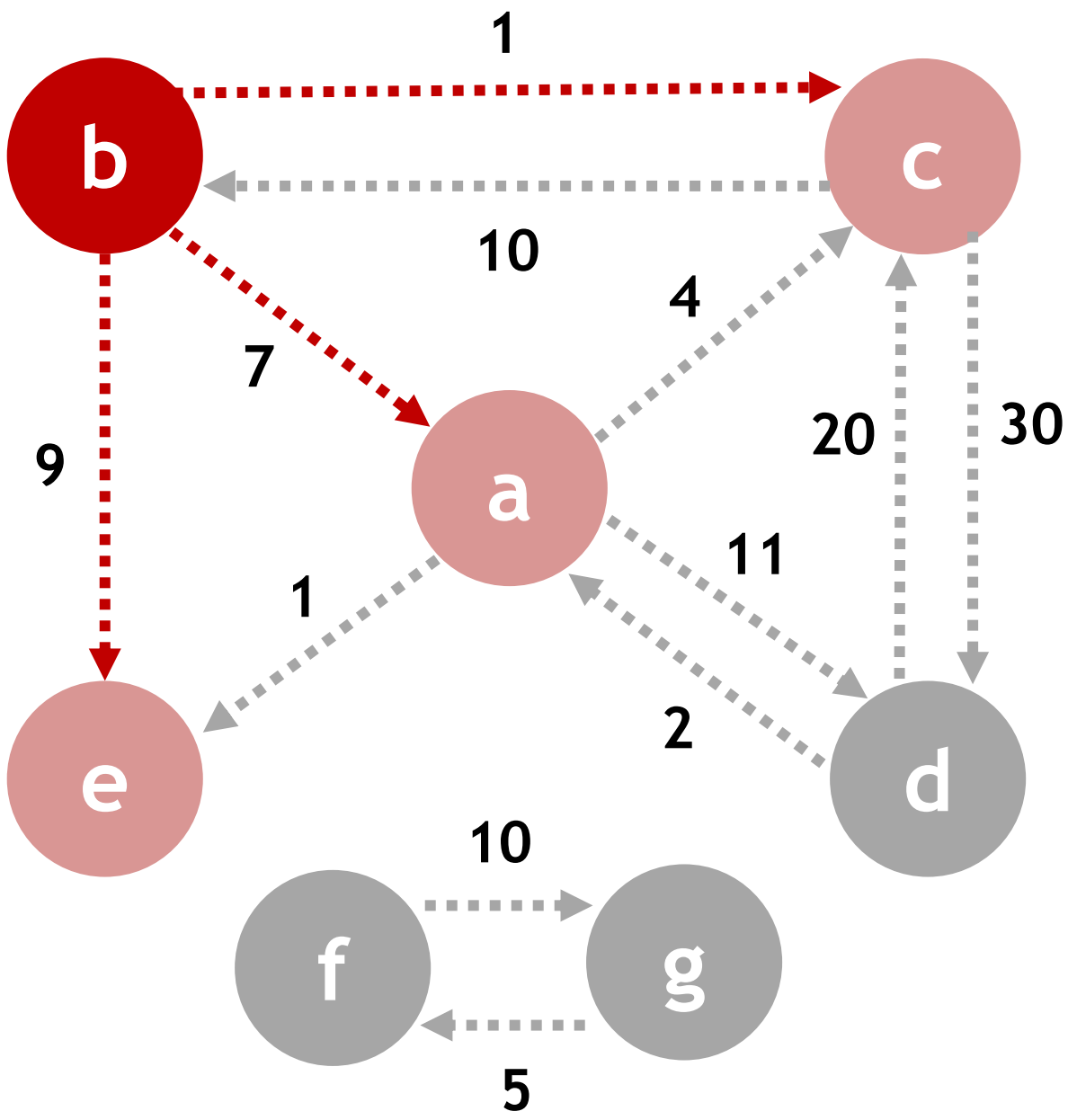

1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5

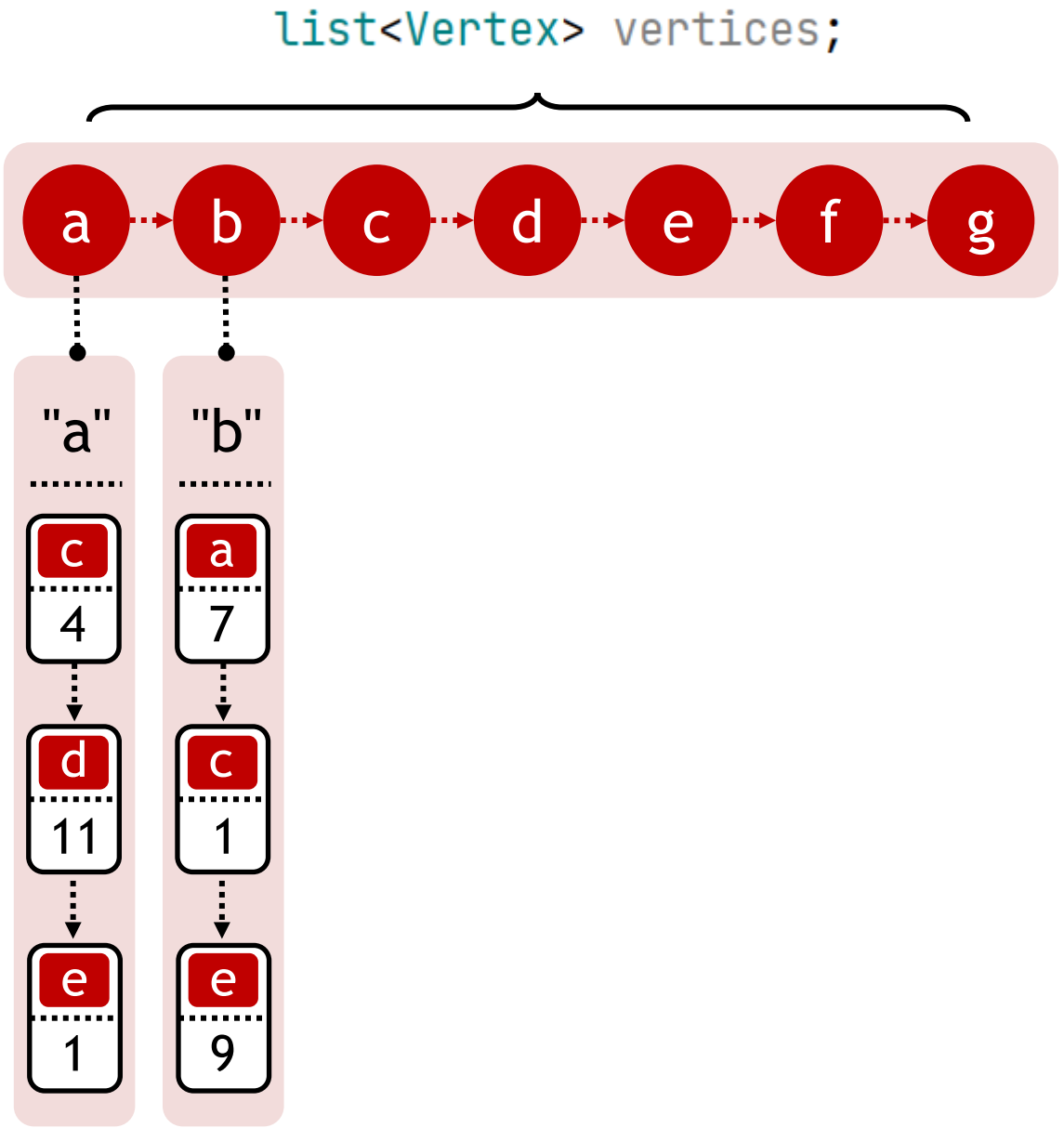

list<Vertex> vertices;
a
b
c
d
e
f
g
"a"
string name;
c
4
Vertex *endVertex;
unsigned length;
d
11
e
1
struct EdgeTo
list<EdgeTo> edges;
struct Vertex

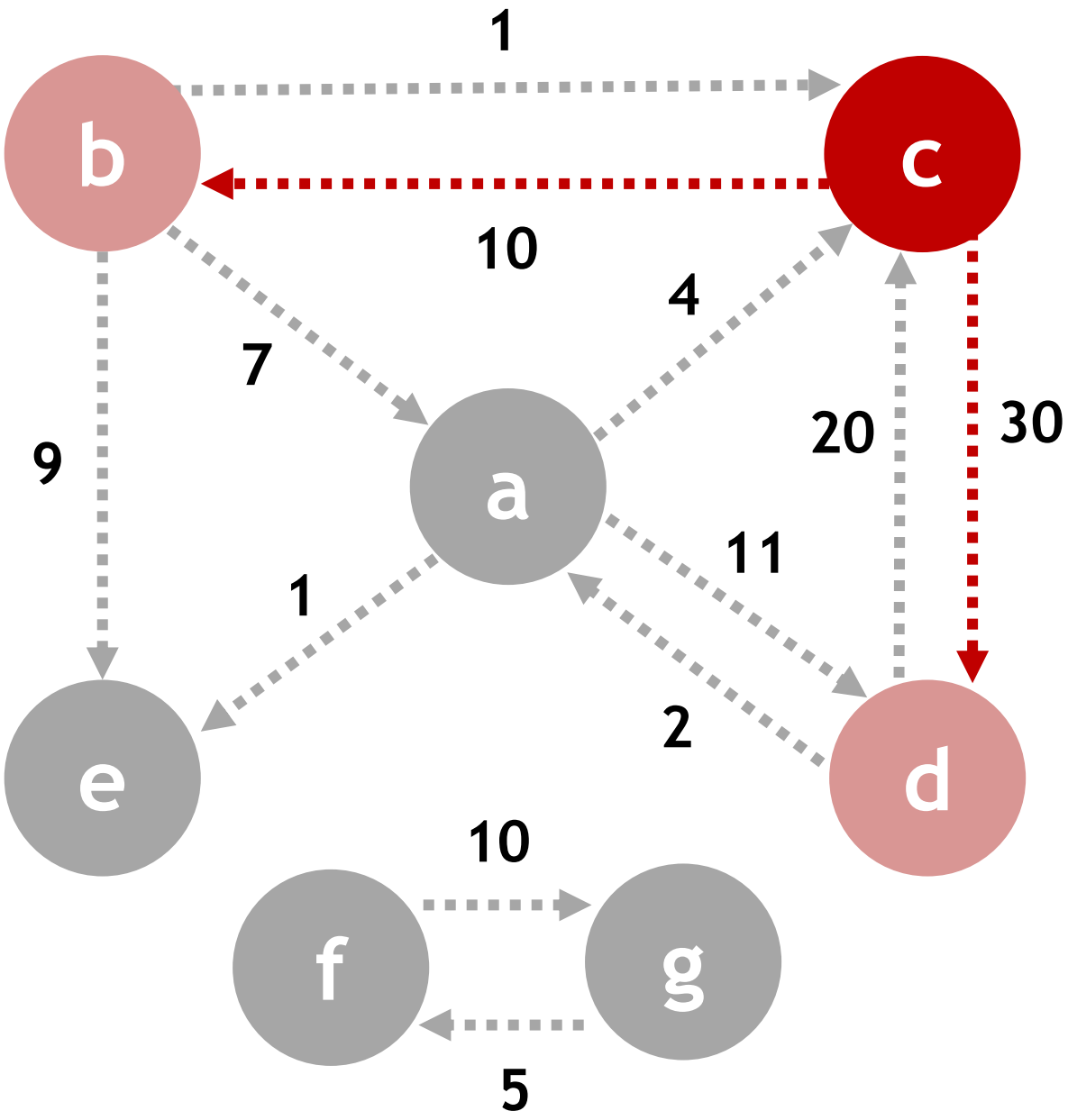


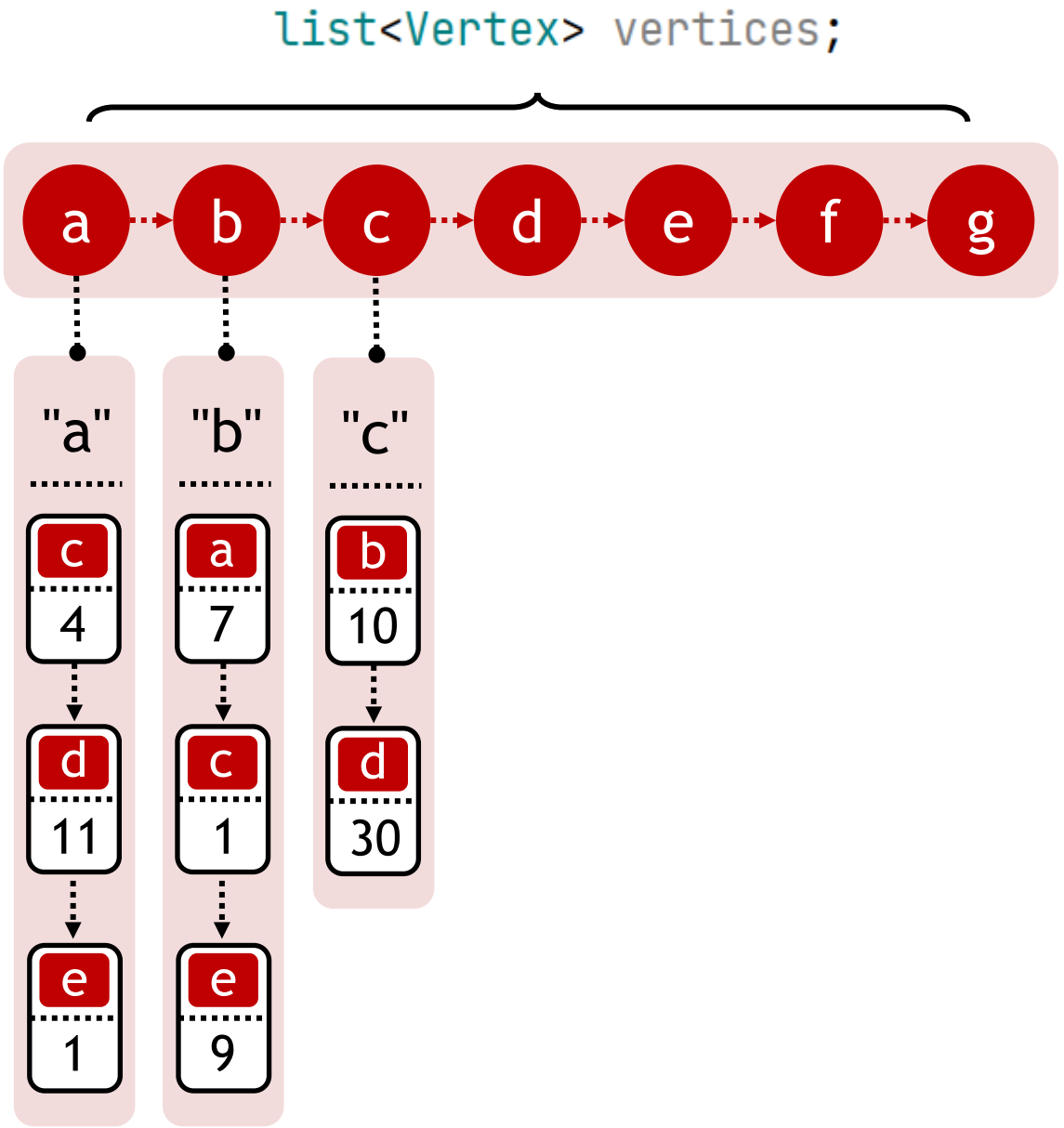

```
list<Vertex> vertices;
```

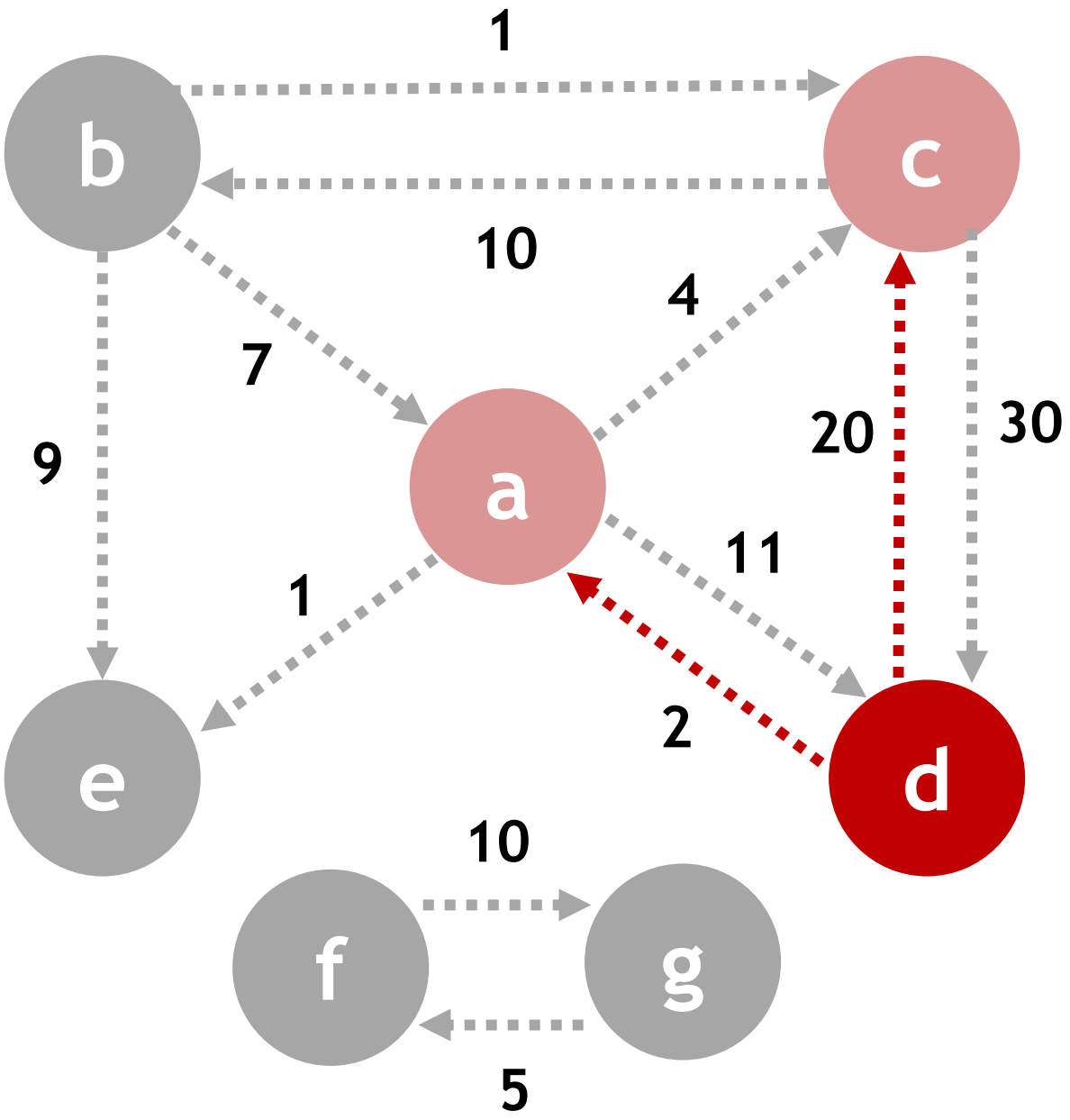


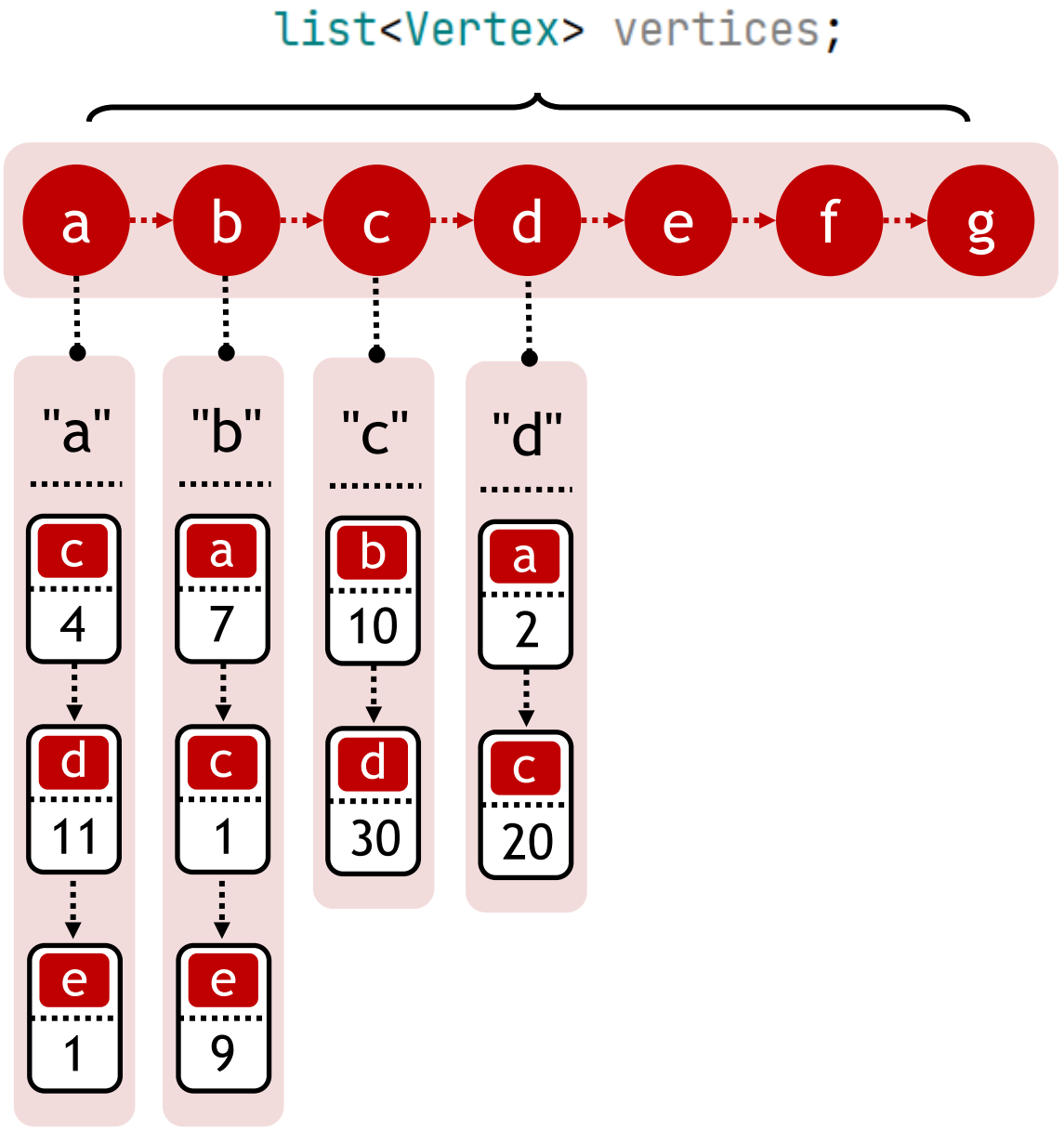

1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5

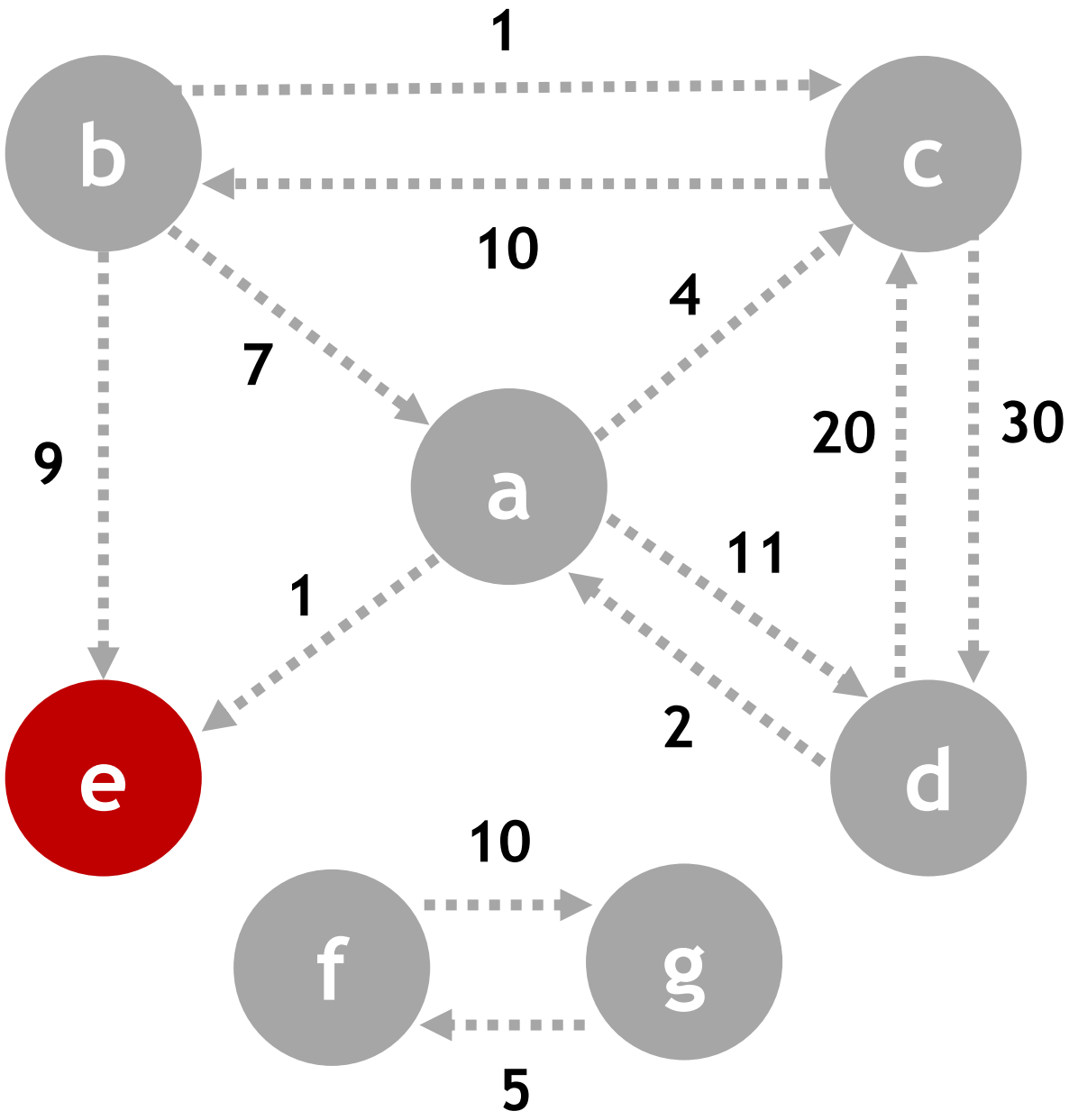

list<Vertex> vertices;
a
b
c
d
e
f
g
"a"
"b"
c
4
d
11
e
1
a
7
c
1
e
9


1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5

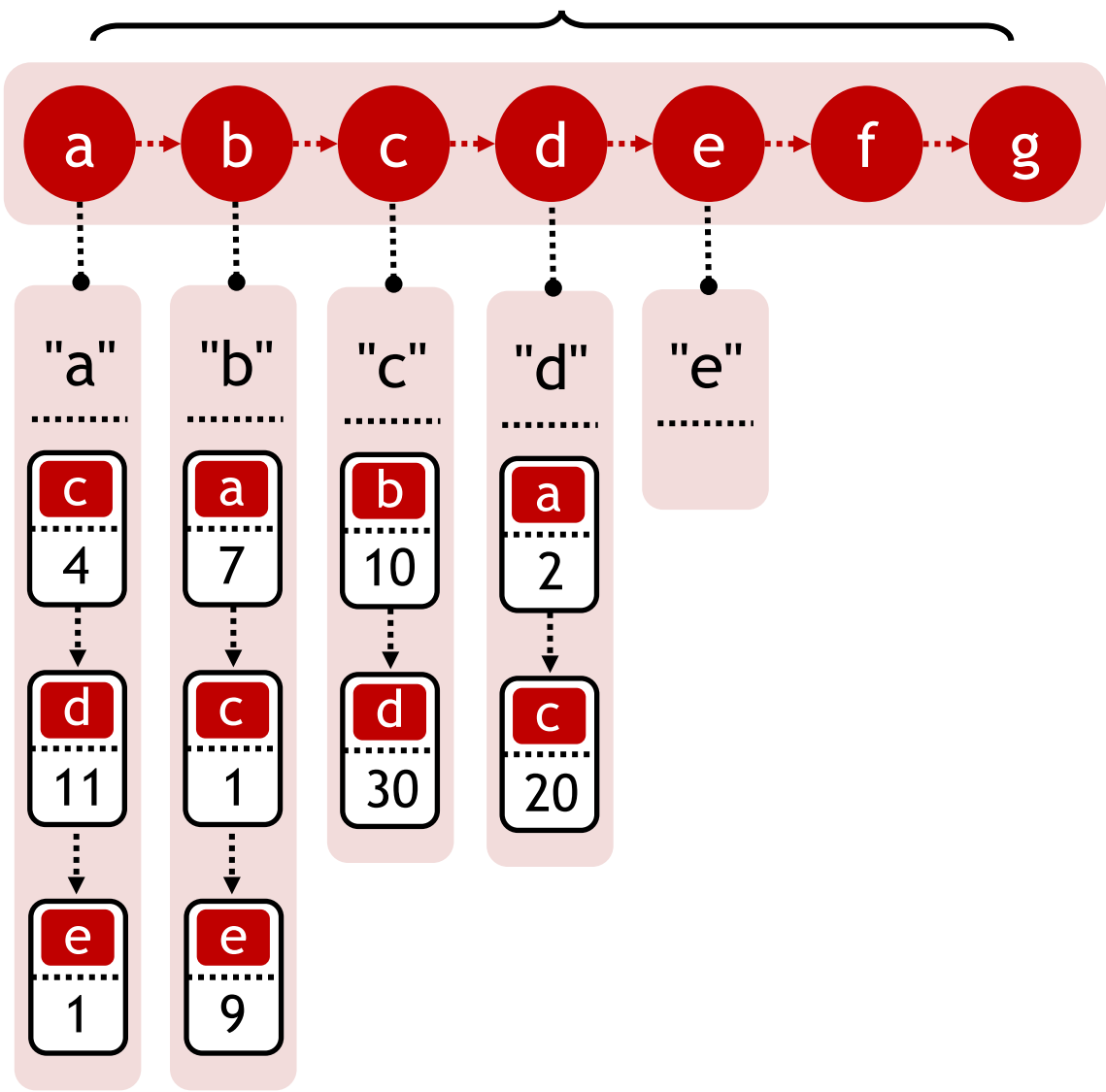

list<Vertex> vertices;
a
b
c
d
e
f
g
"a"
"b"
"c"
c
4
d
11
e
1
a
7
c
1
e
9
b
10
d
30

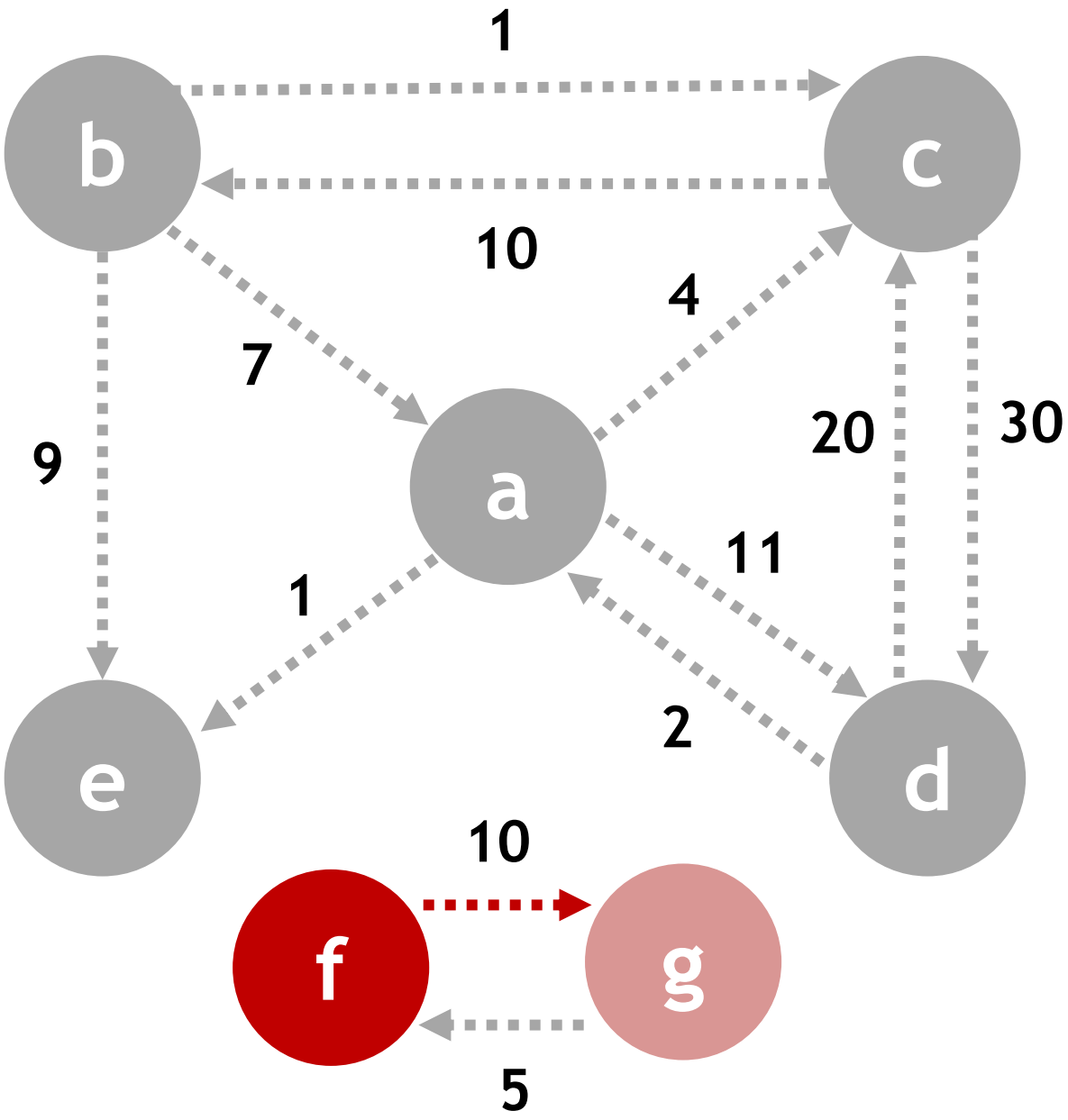

1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5


list<Vertex> vertices;
a
b
c
d
e
f
g
"a"
"b"
"c"
"d"
c
4
d
11
e
1
a
7
c
1
e
9
b
10
d
30
a
2
c
20

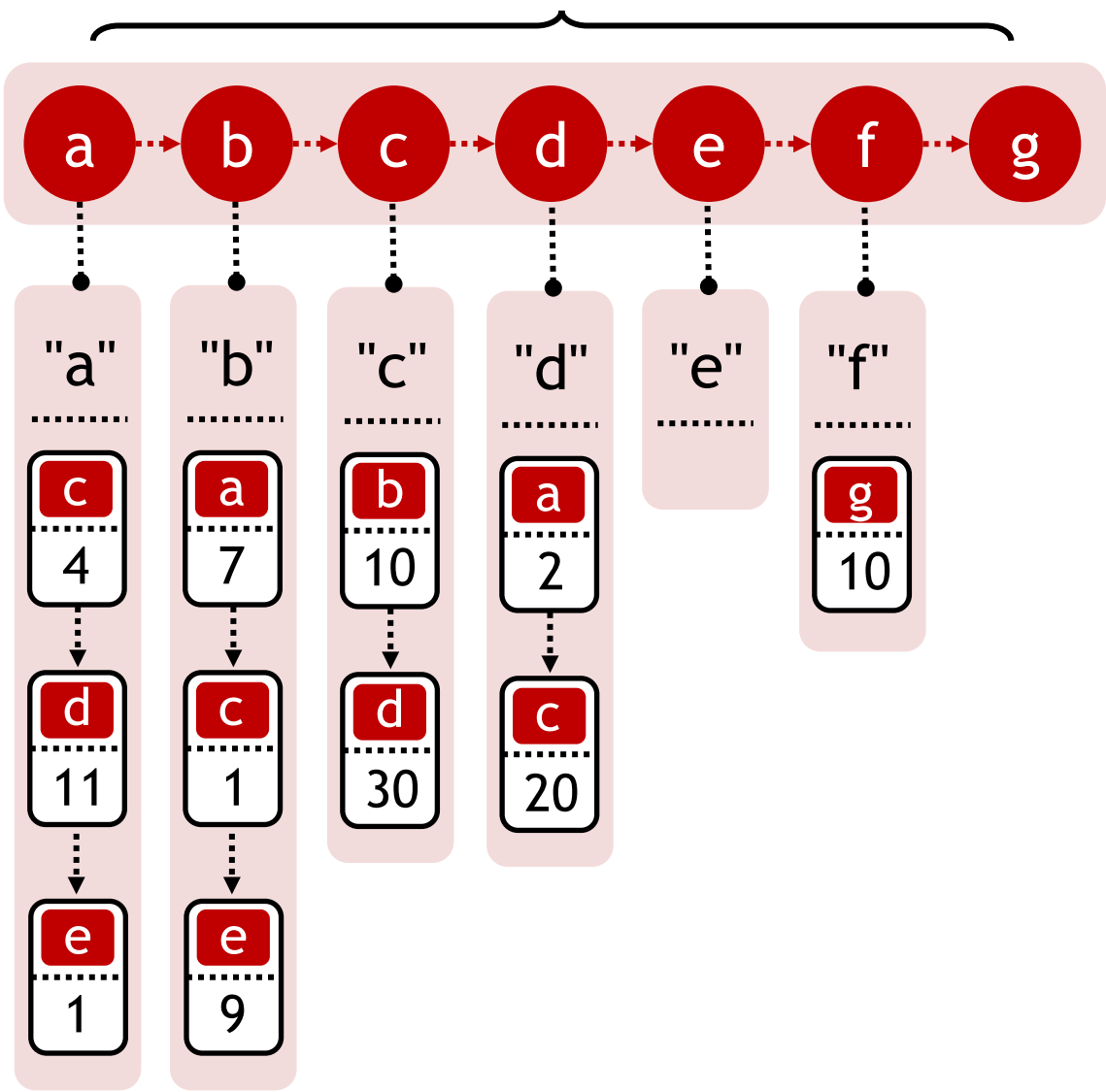


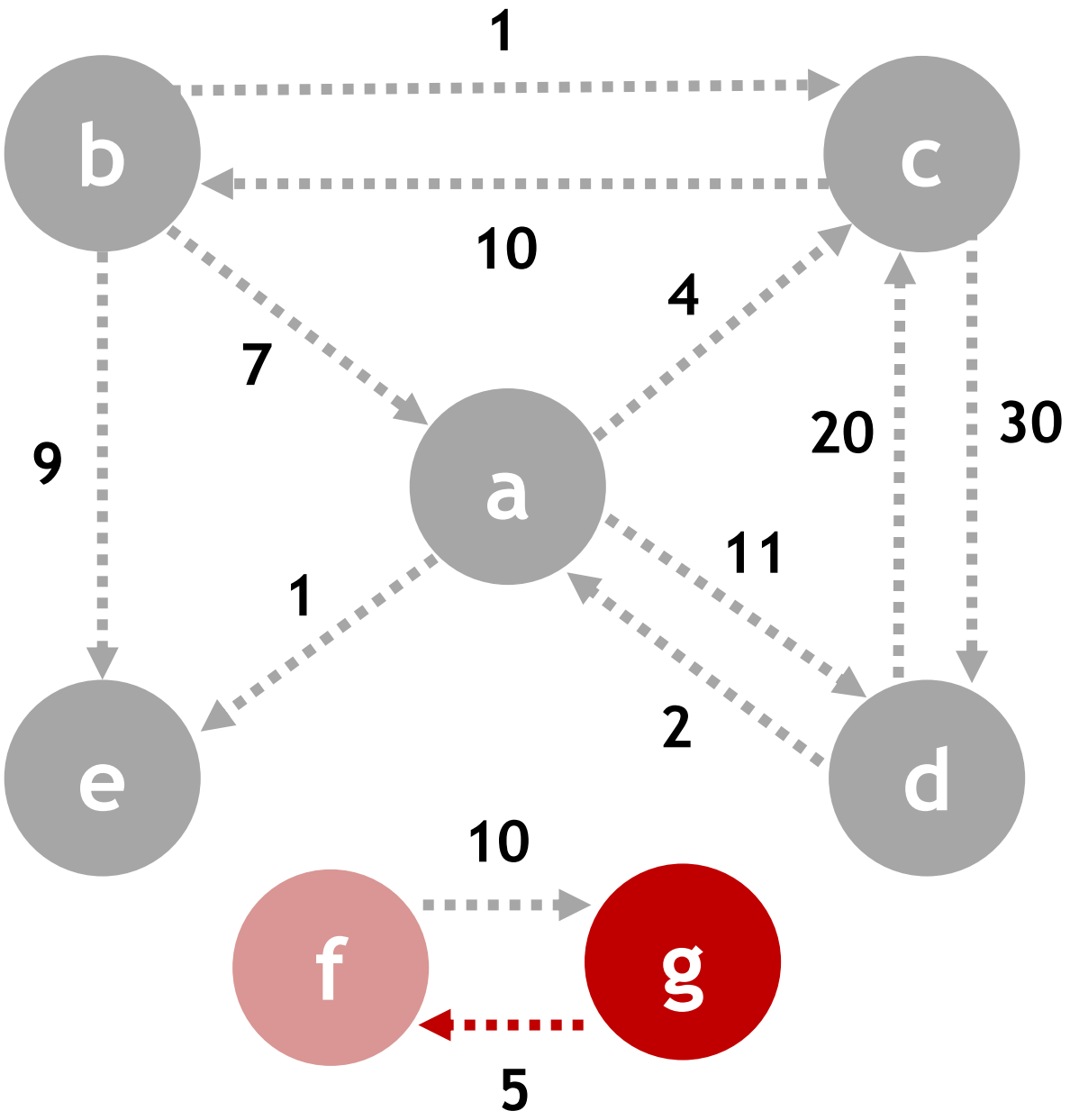

```
list<Vertex> vertices;
```



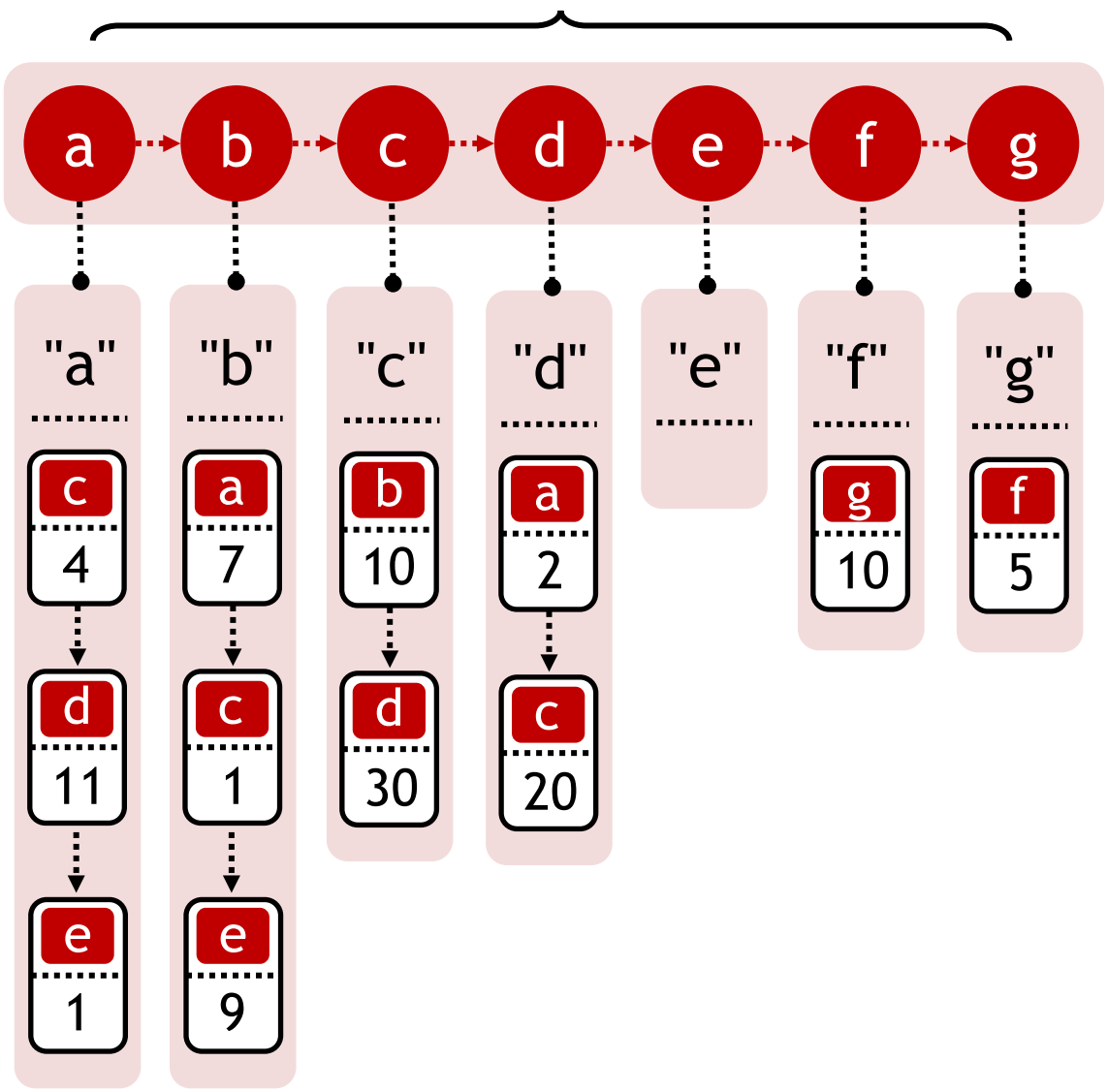



```
list<Vertex> vertices;
```






```
list<Vertex> vertices;
```

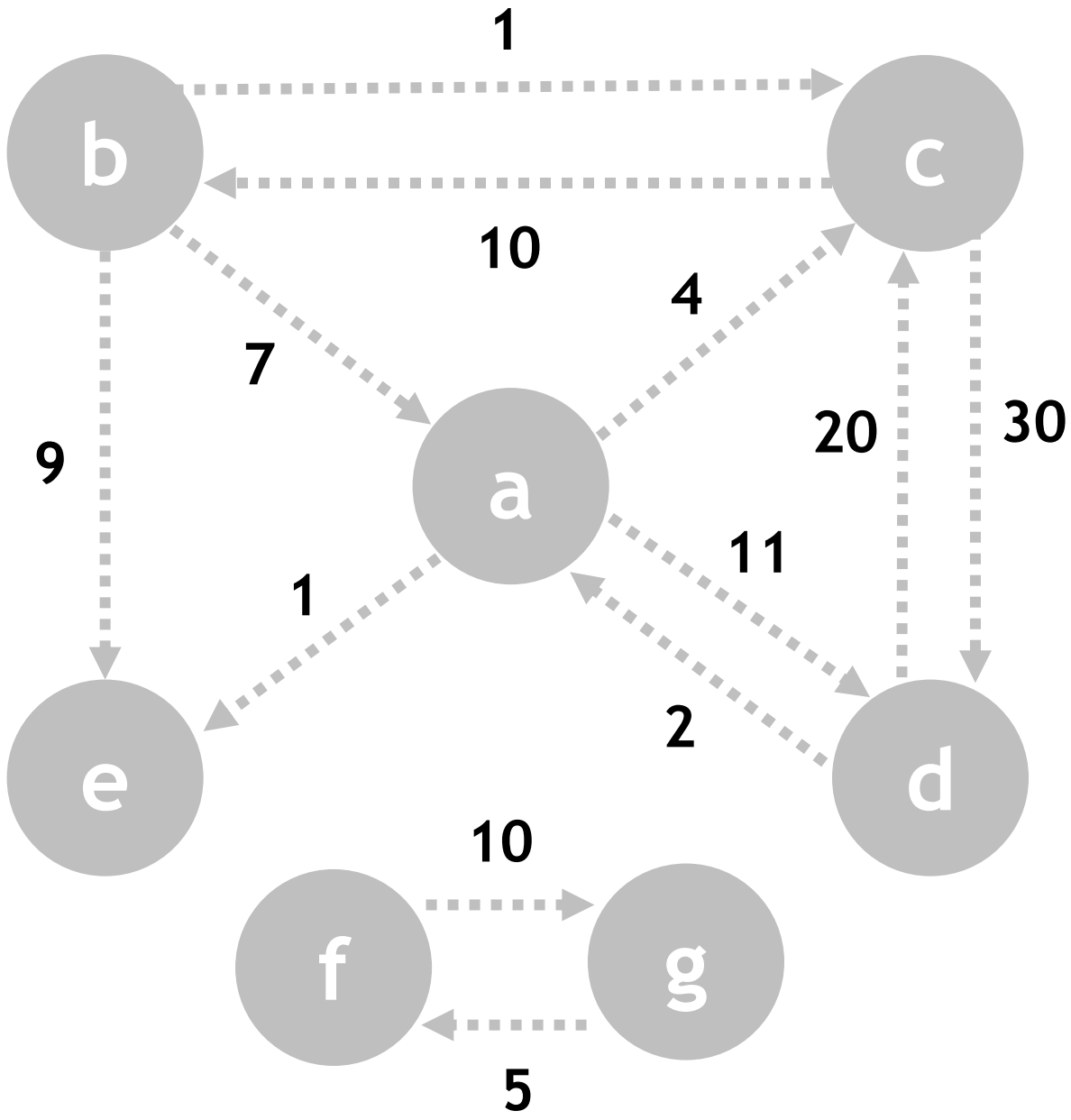


Prehľadávanie do hĺbky

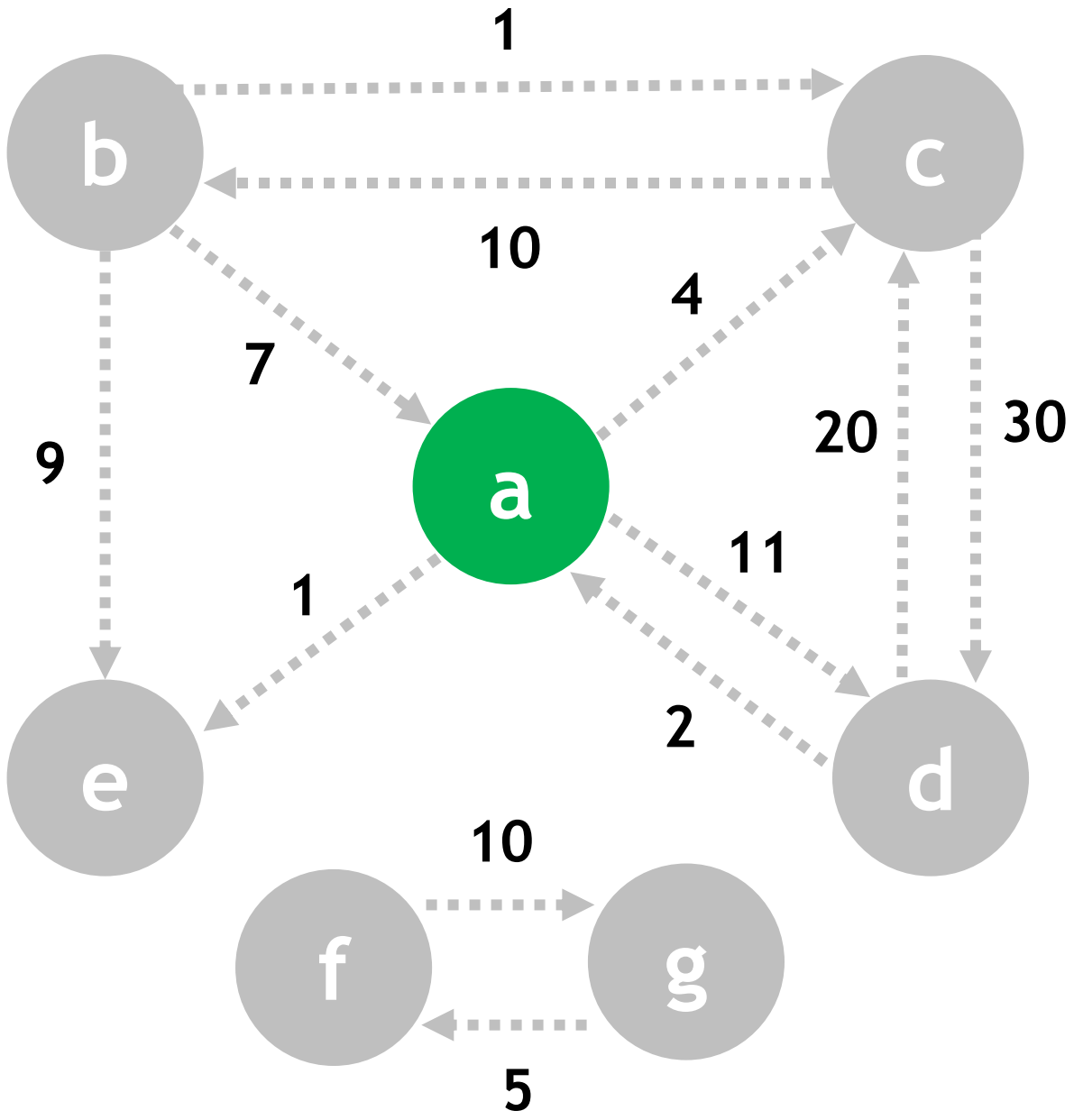
Pseudocode

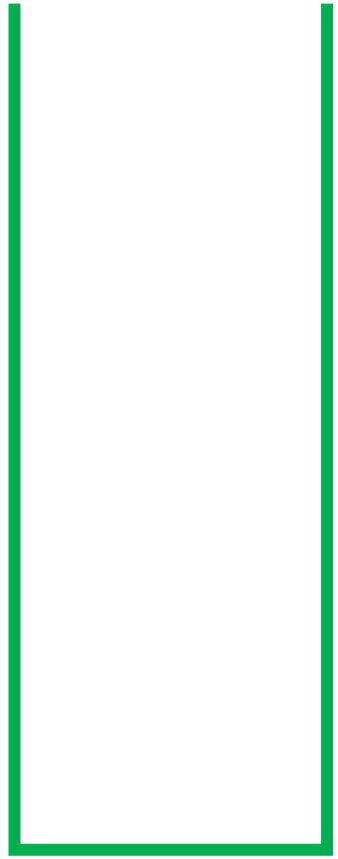
```
DFS(G,v)   ( v is the vertex where the search starts )
      Stack S := {};   ( start with an empty stack )
      for each vertex u, set visited[u] := false;
      push S, v;
      while (S is not empty) do
        u := pop S;
        if (not visited[u]) then
          visited[u] := true;
          for each unvisited neighbour w of uu
            push S, w;
        end if
      end while
    END DFS()
```

Zdroj: https://www.javatpoint.com/depth-first-search-algorithm


1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5

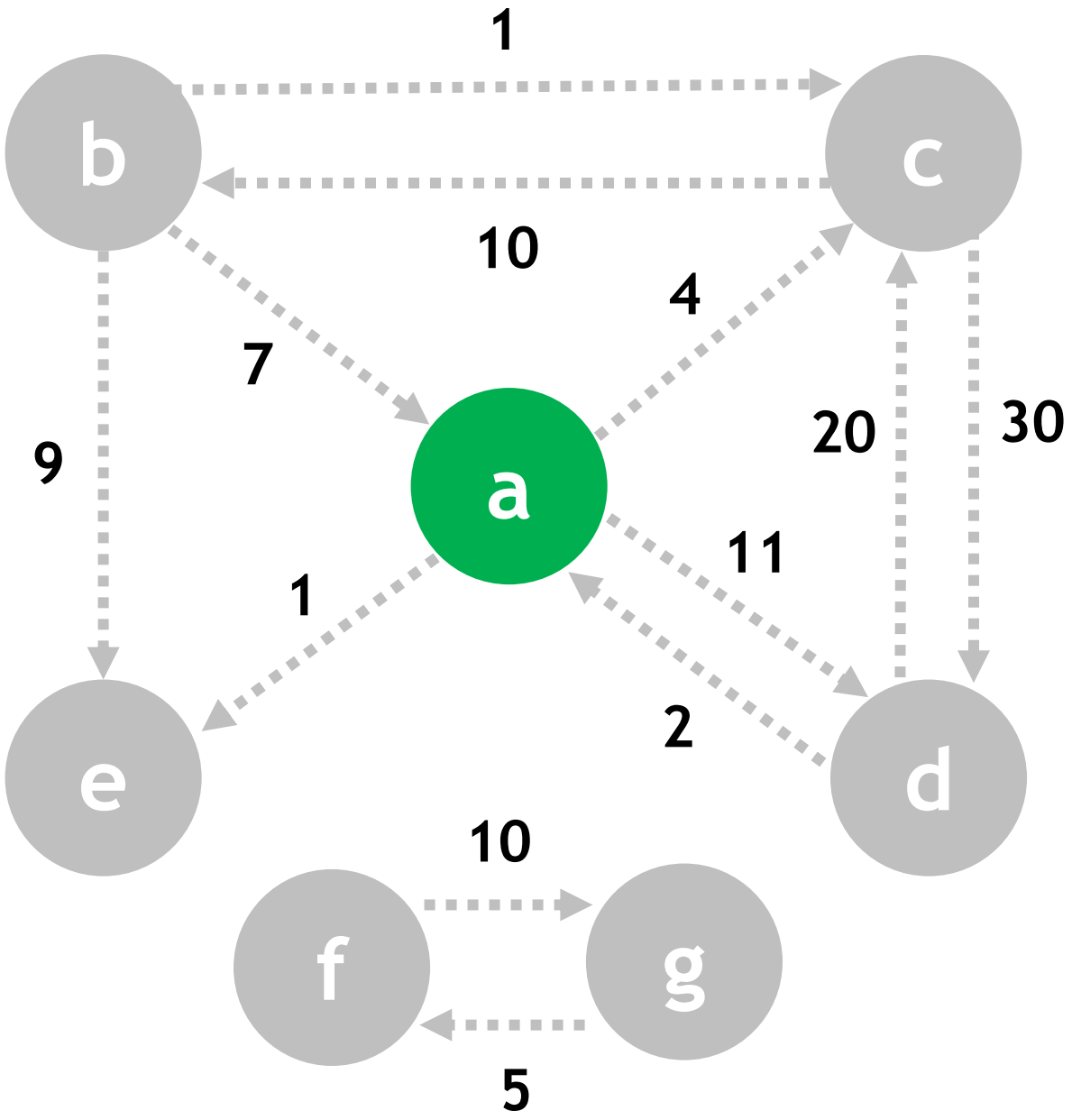
Začiatok prehľadavánia z vrcholu a

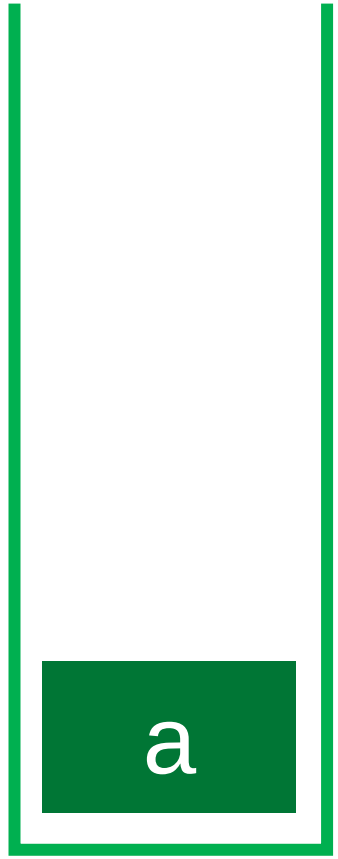




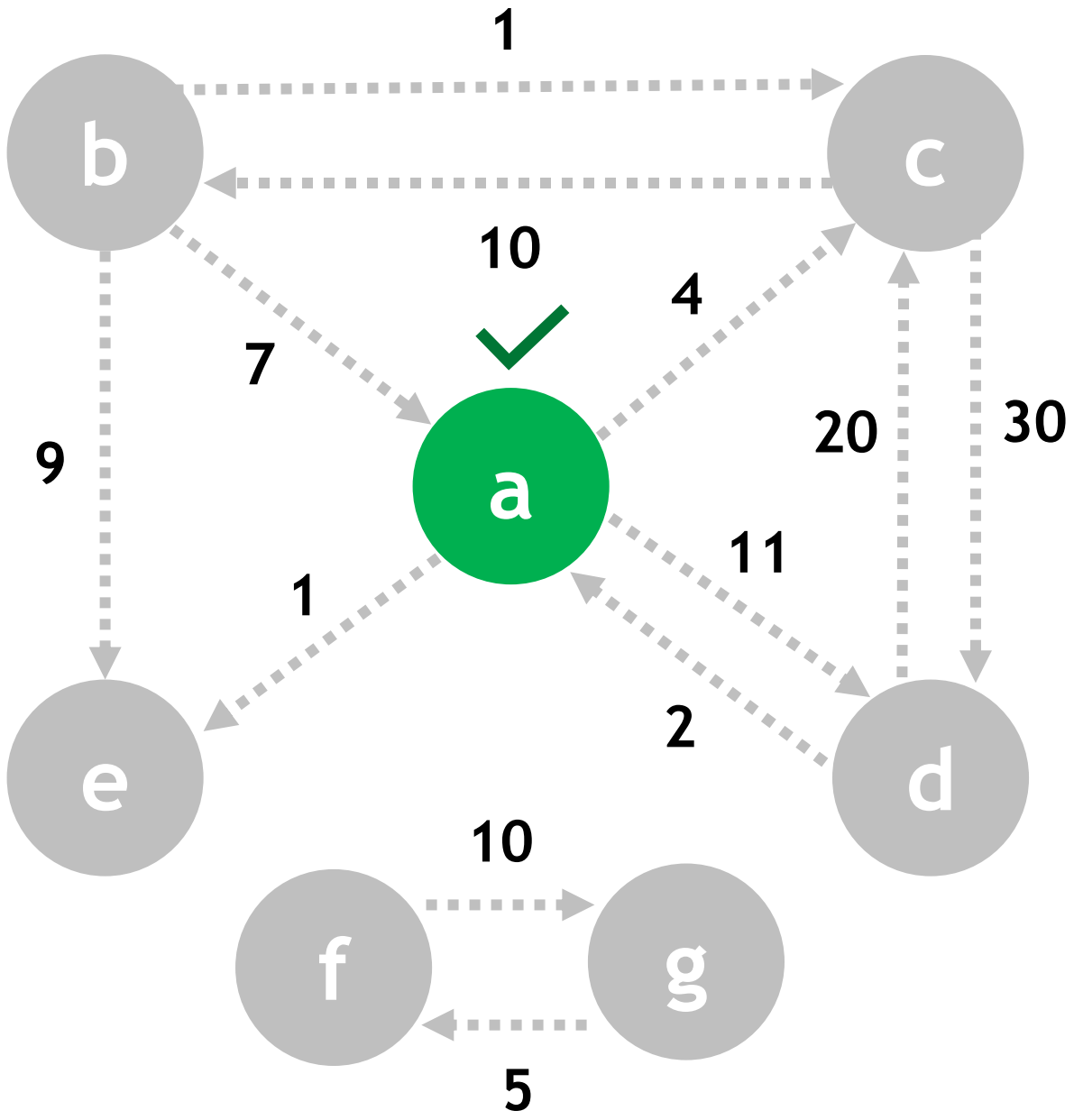
Zásobník

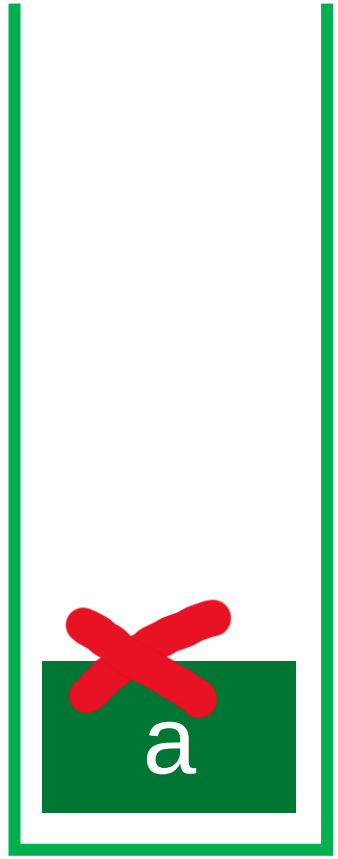
Začiatok prehľadavánia z vrcholu a





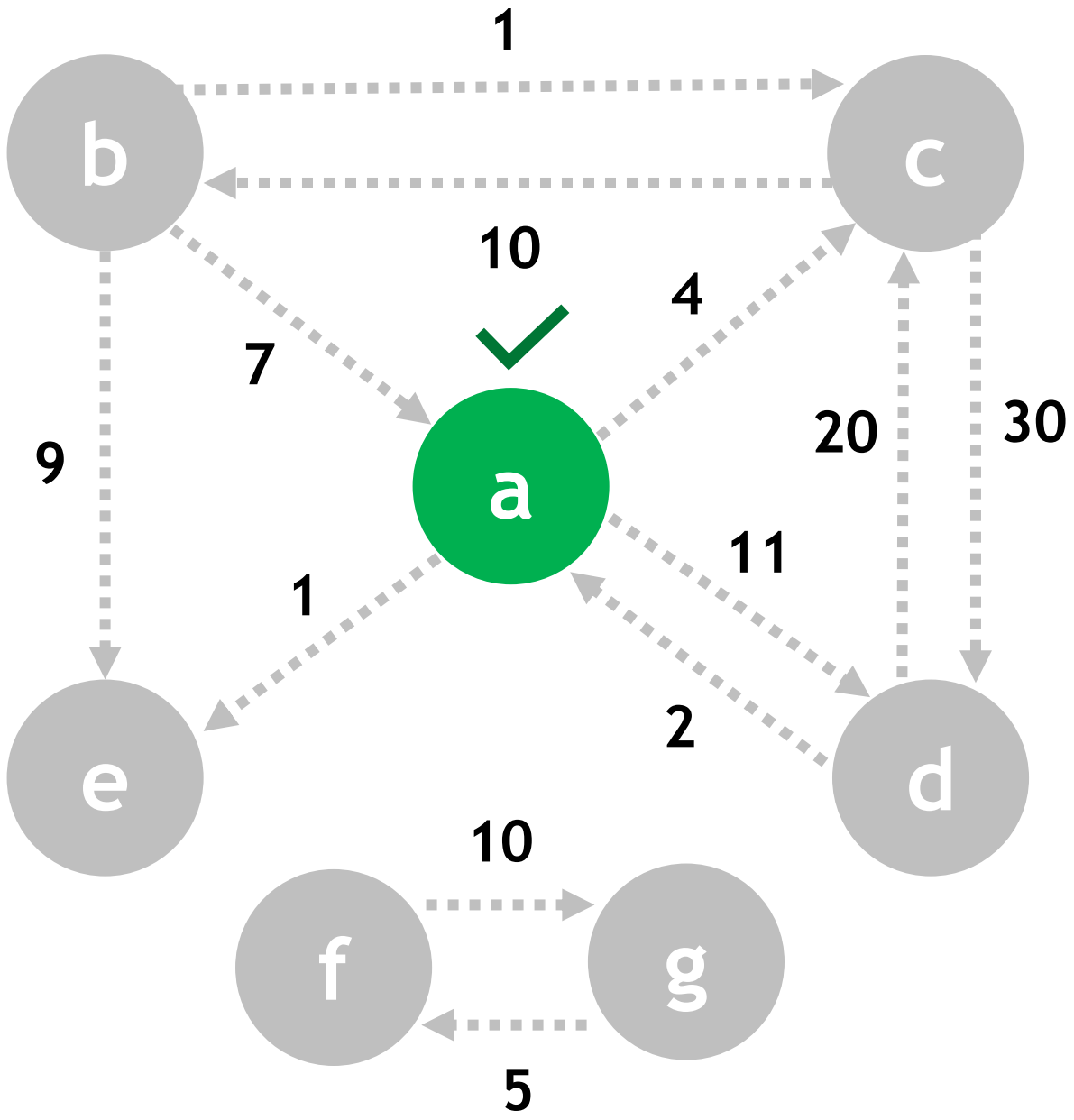
Zásobník

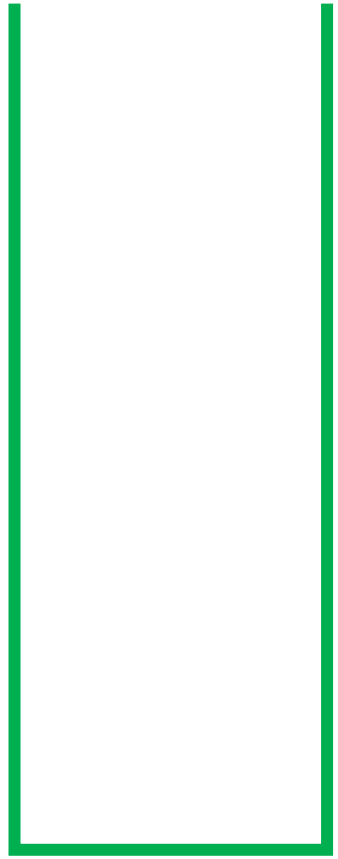




Zásobník

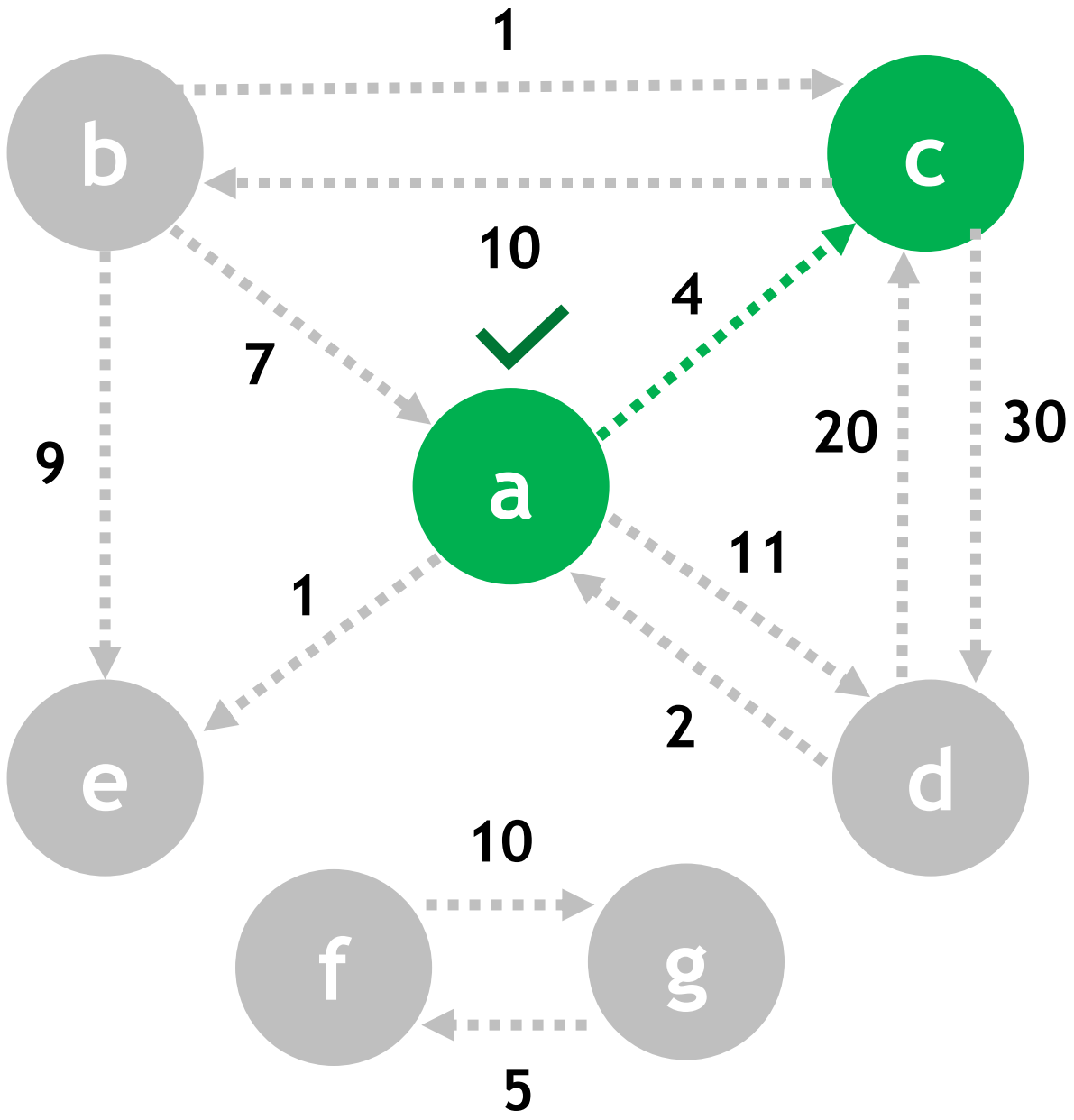
Print: a

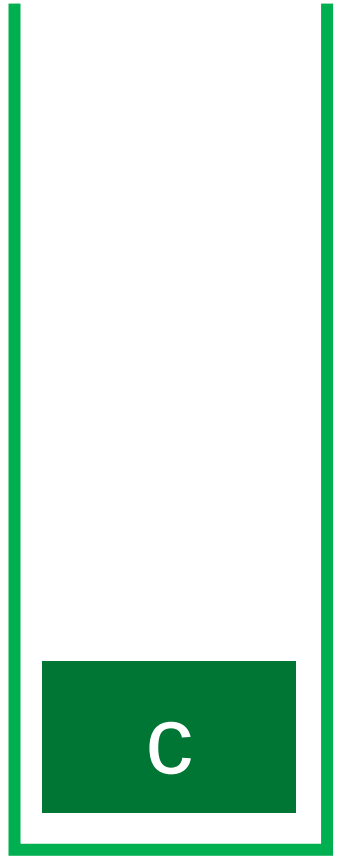




Zásobník

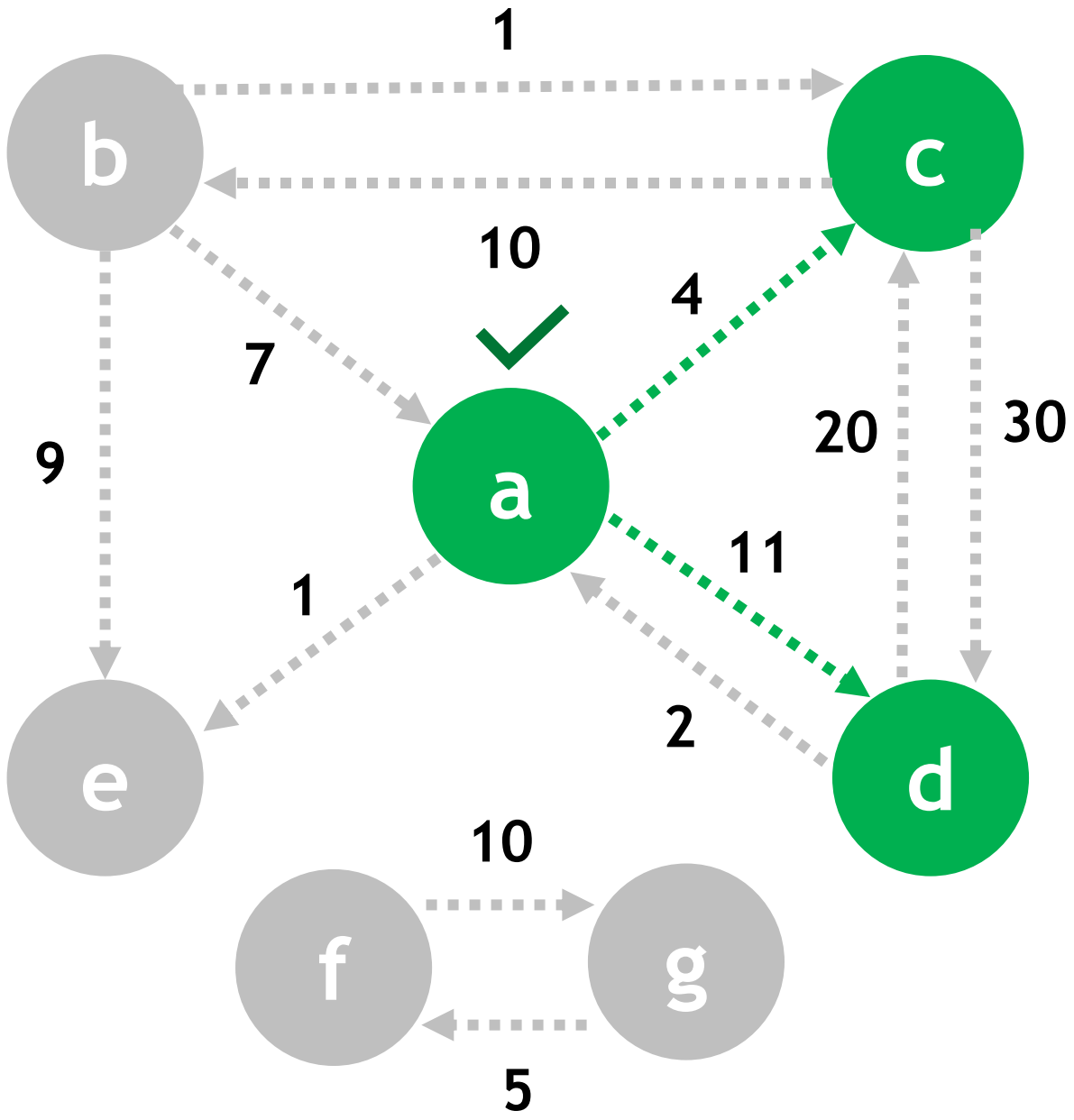
Print: a

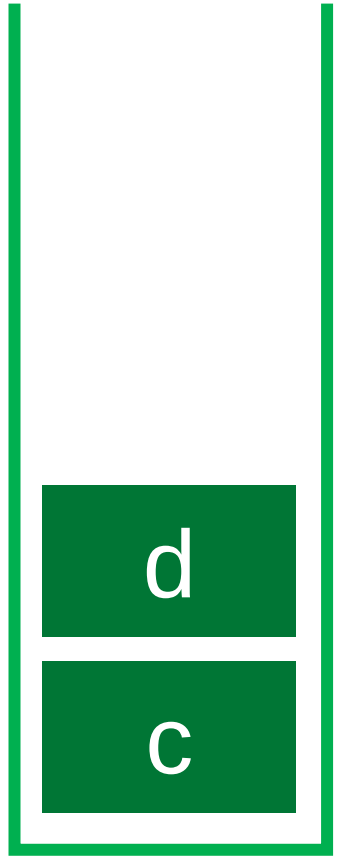




Zásobník

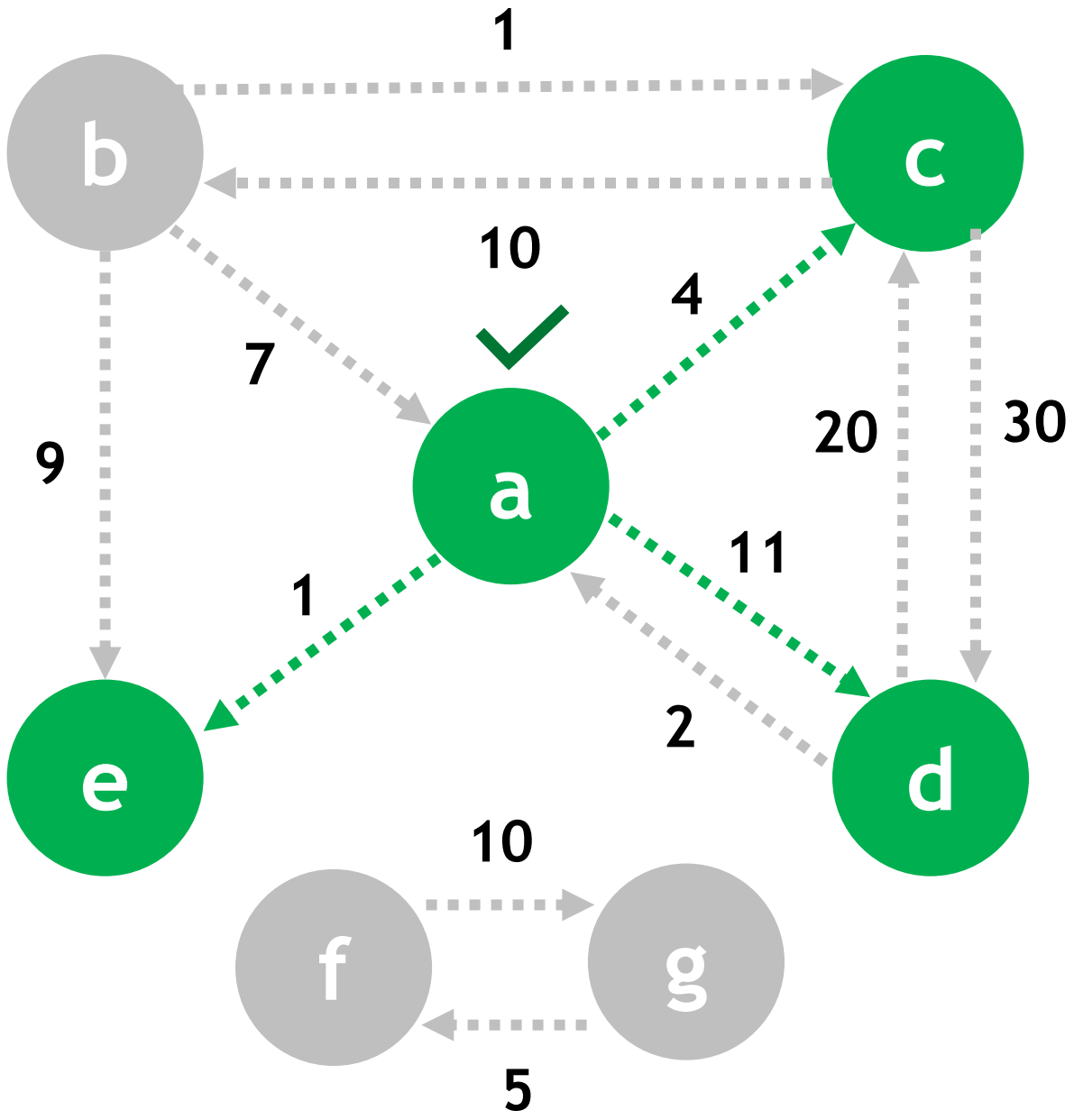
Print: a

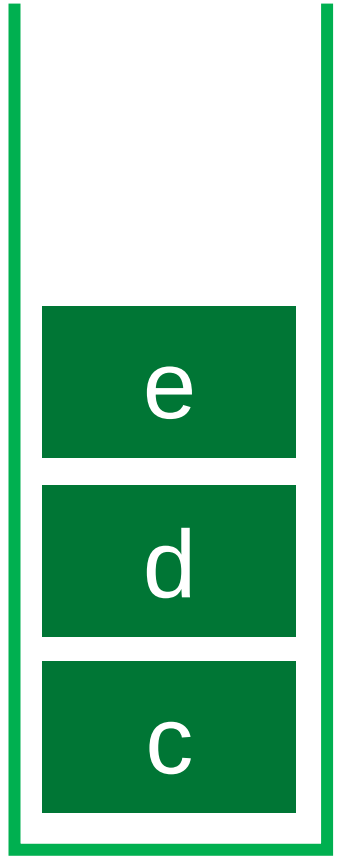




Zásobník

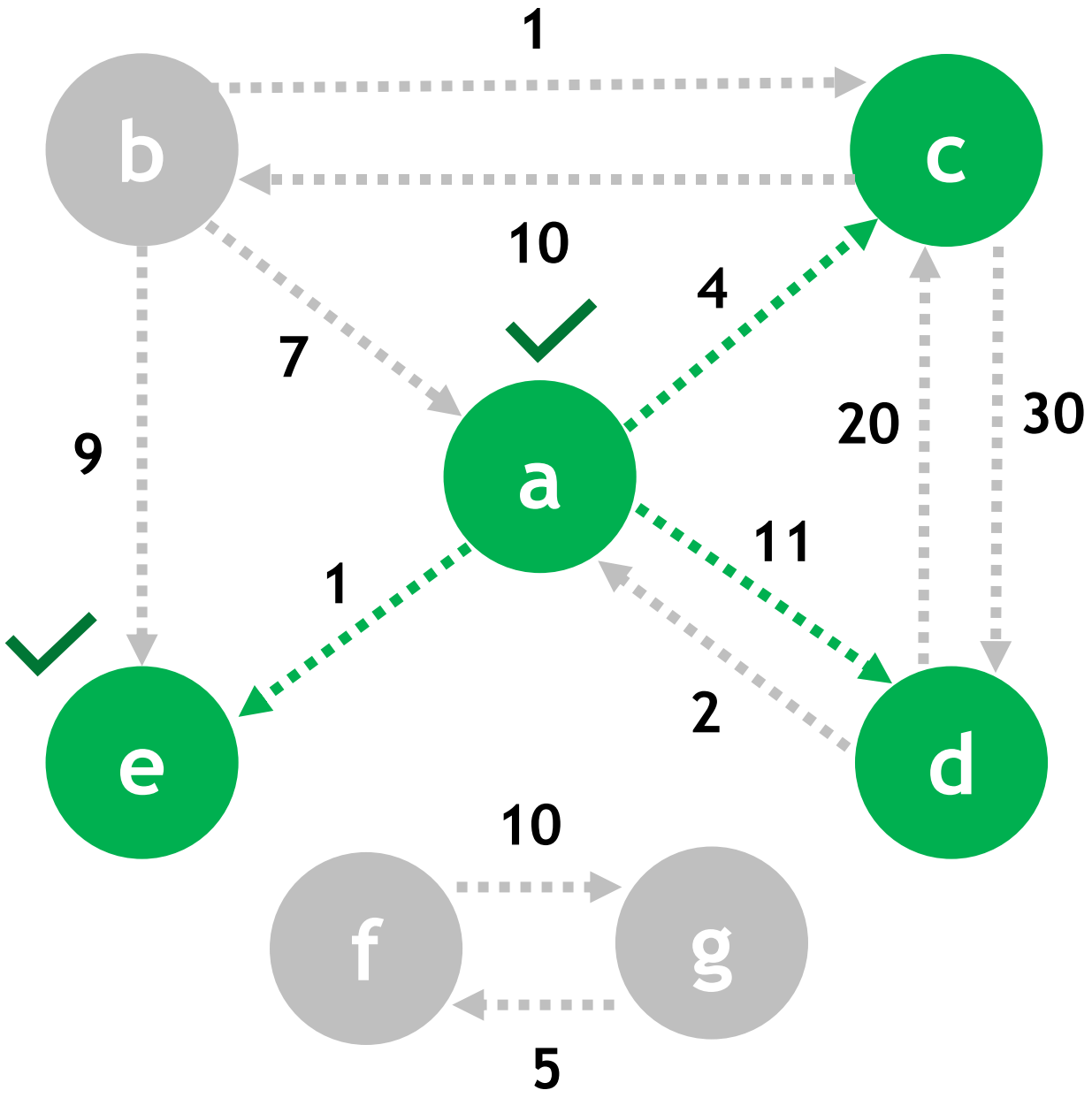
Print: a

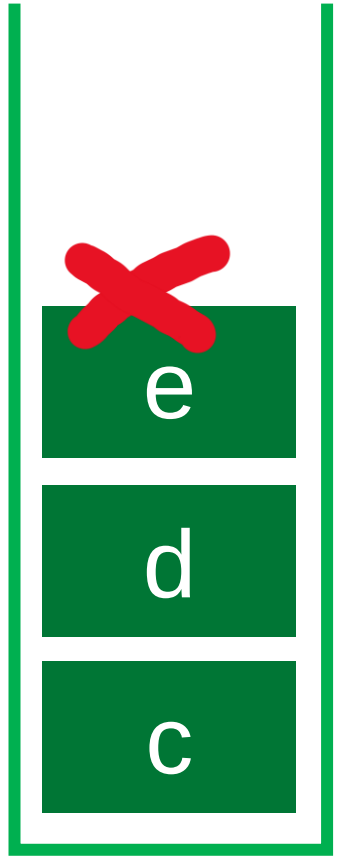




Zásobník

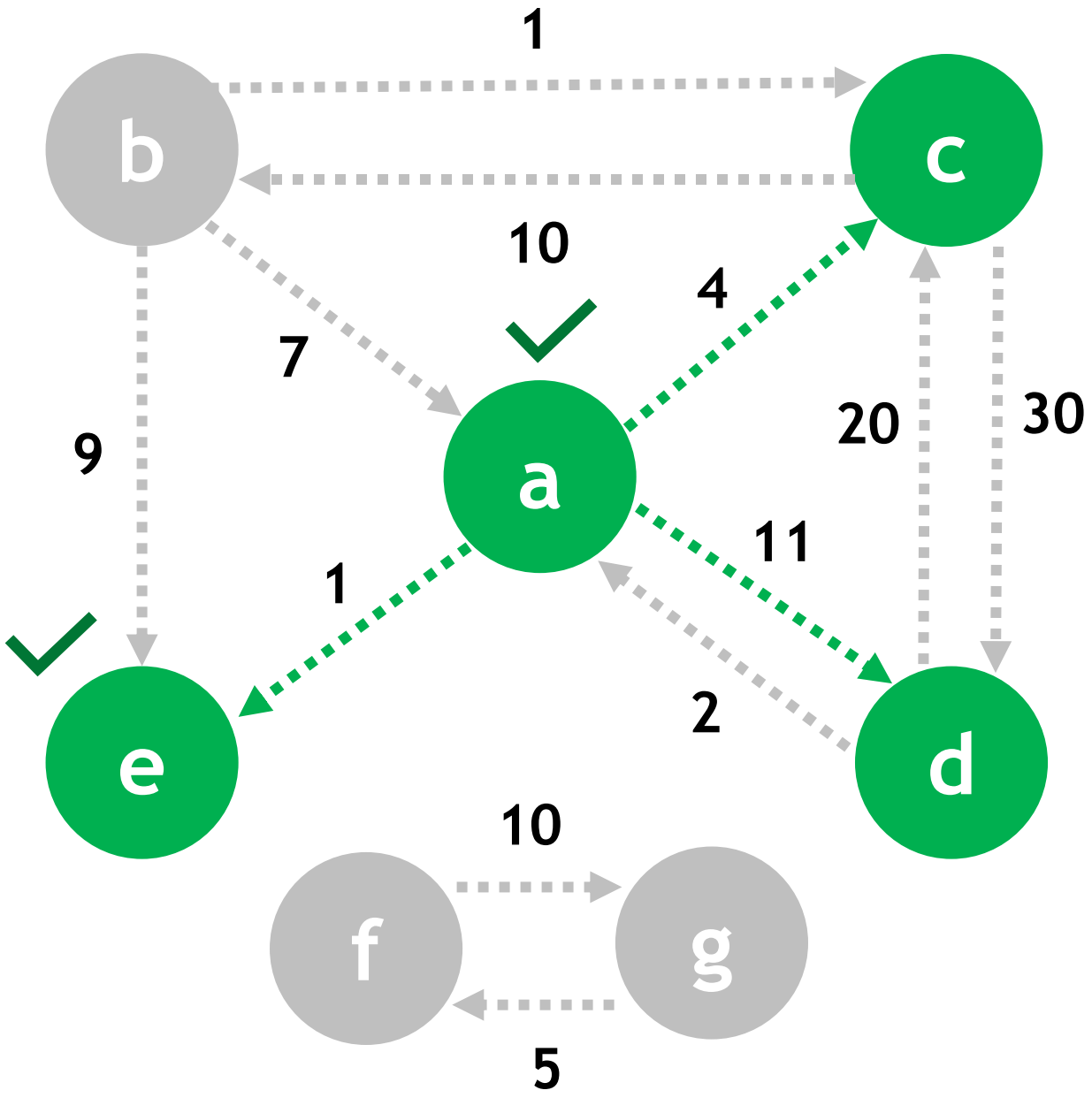
Print: a

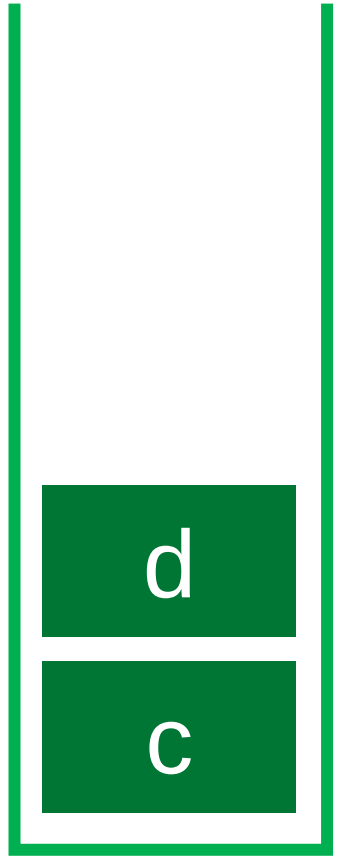




Zásobník

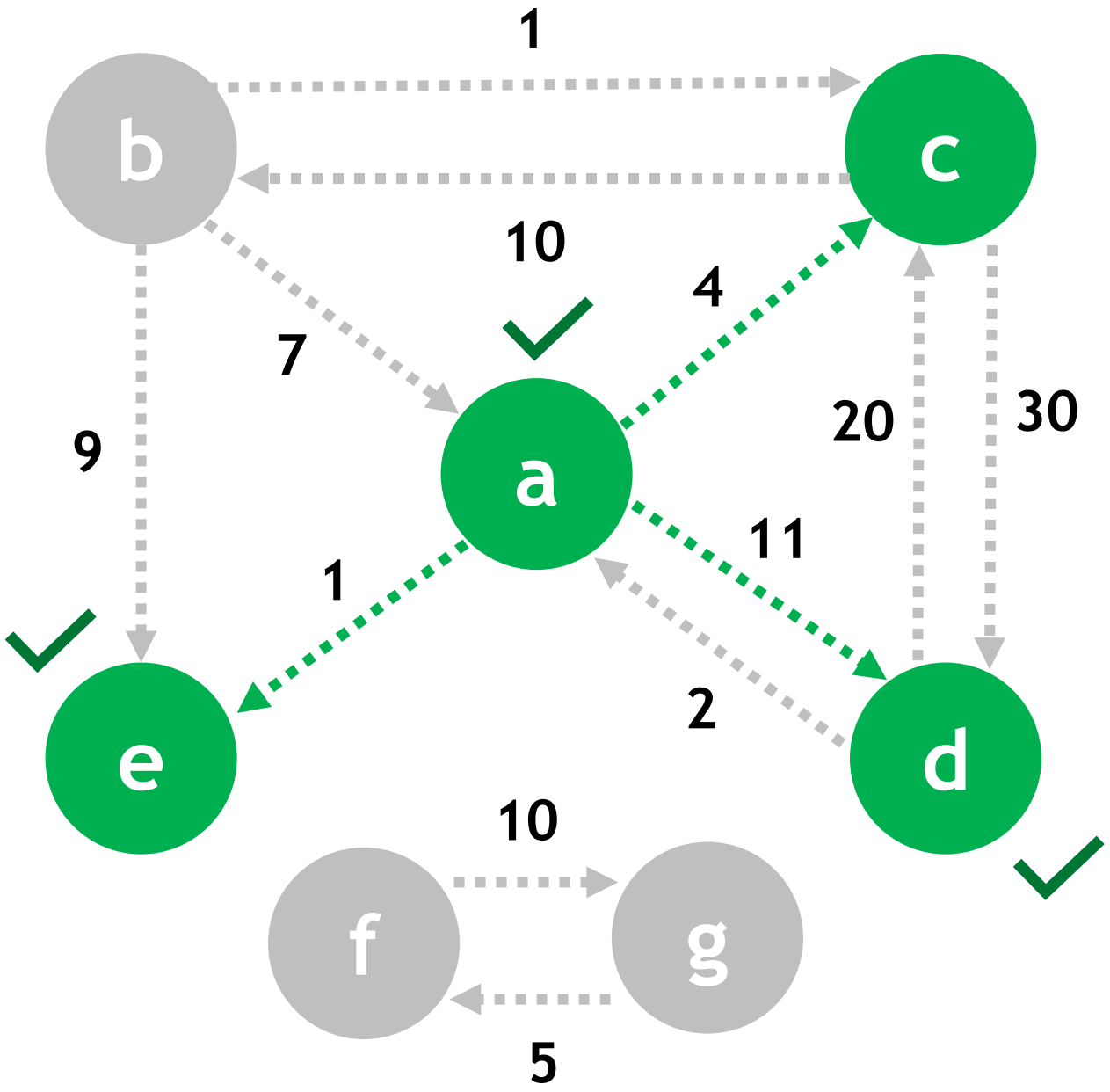
Print: a,e





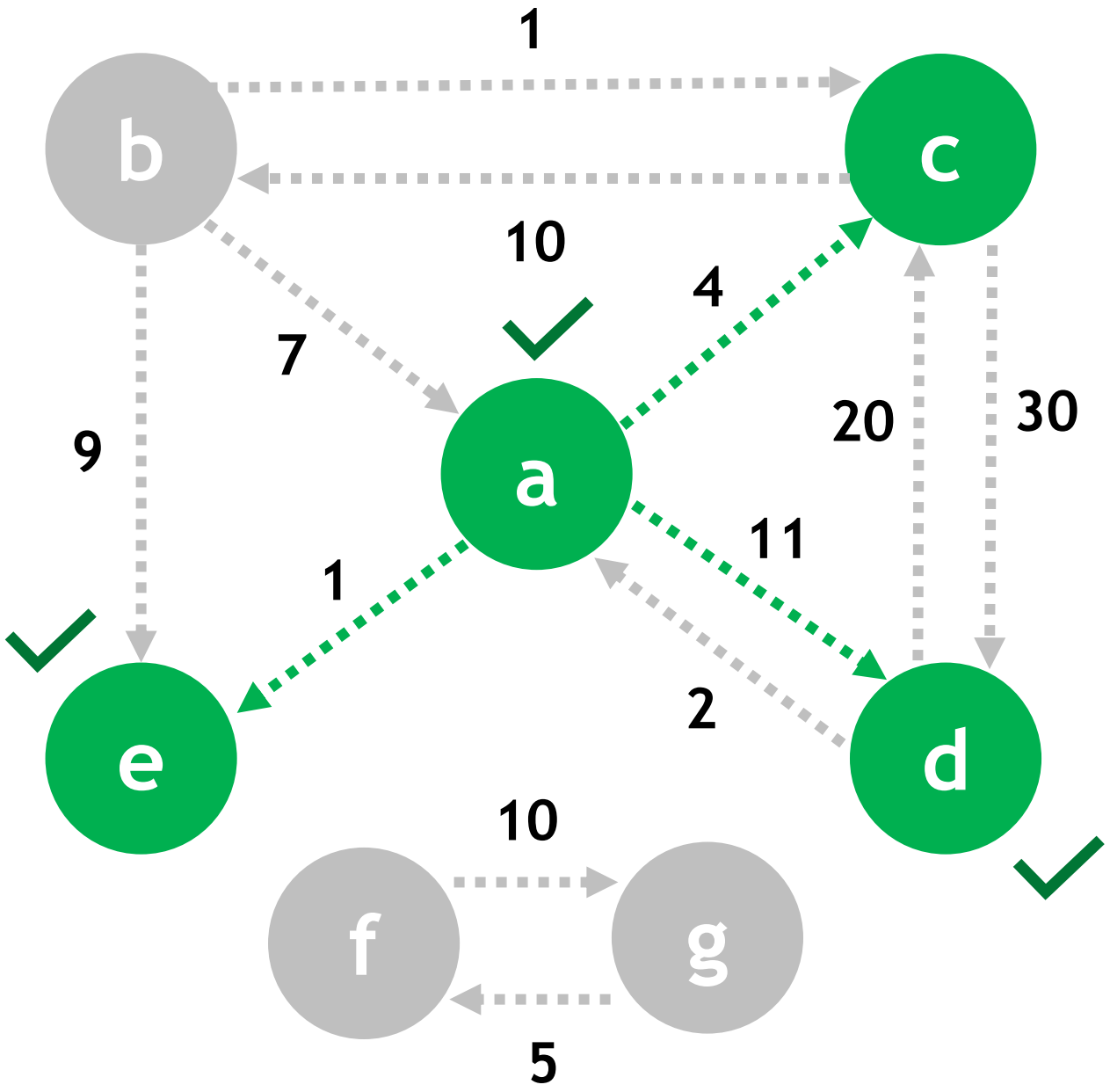
Zásobník

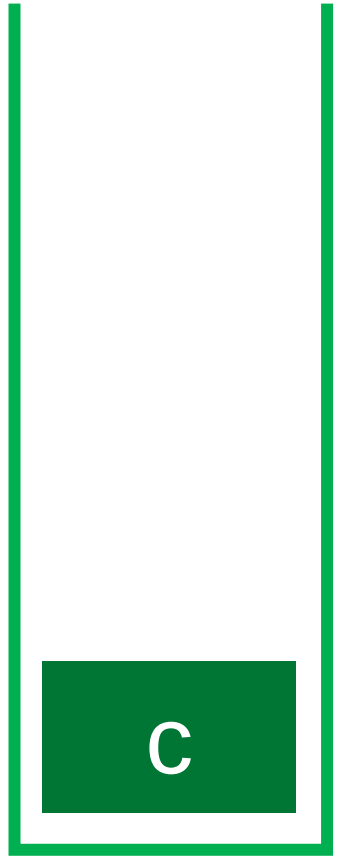
Print: a,e



Zásobník

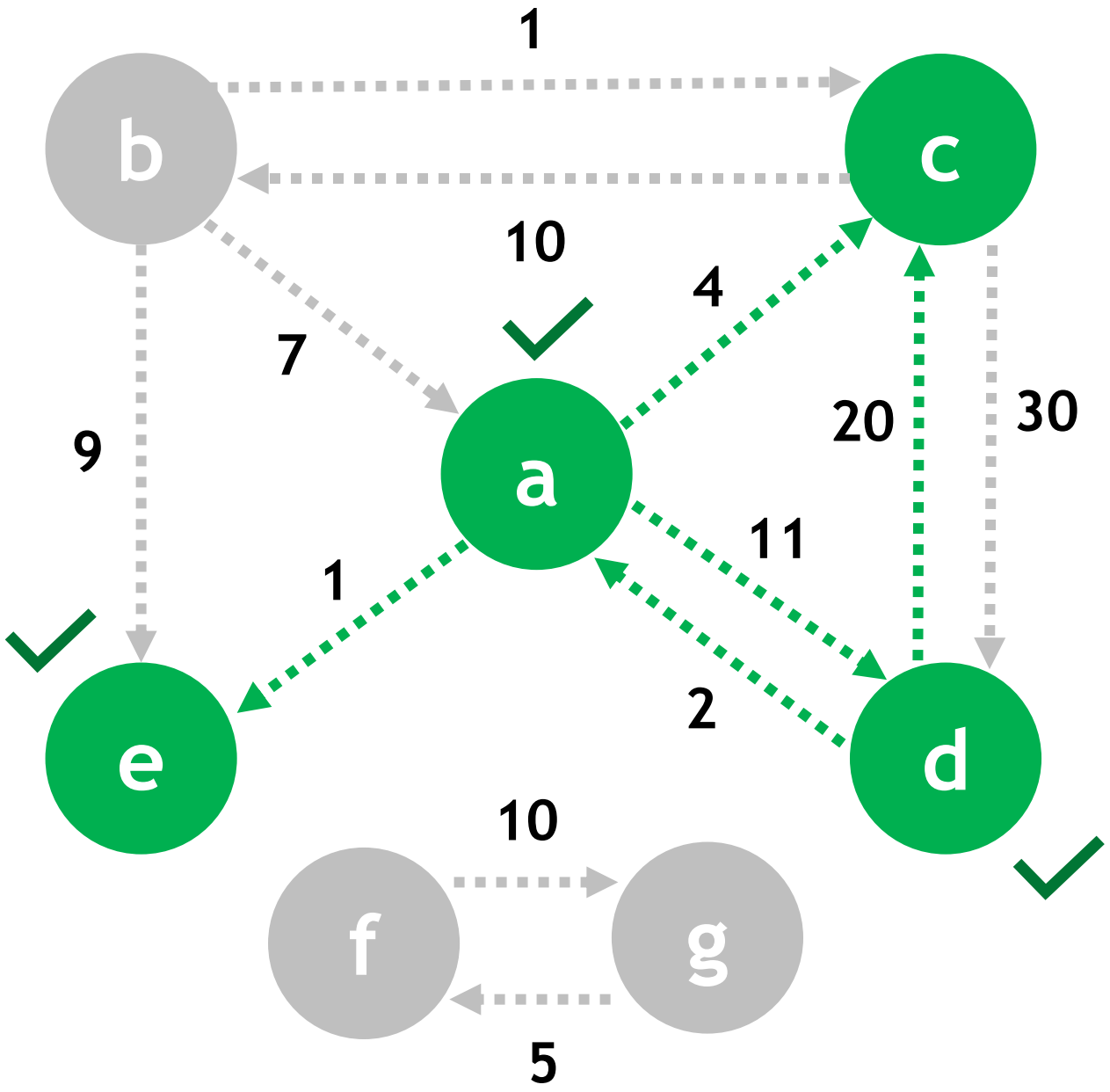
Print: a,e,d

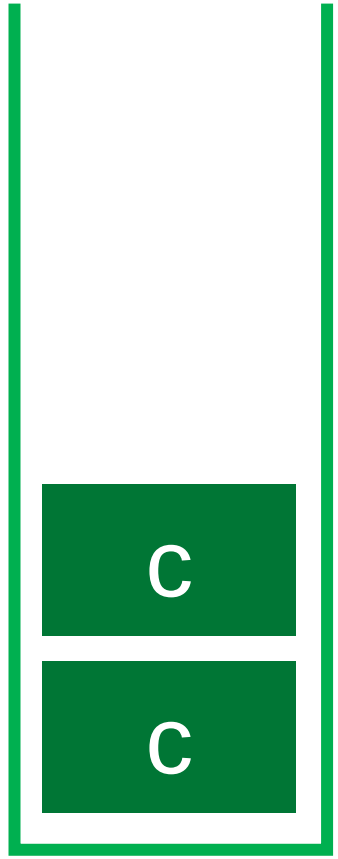




Zásobník

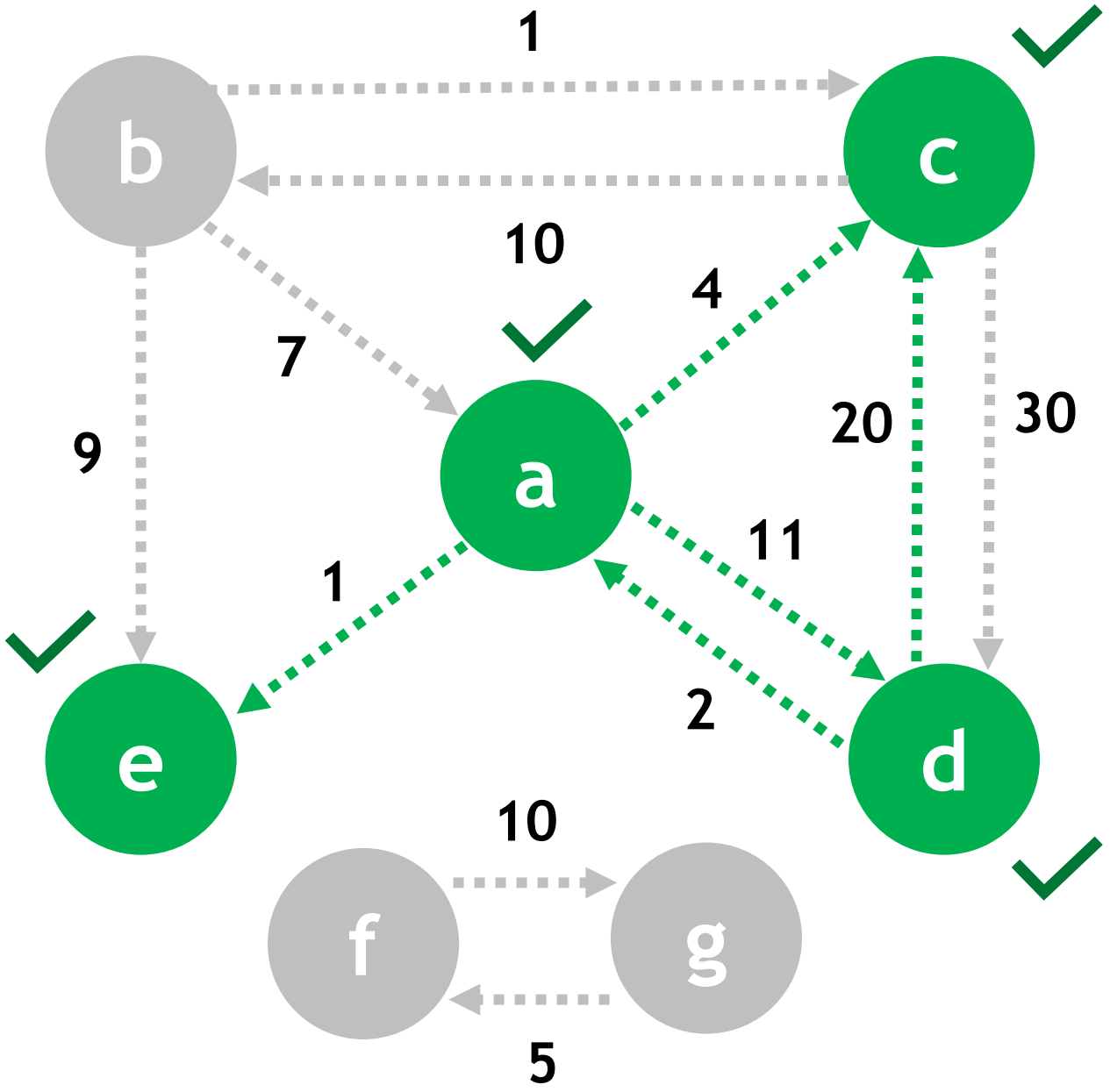
Print: a,e,d





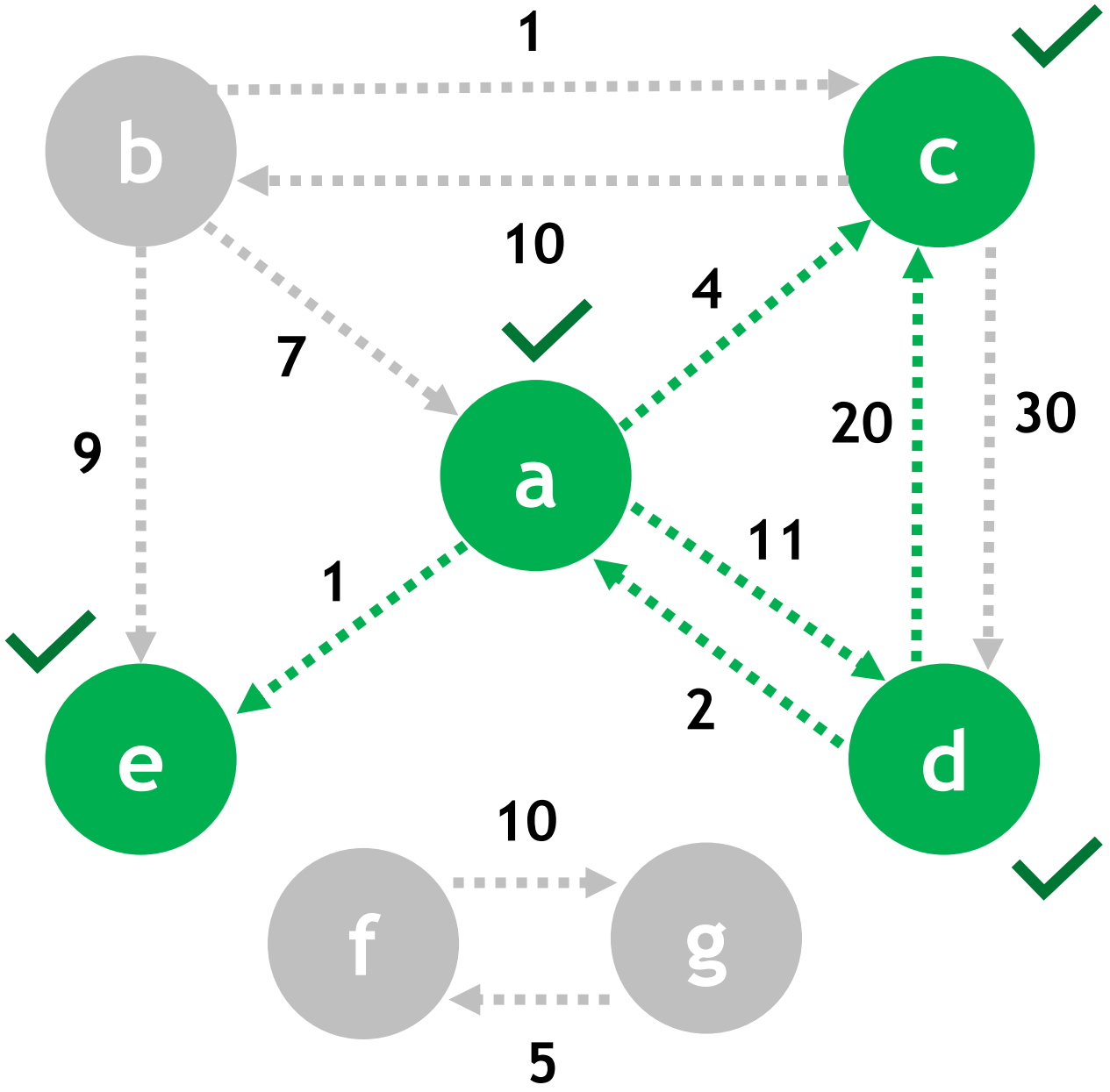
Zásobník

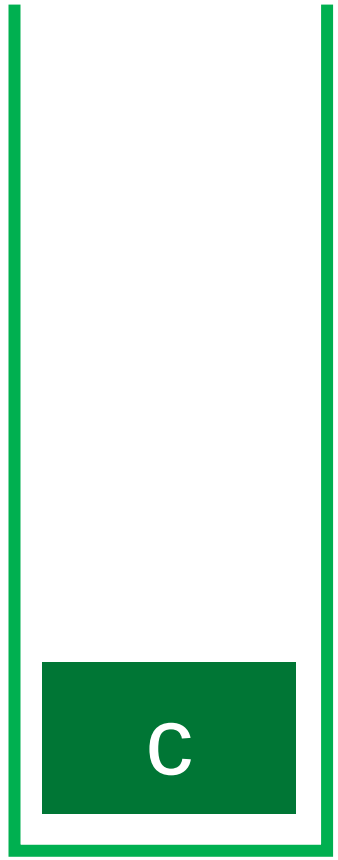
Print: a,e,d



Zásobník

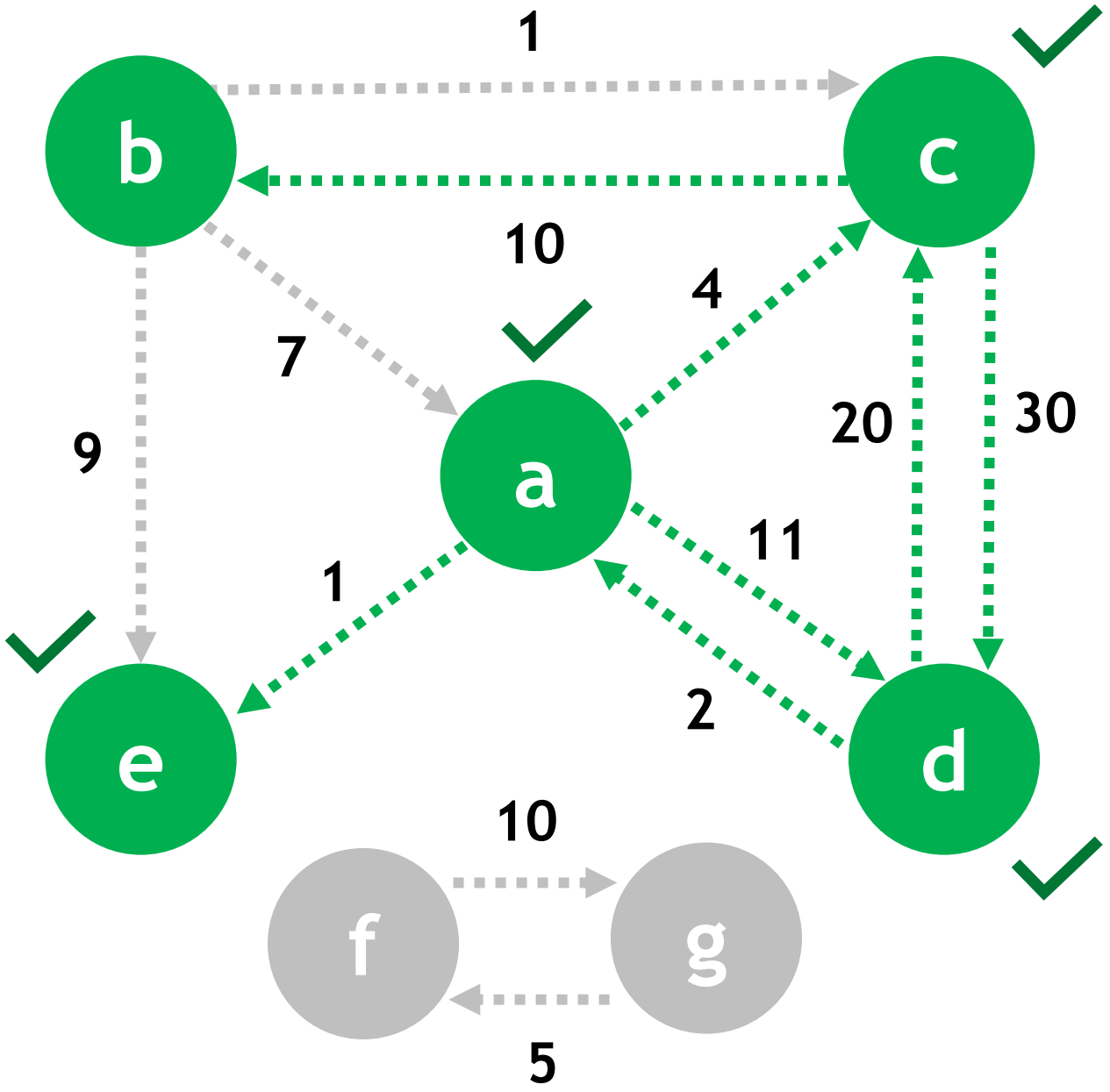
Print: a,e,d,c

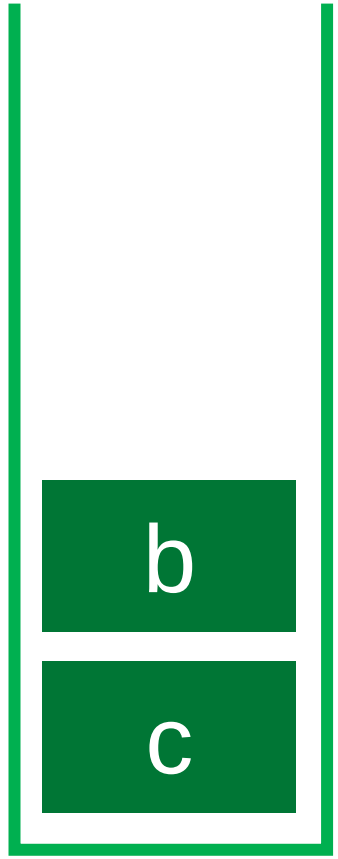




Zásobník

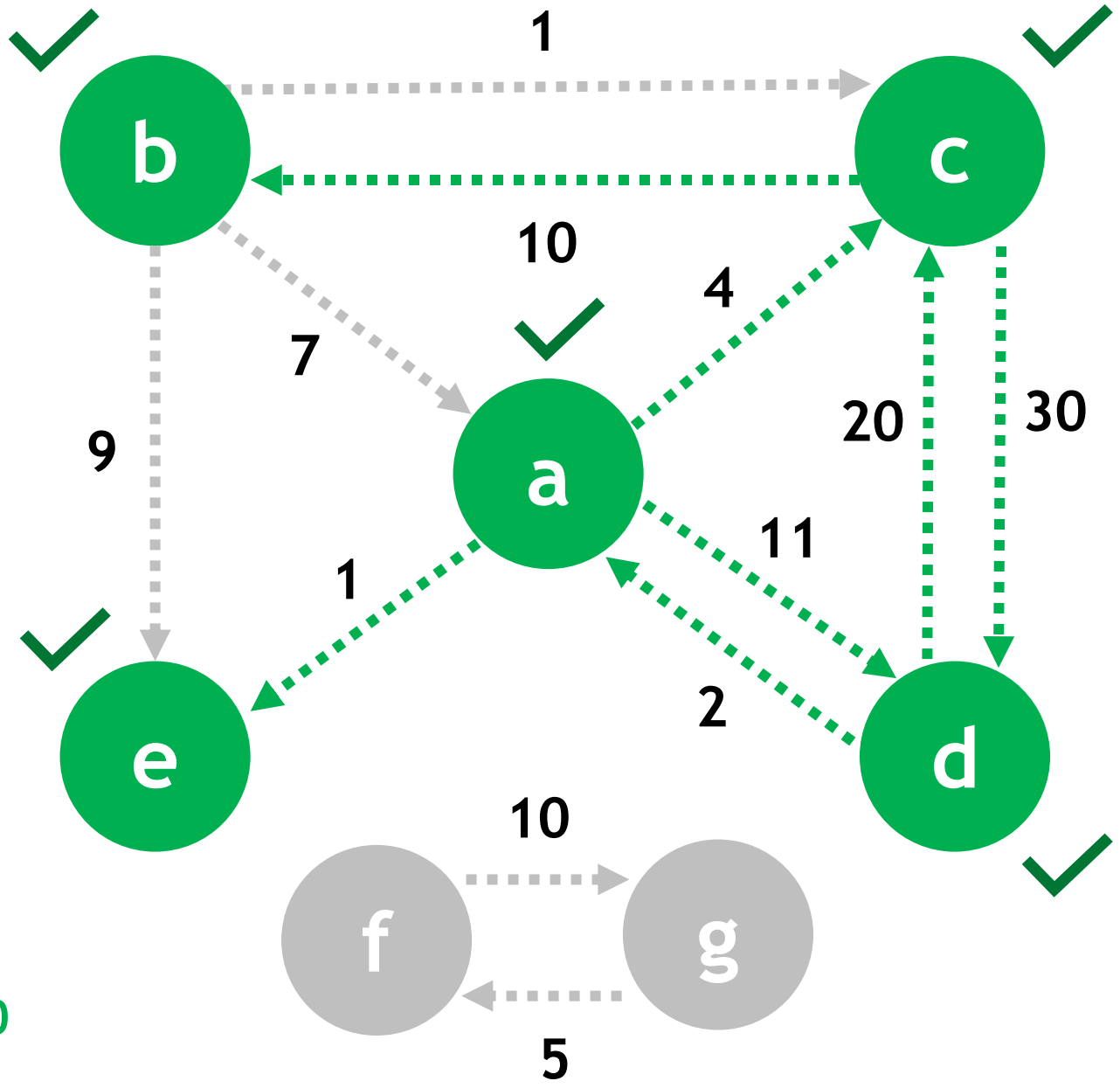
Print: a,e,d,c





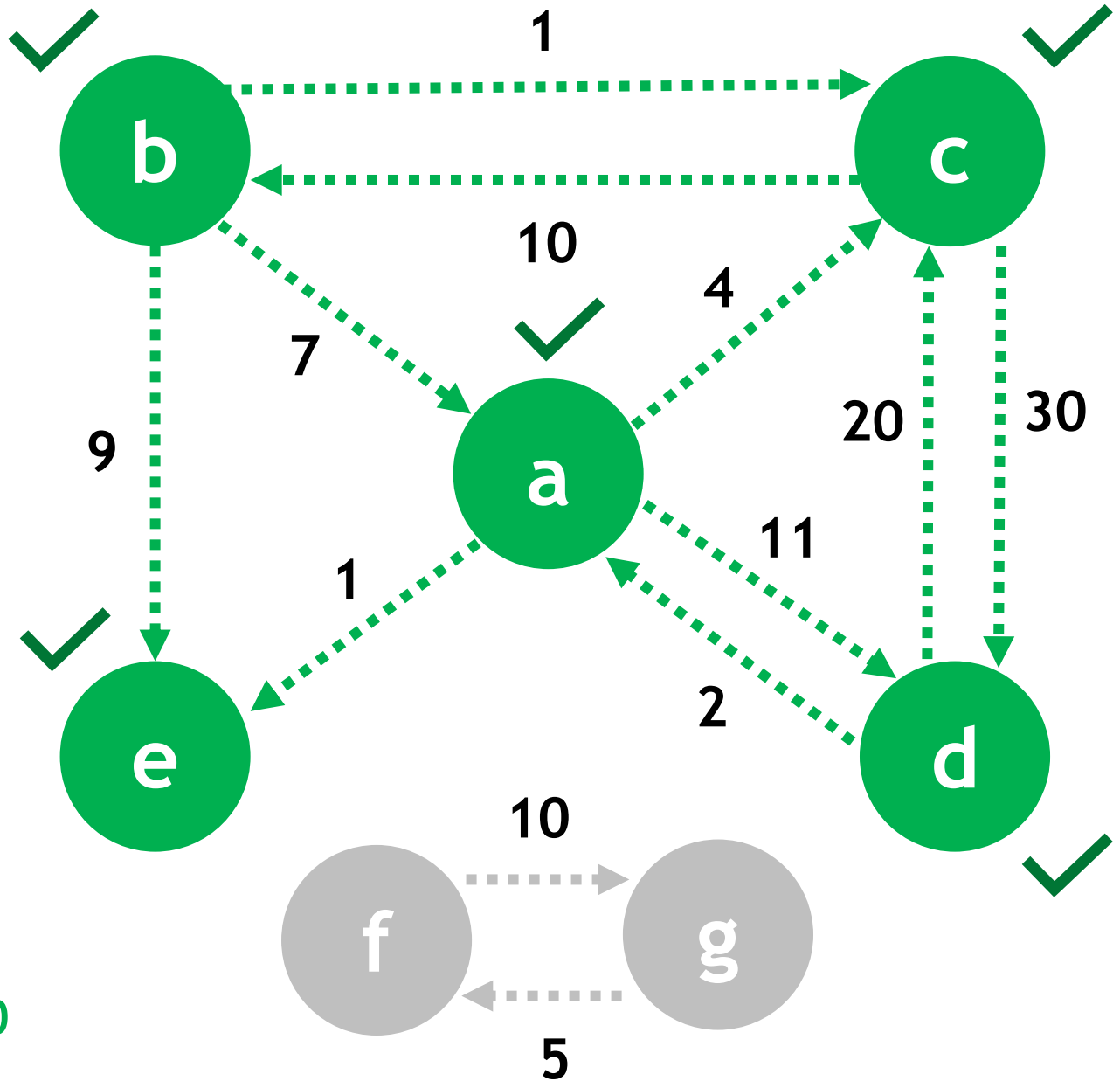
Zásobník

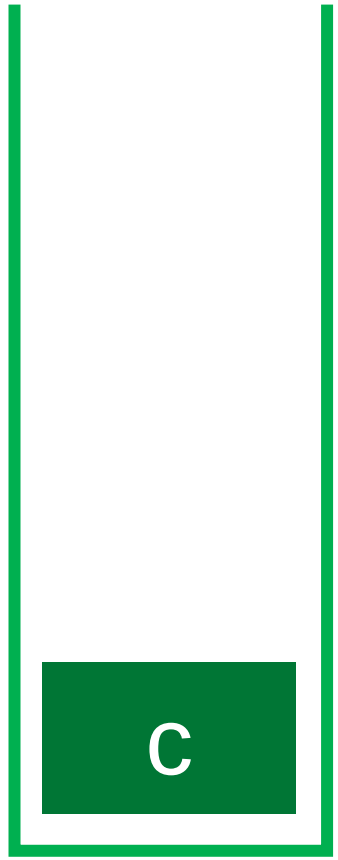
Print: a,e,d,c



Zásobník

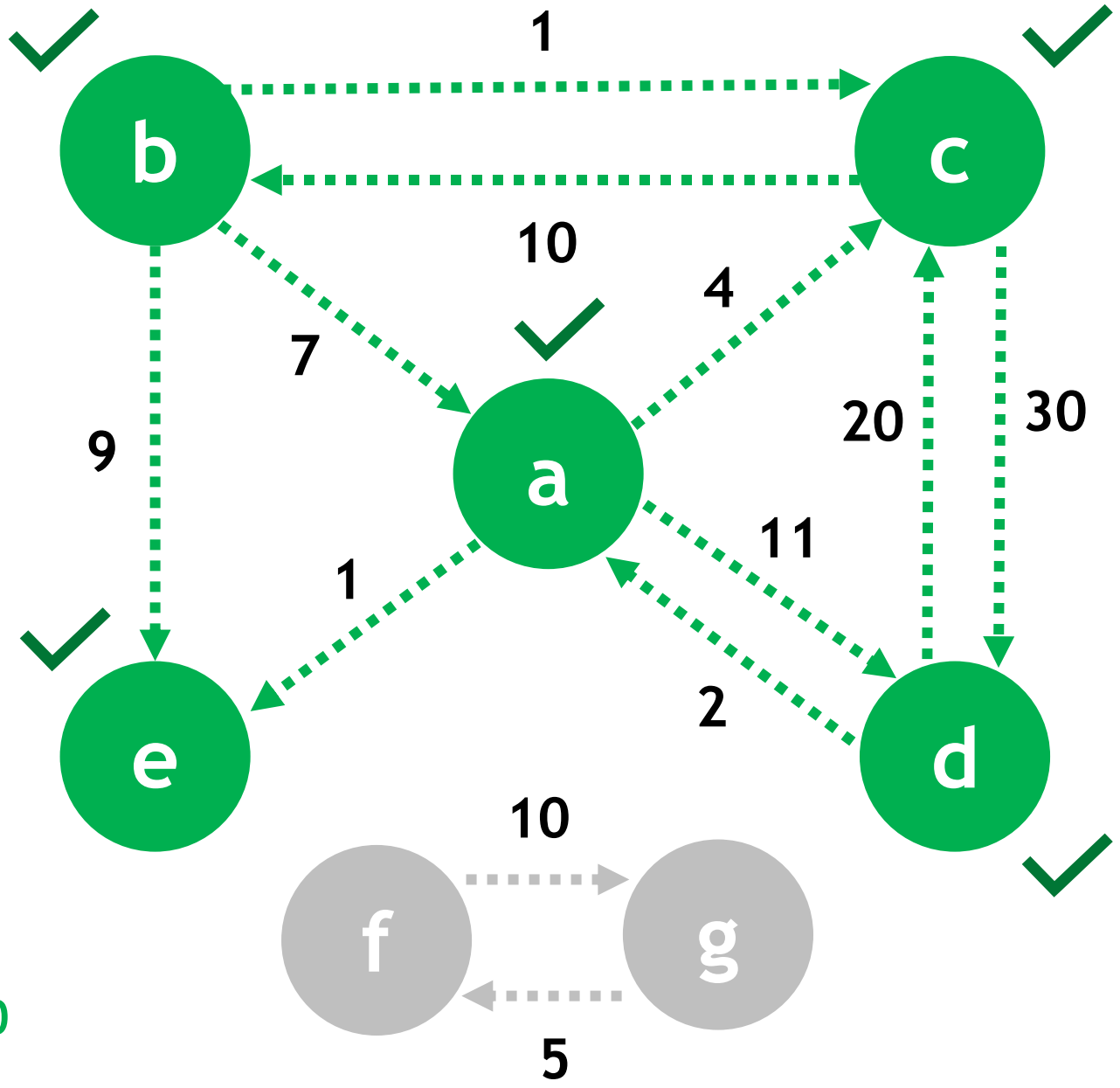
Print: a,e,d,c,b

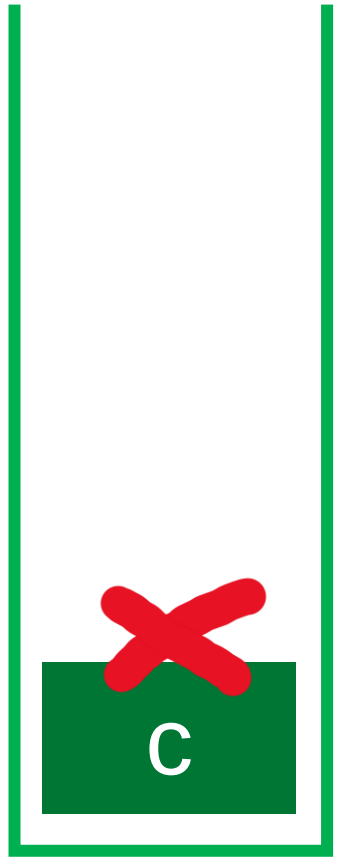




Zásobník

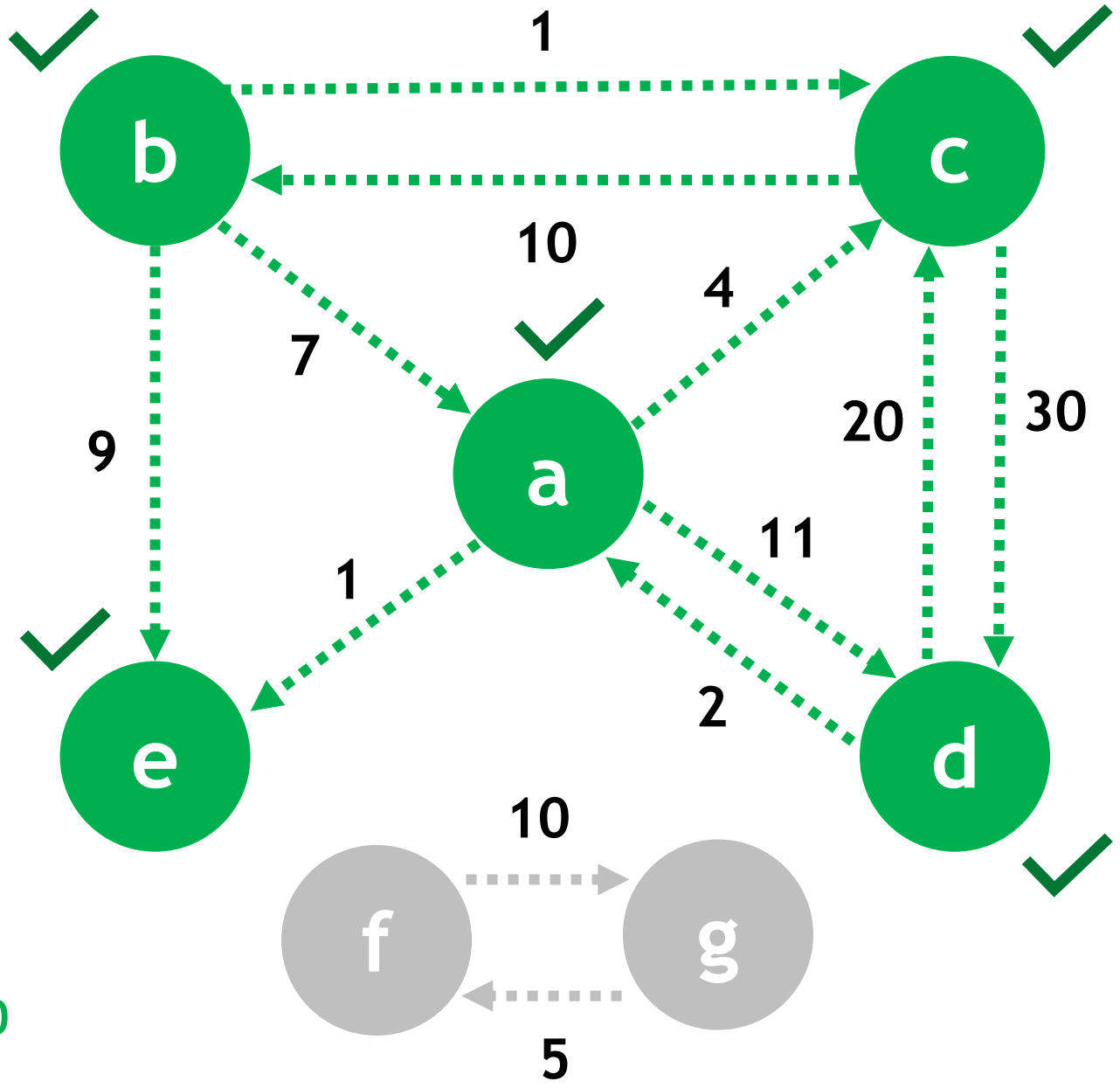
Print: a,e,d,c,b

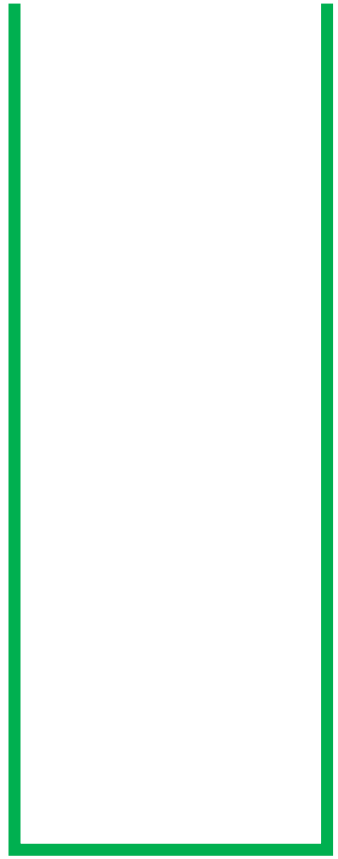




Zásobník

Print: a,e,d,c,b





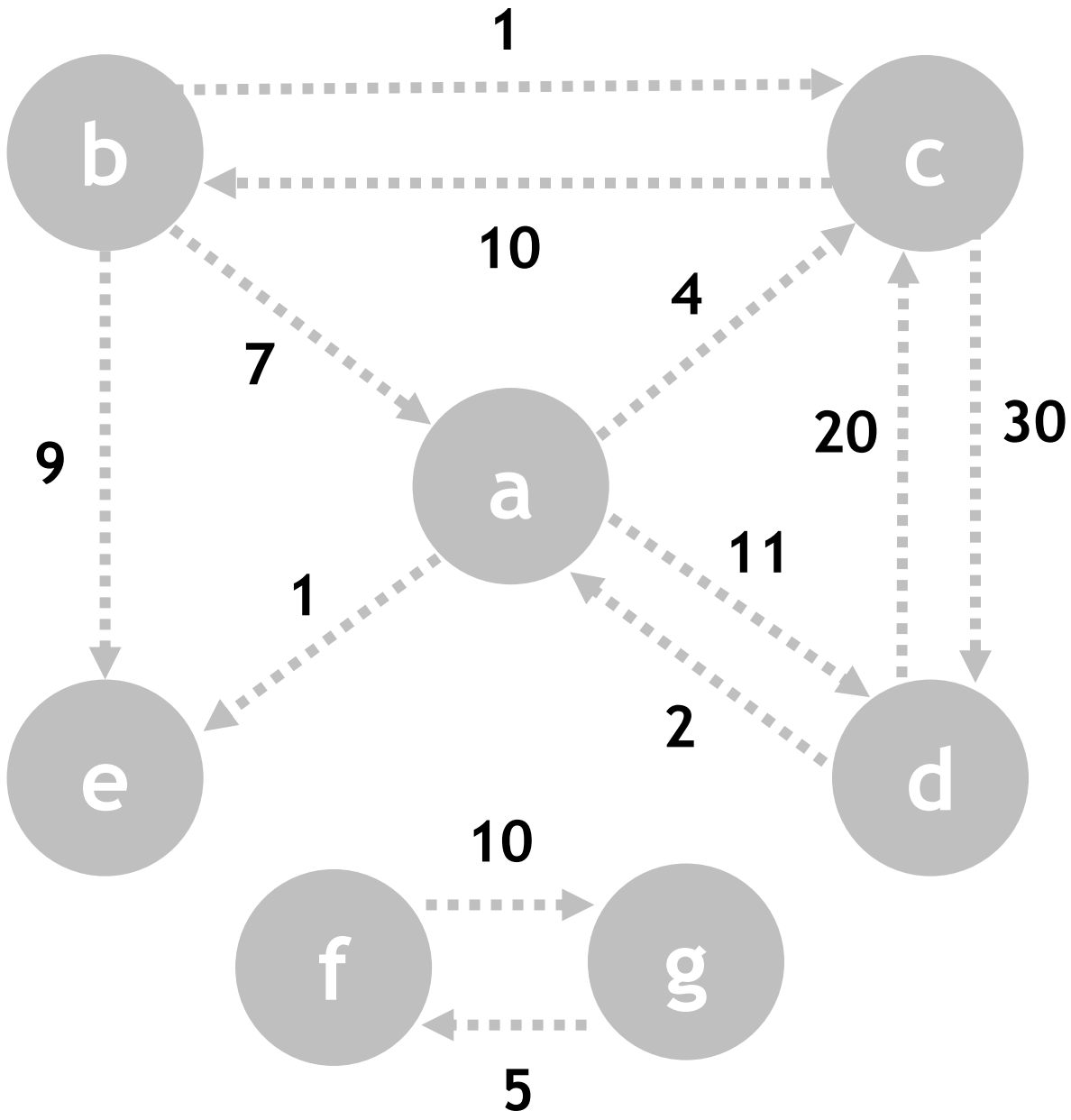
Zásobník

Print: a,e,d,c,b

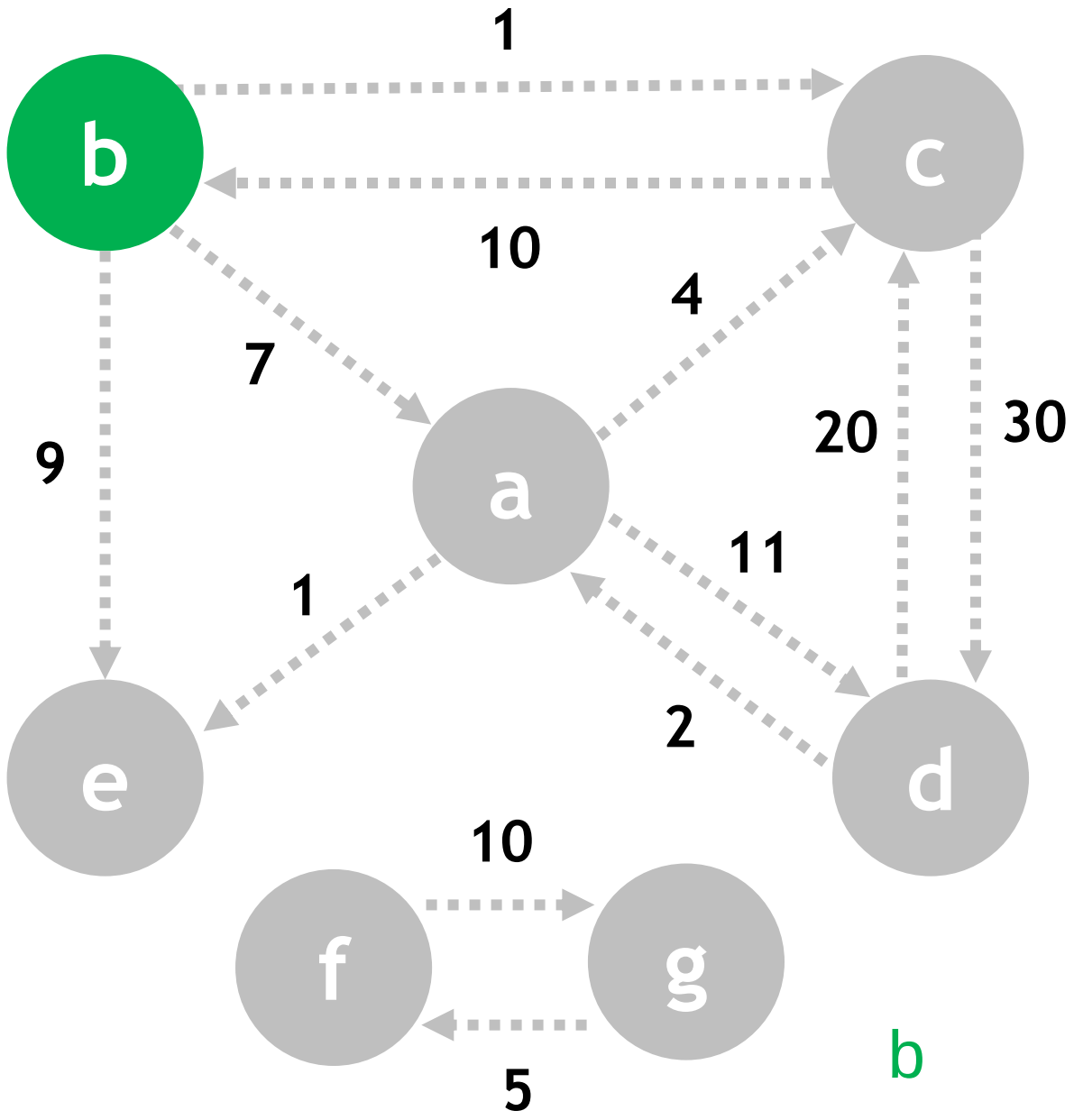
Prehľadávanie do šírky

Prechod grafom do šírky (BFS)

```
// G je graf
// sv je štartovací vrchol

1  BFS(G,sv):
2      nech Q je prázdna fronta
3      označ sv ako navštívený
4      Q.push(sv)
5      while Q je neprázdna
6          v = Q.pop()
9          for all hrany vychádzajúce z v do w do
10             if w je nenavštívený:
11                 označ w ako navštívený
12                 Q.push(w)
```


1
b
c
10
4
7
9
a
20
30
11
1
2
e
d
10
f
g
5

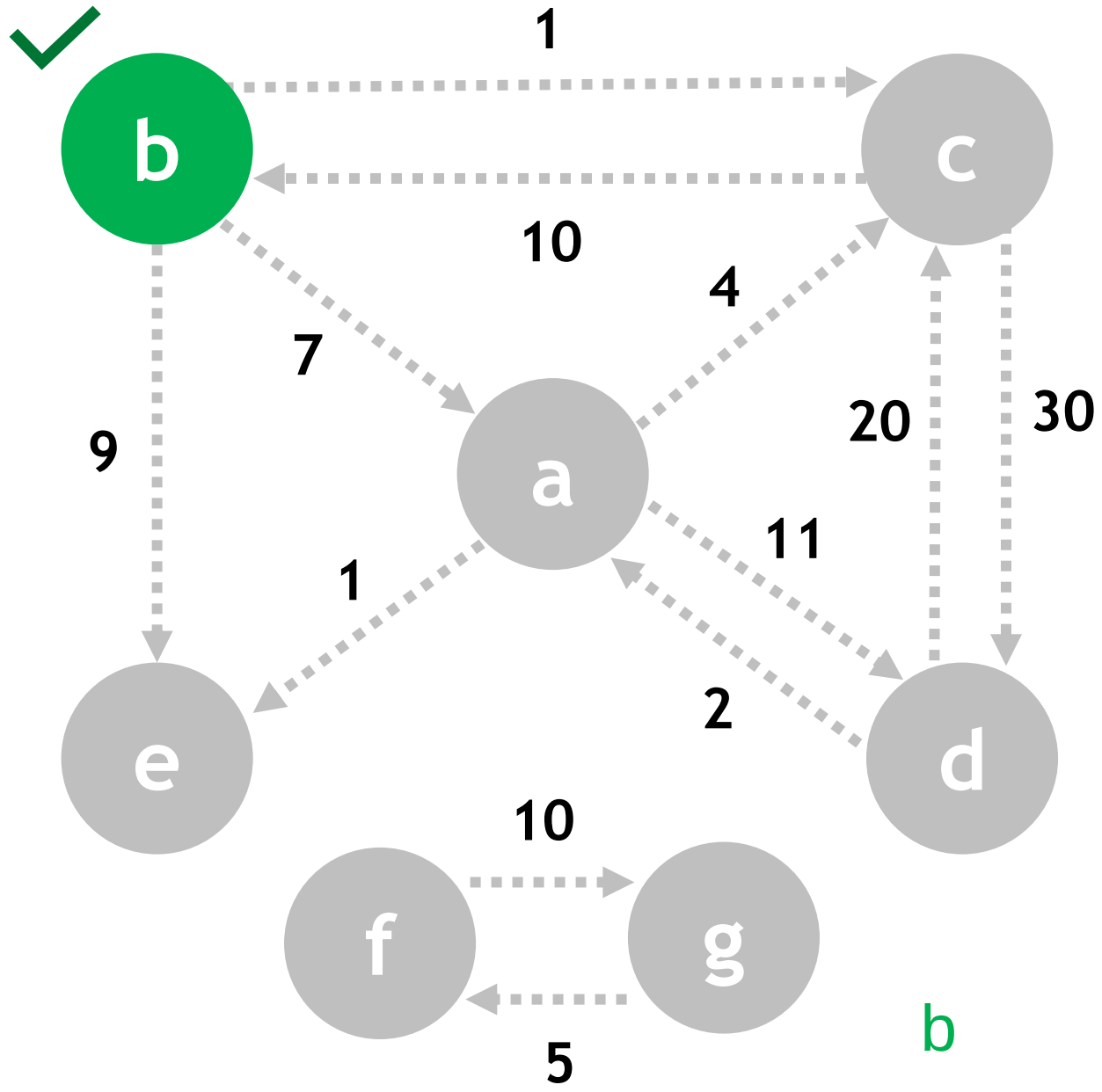
Začiatok prehľadavánia z vrcholu b



b

Fronta

Začiatok prehľadavánia z vrcholu b



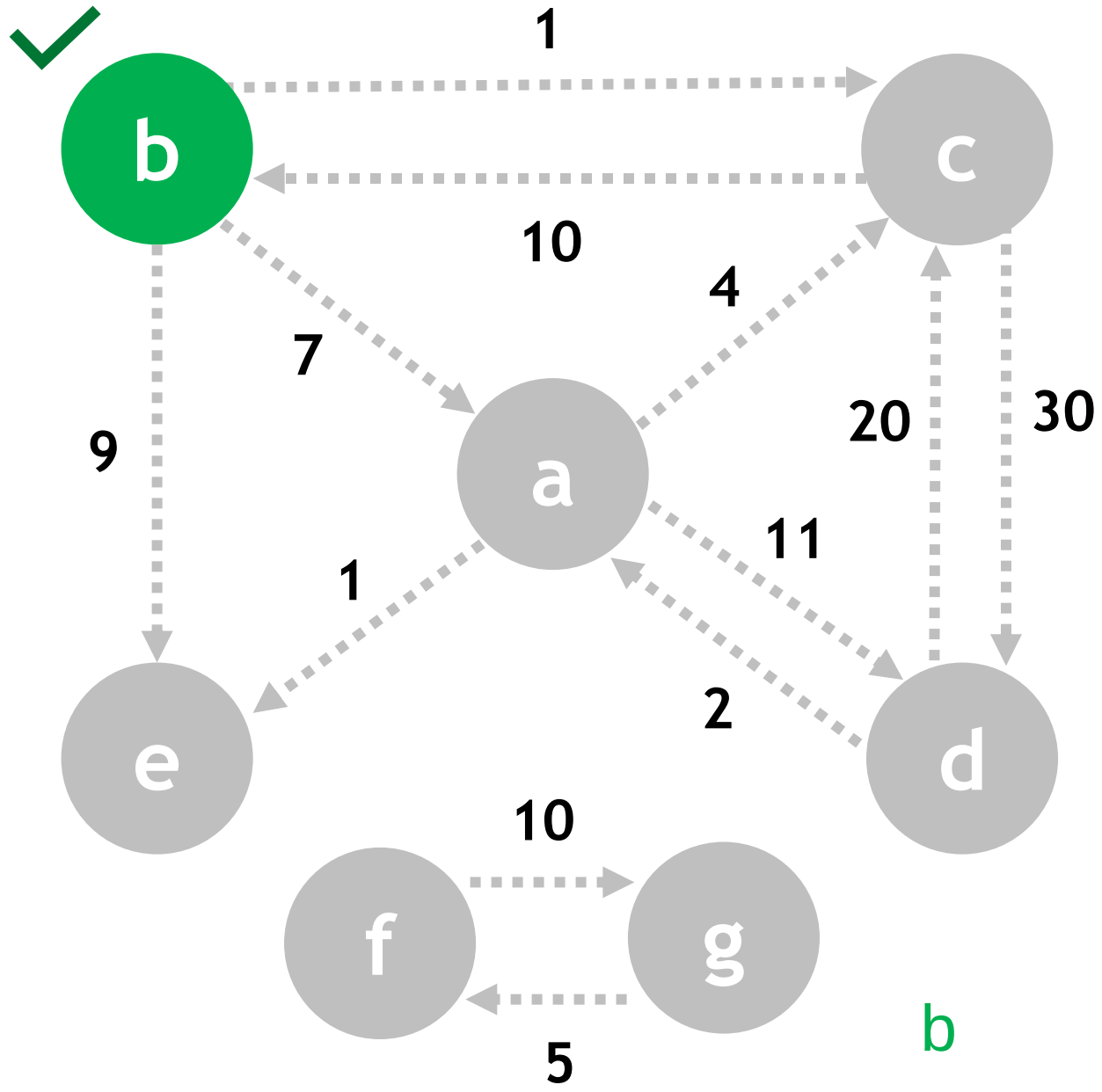
Print: b

b

Fronta

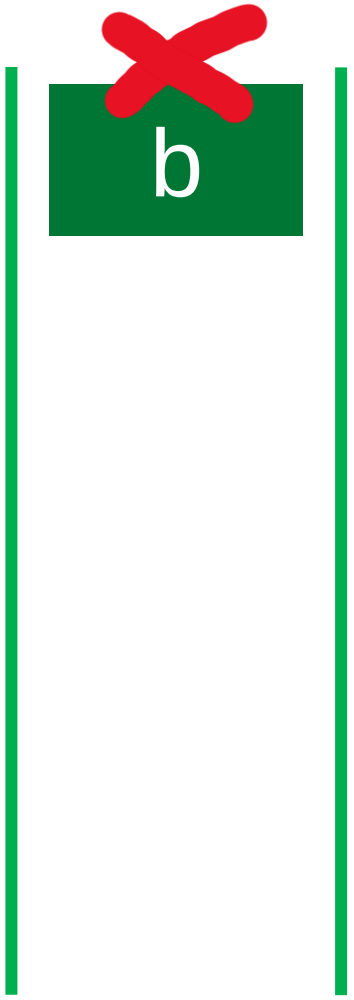
b

Začiatok prehľadavánia z vrcholu b



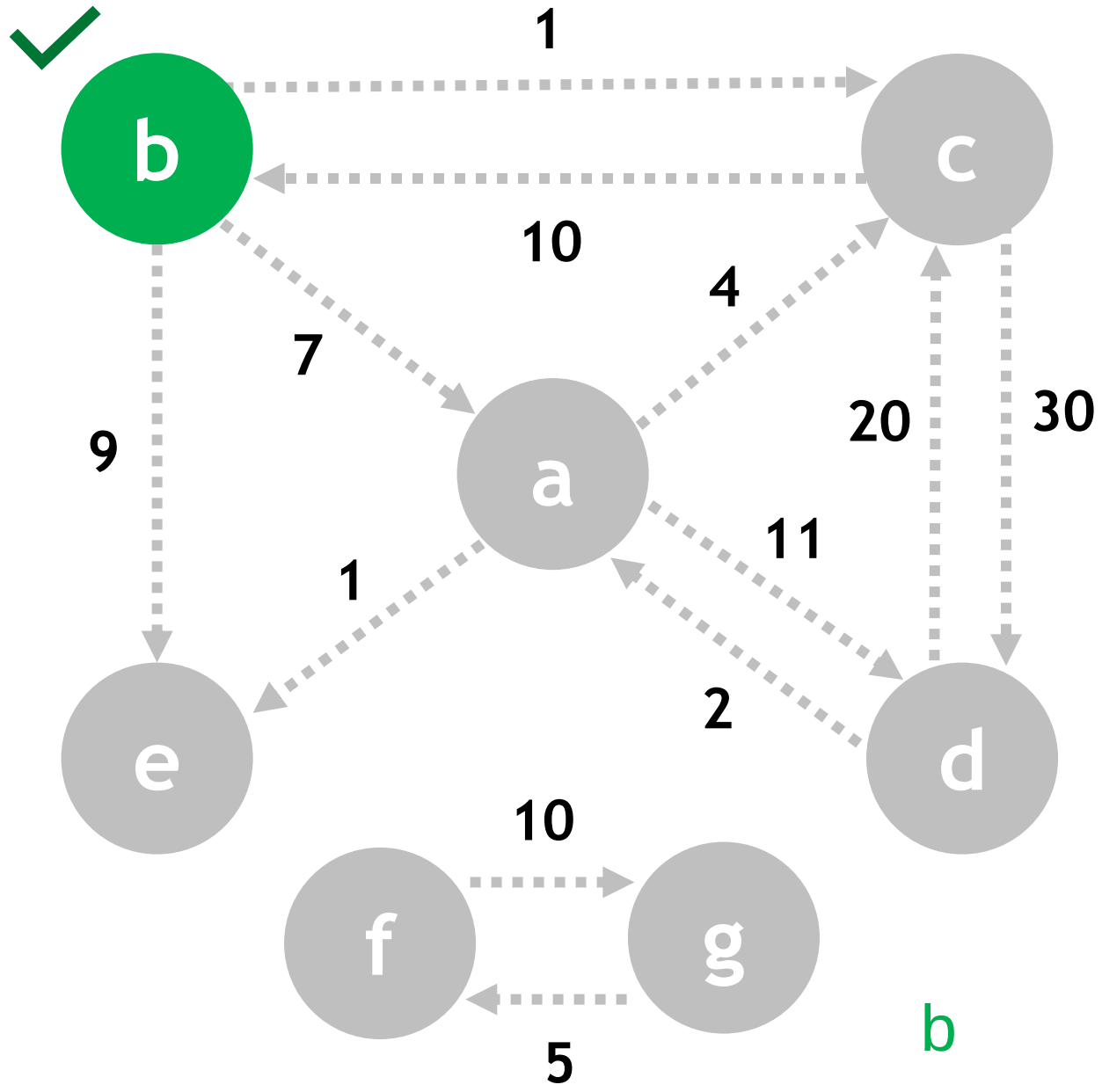
Print: b

b



Fronta

Začiatok prehľadavánia z vrcholu b

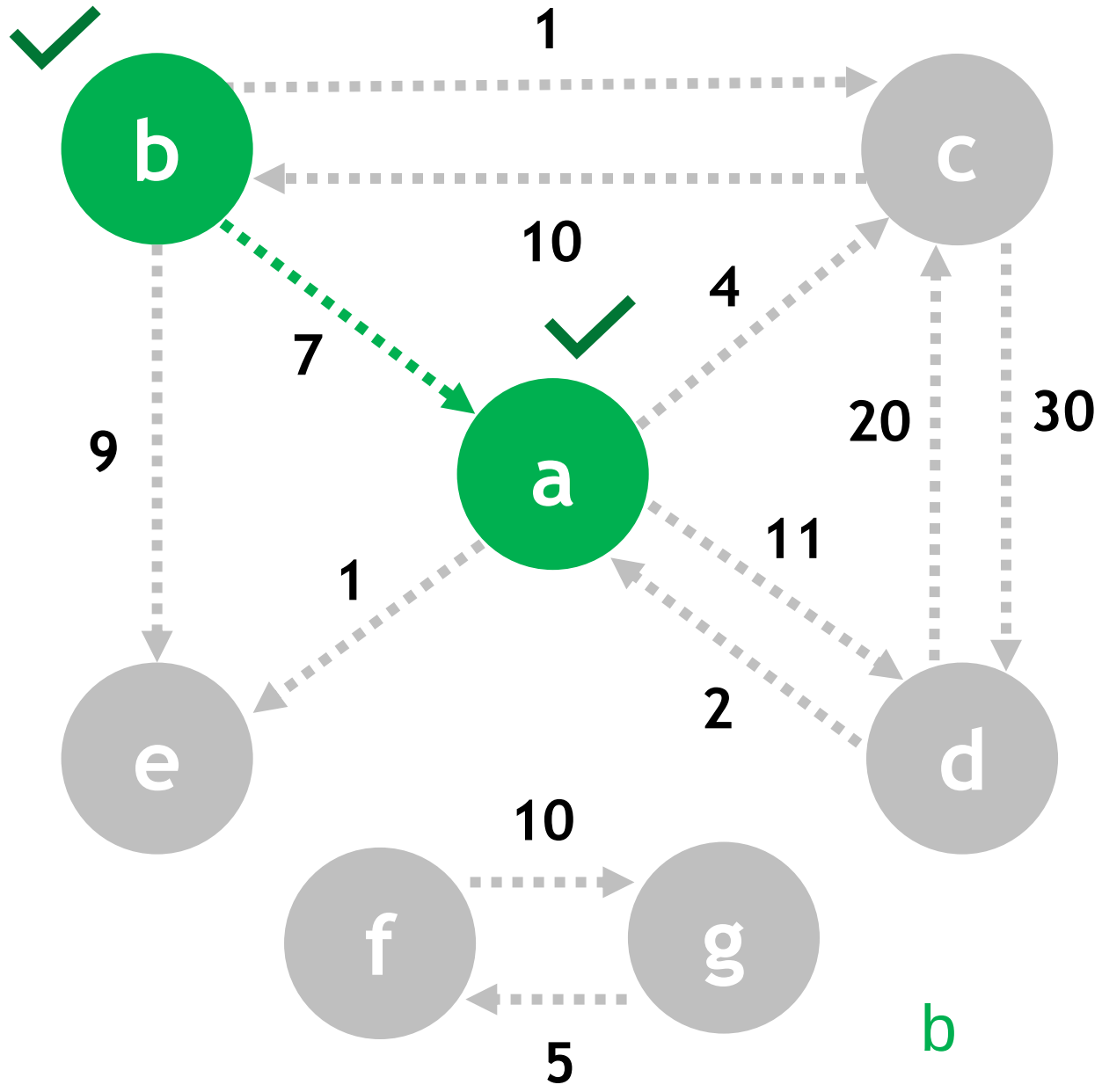


Print: b

b

Fronta

Začiatok prehľadavánia z vrcholu b



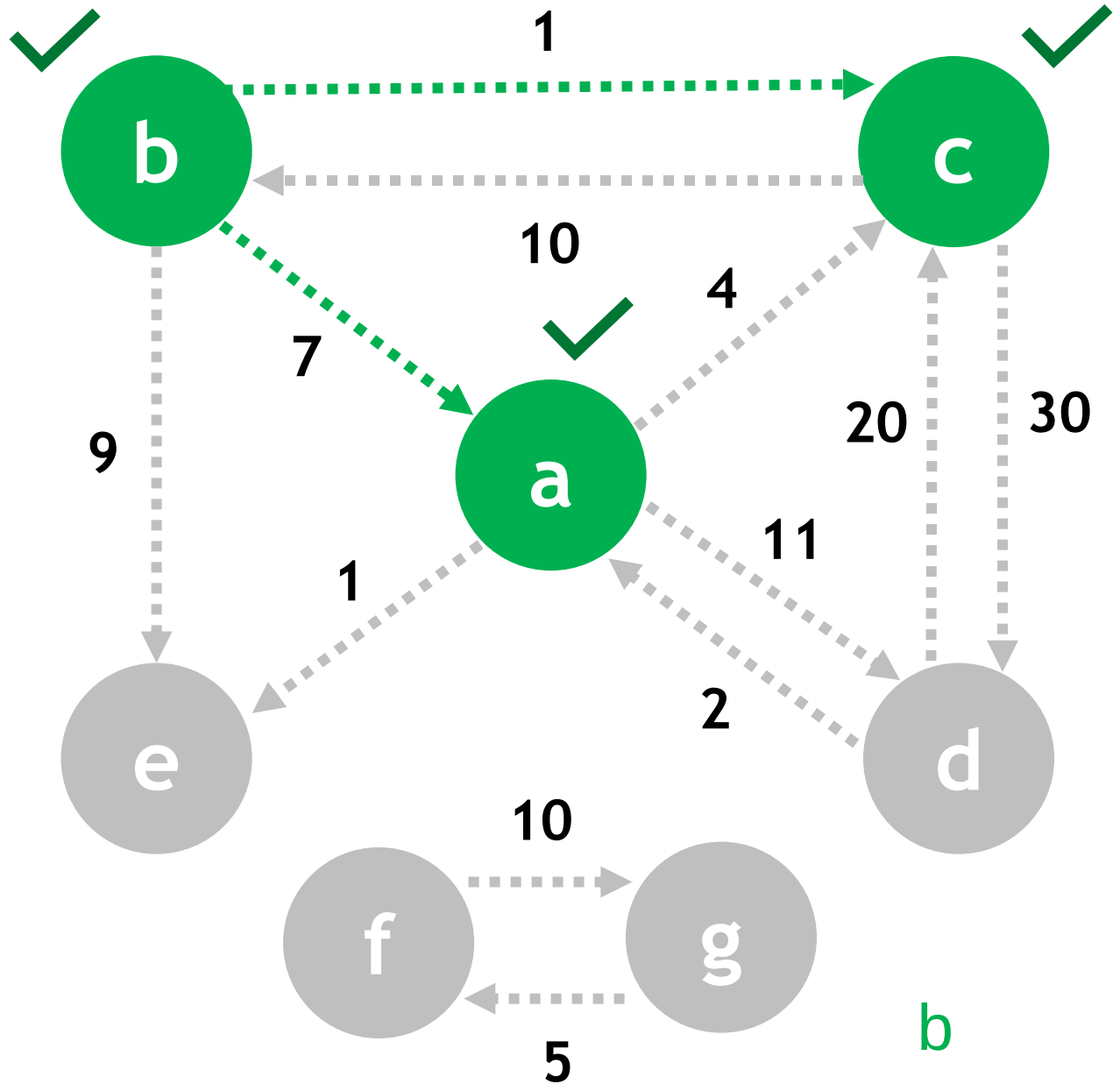
Print: b,a

b

a

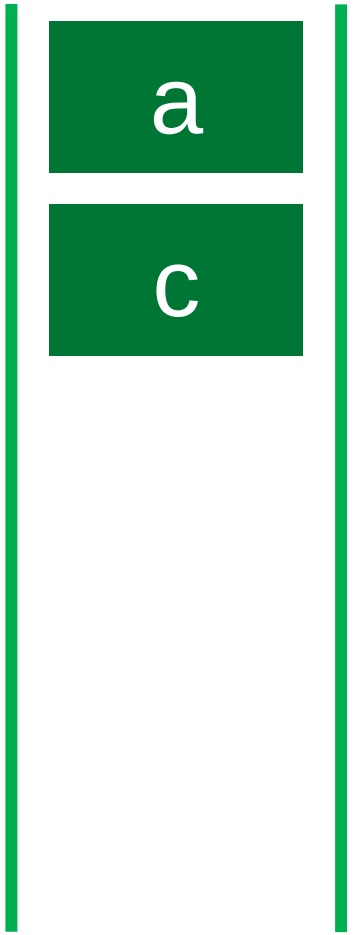
Fronta

Začiatok prehľadavánia z vrcholu b



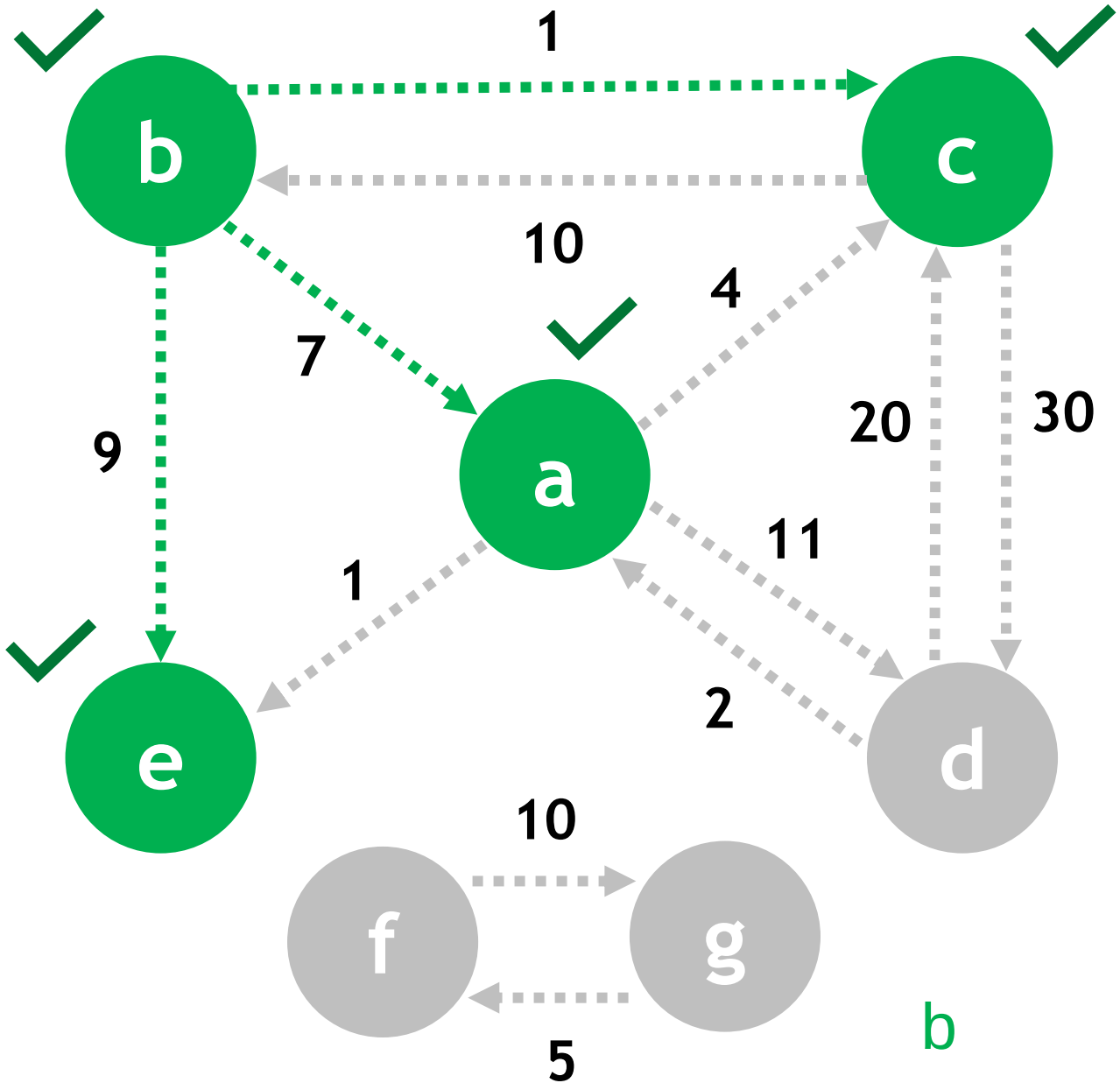
Print: b,a,c

b



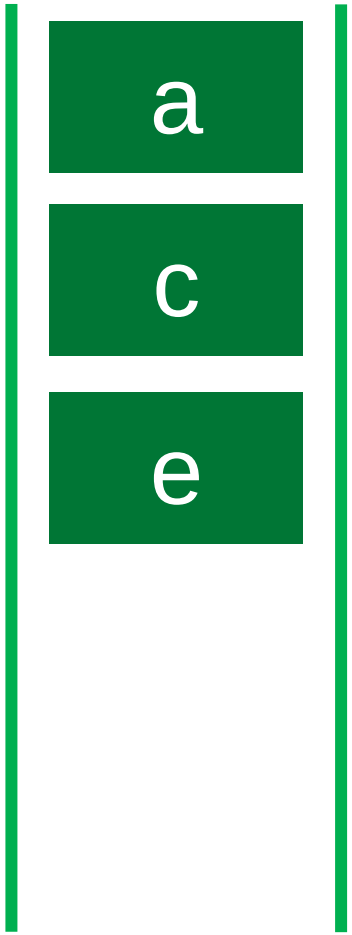
Fronta

Začiatok prehľadavánia z vrcholu b



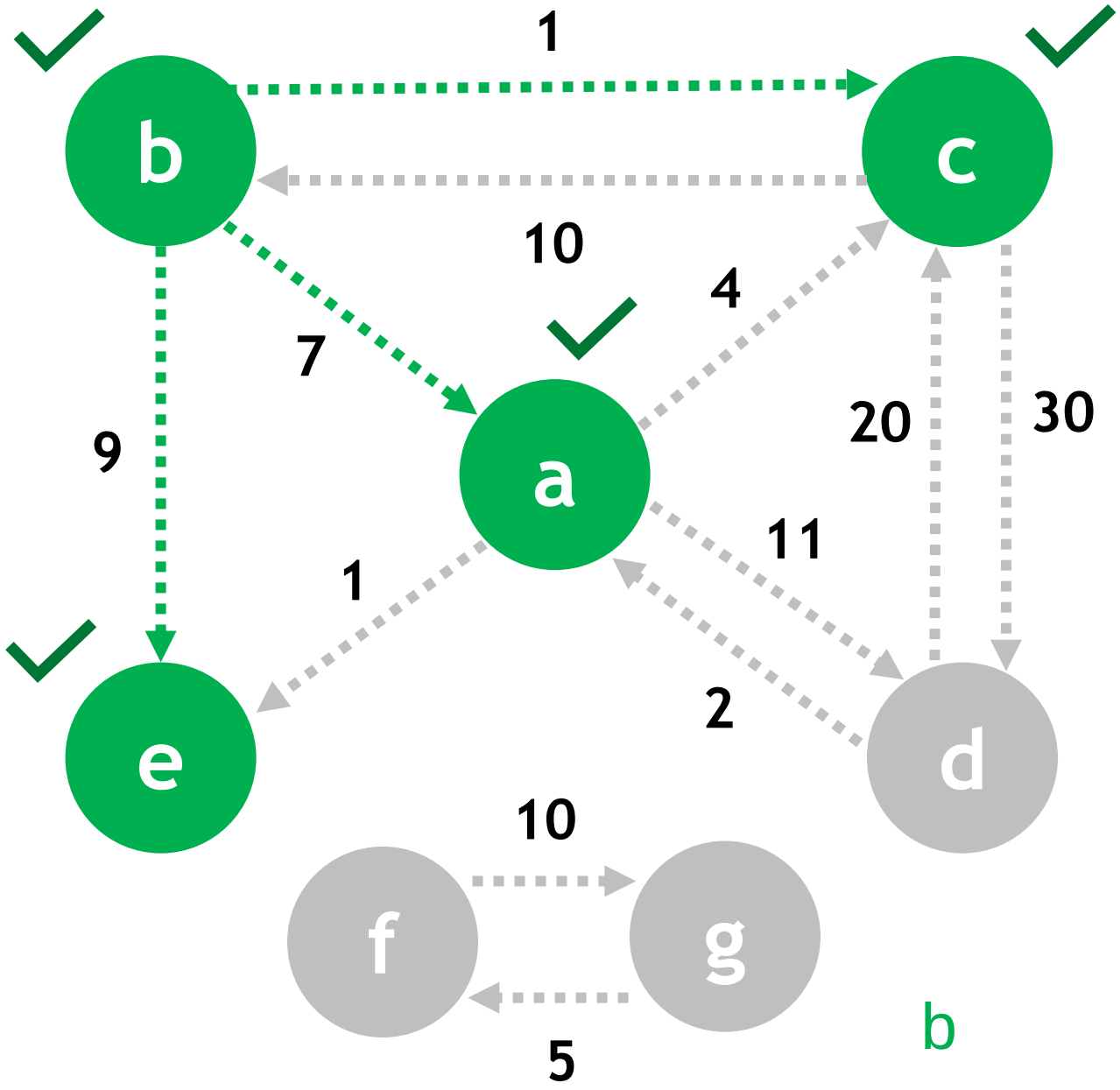
Print: b,a,c,e

b



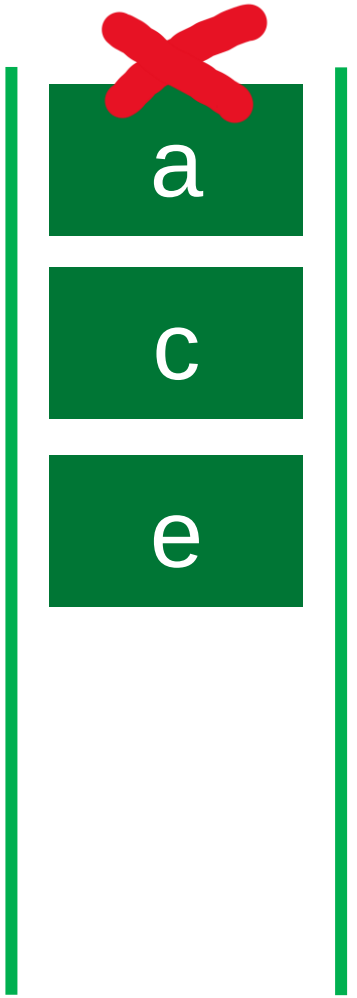
Fronta

Začiatok prehľadavánia z vrcholu b



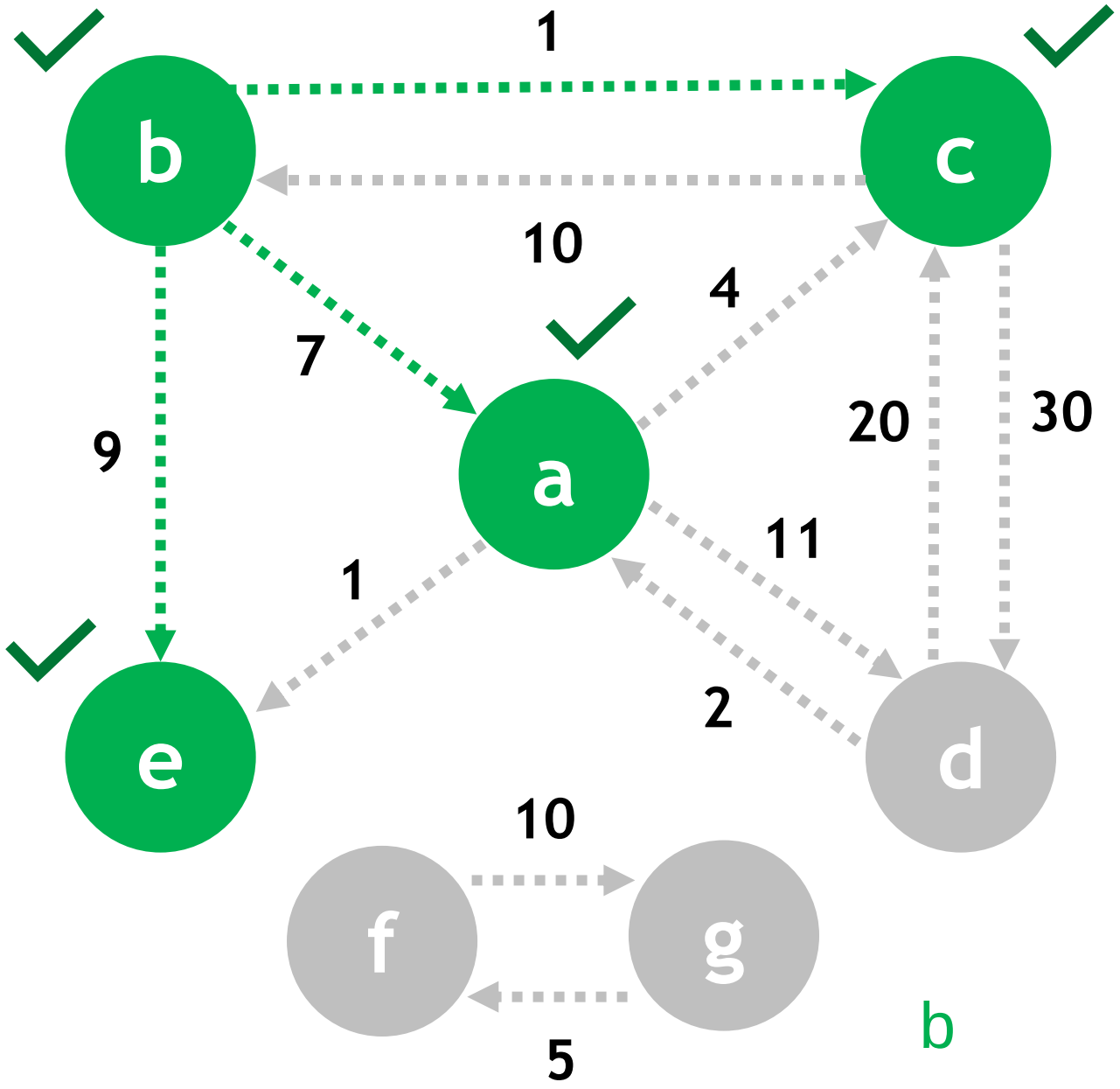
Print: b,a,c,e

b



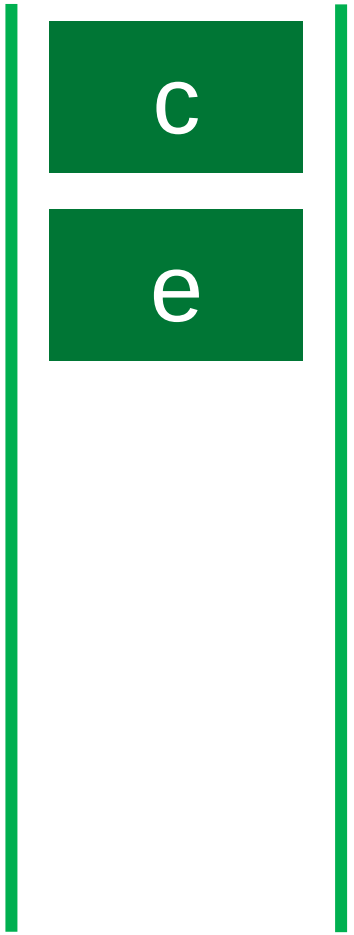
Fronta

Začiatok prehľadavánia z vrcholu b



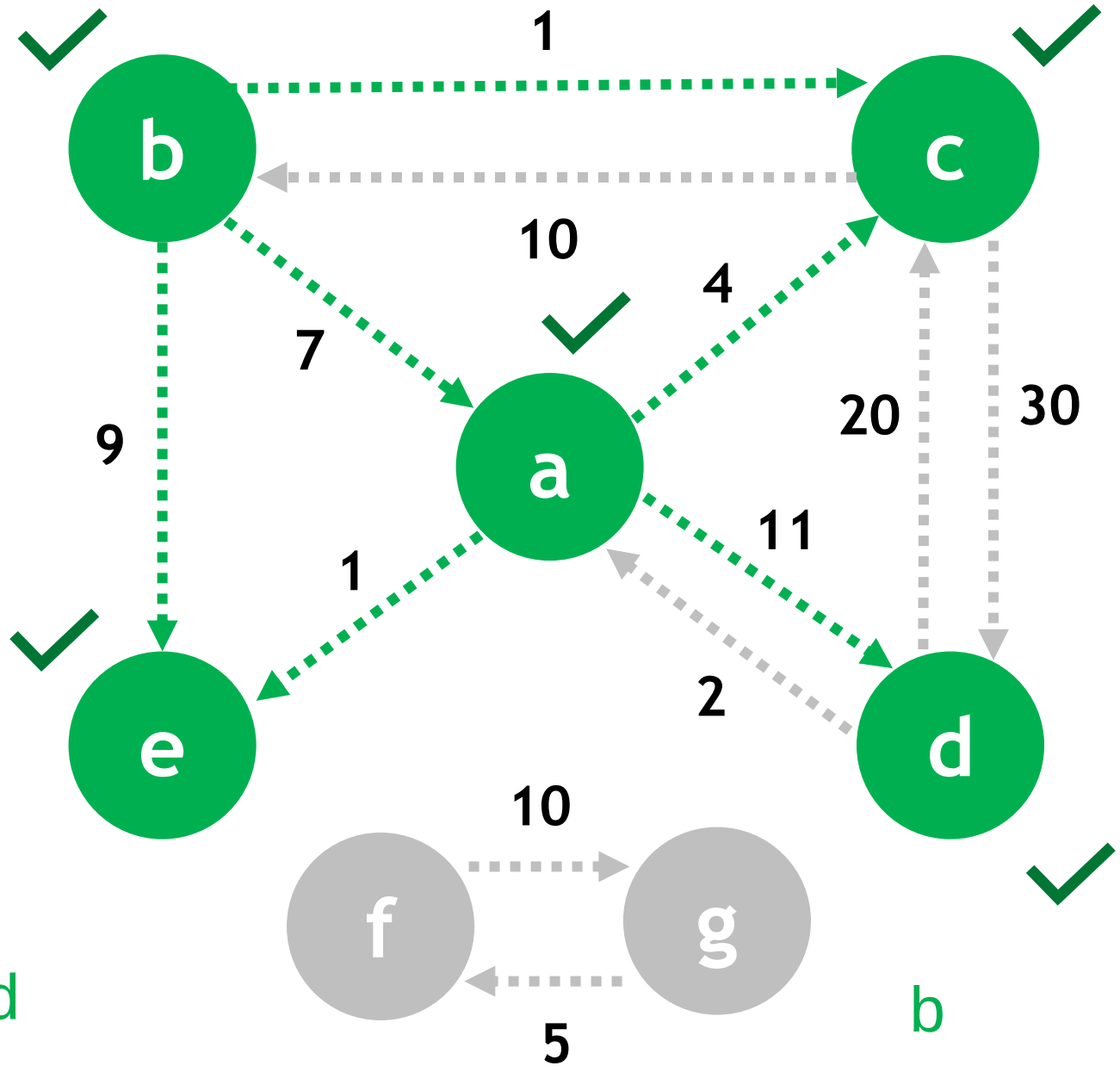
Print: b,a,c,e

b



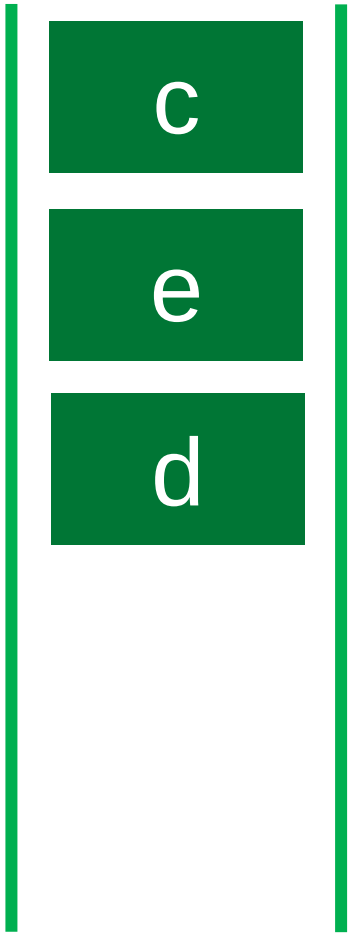
Fronta

Začiatok prehľadavánia z vrcholu b



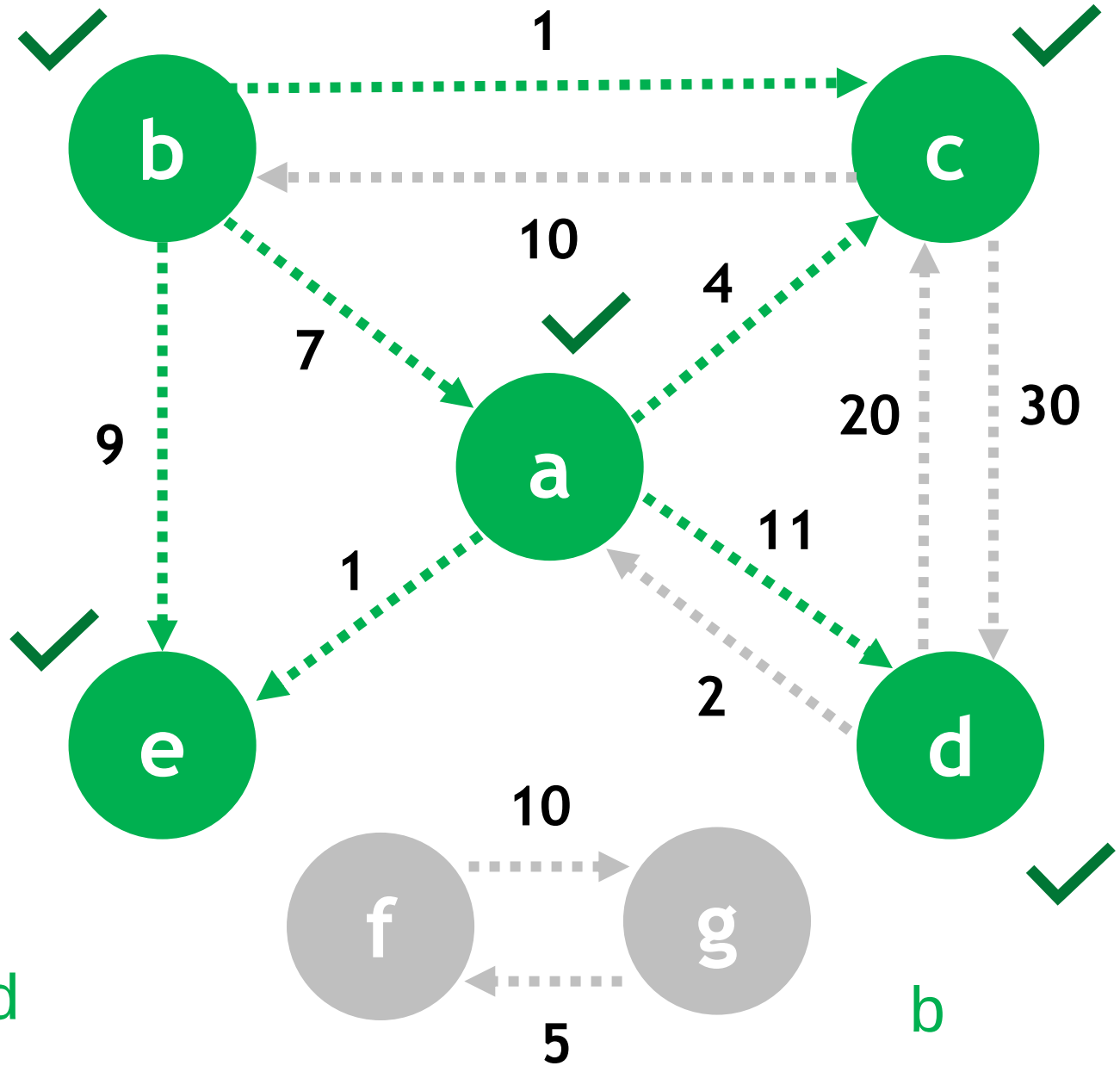
Print: b,a,c,e,d

b



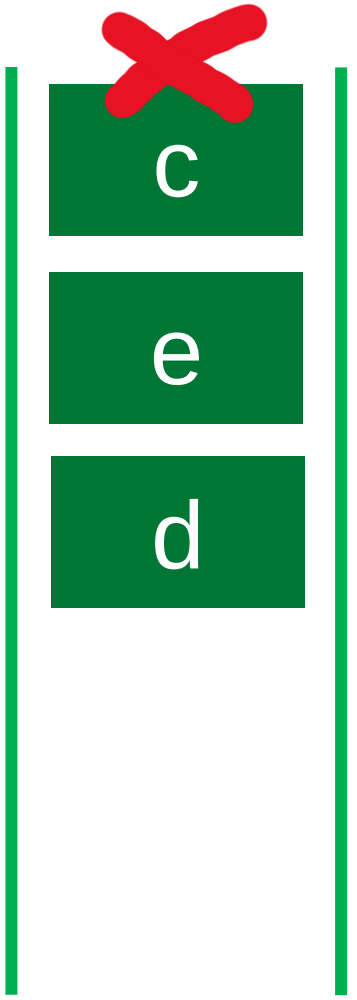
Fronta

Začiatok prehľadavánia z vrcholu b



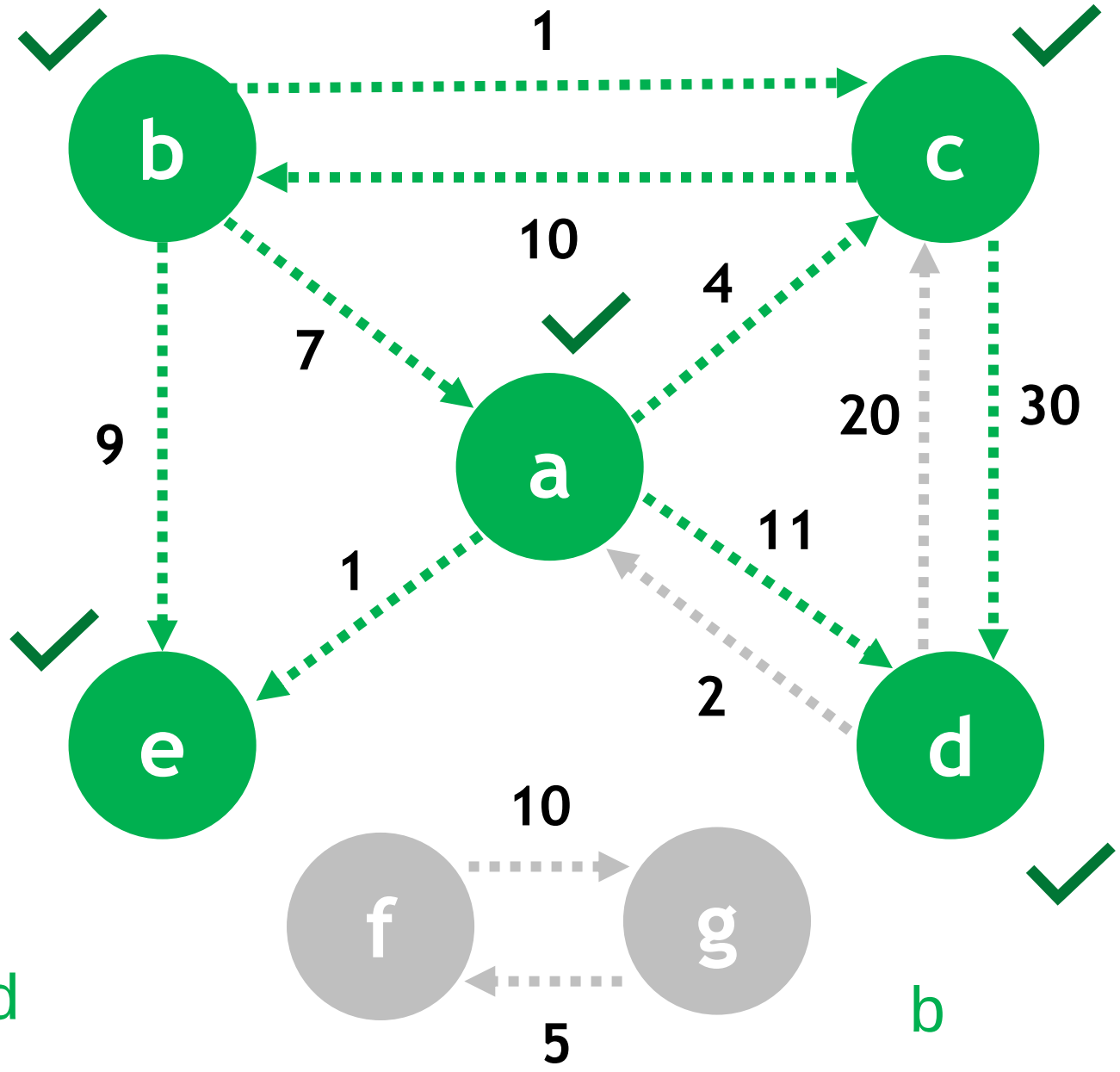
Print: b,a,c,e,d

b



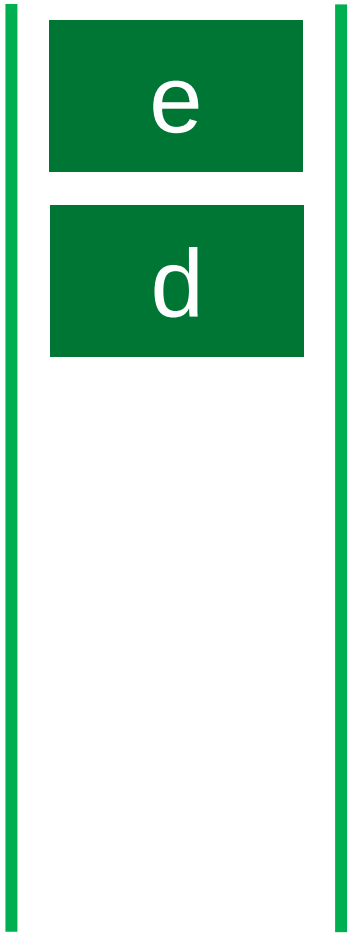
Fronta

Začiatok prehľadavánia z vrcholu b



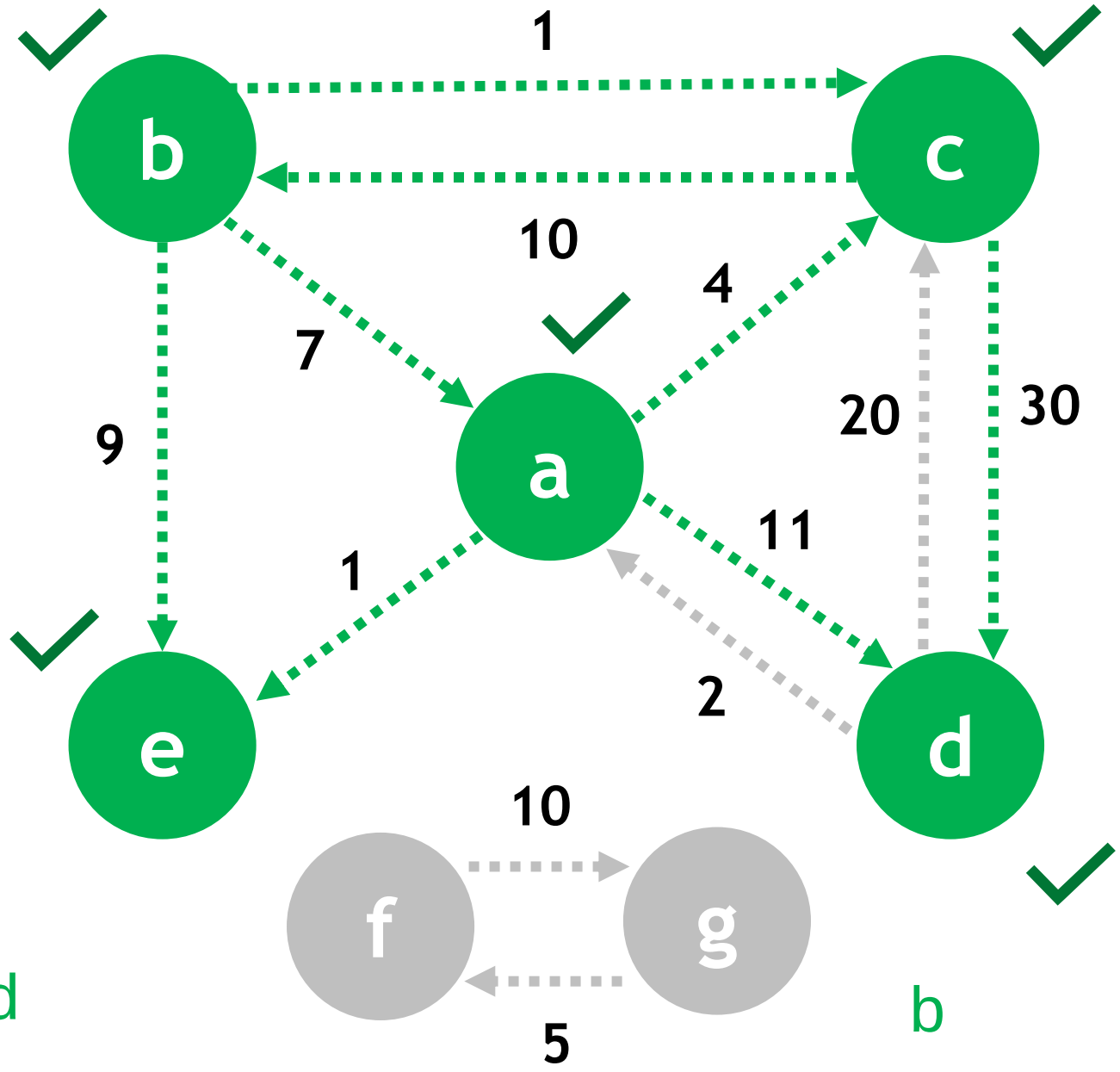
Print: b,a,c,e,d

b



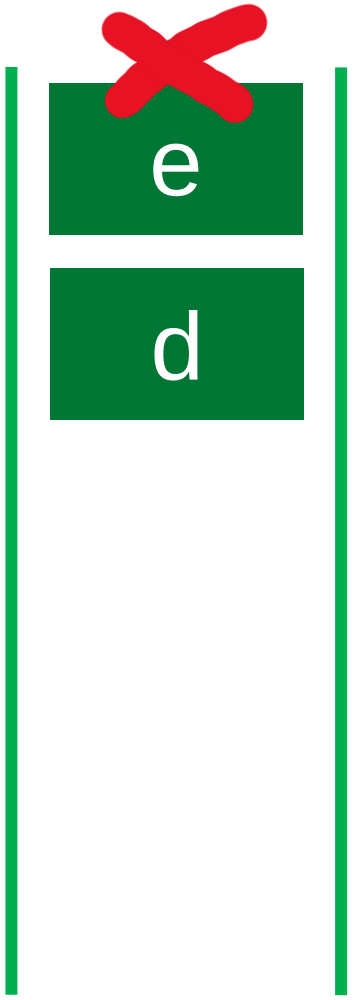
Fronta

Začiatok prehľadavánia z vrcholu b



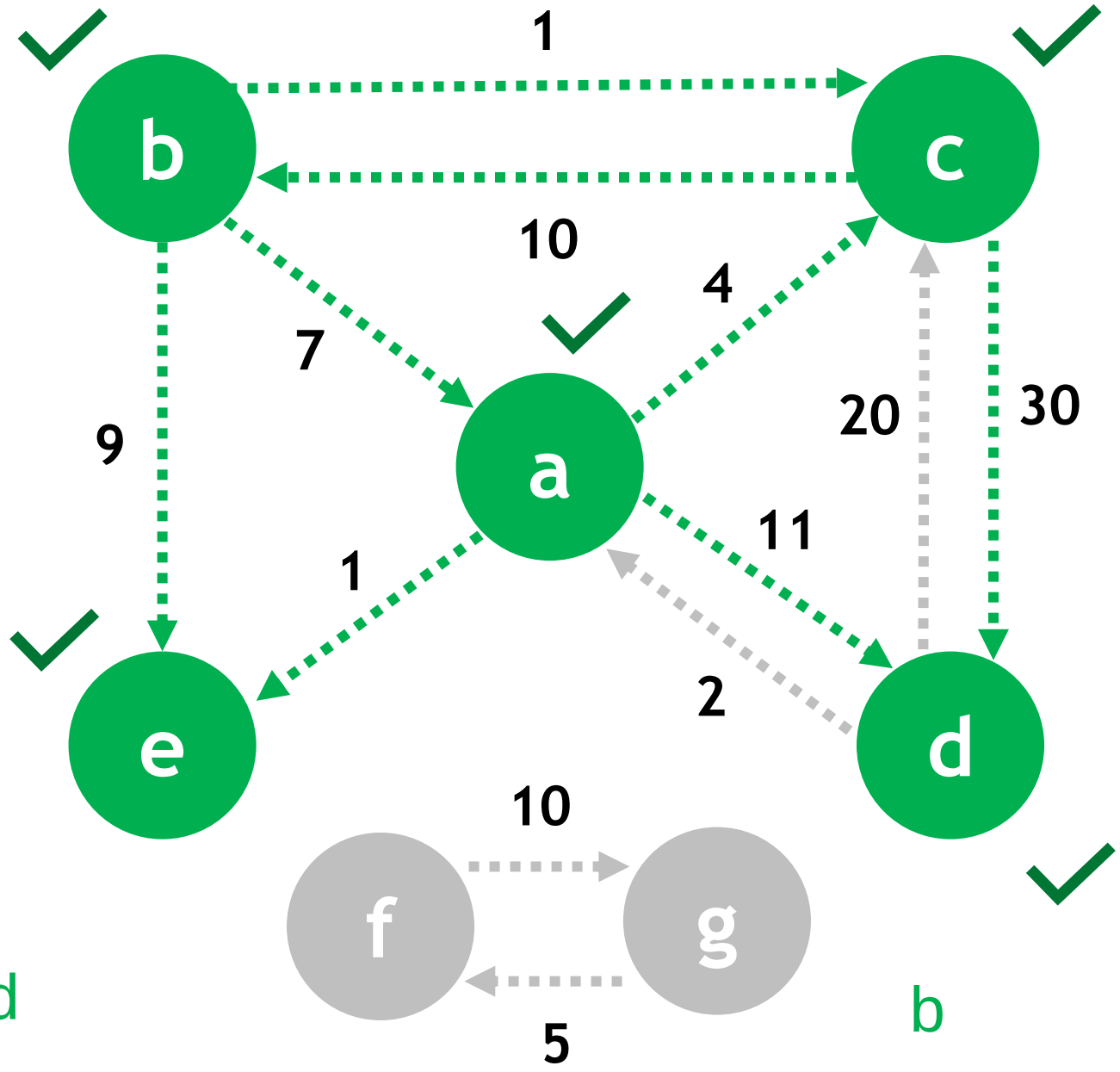
Print: b,a,c,e,d

b



Fronta

Začiatok prehľadavánia z vrcholu b



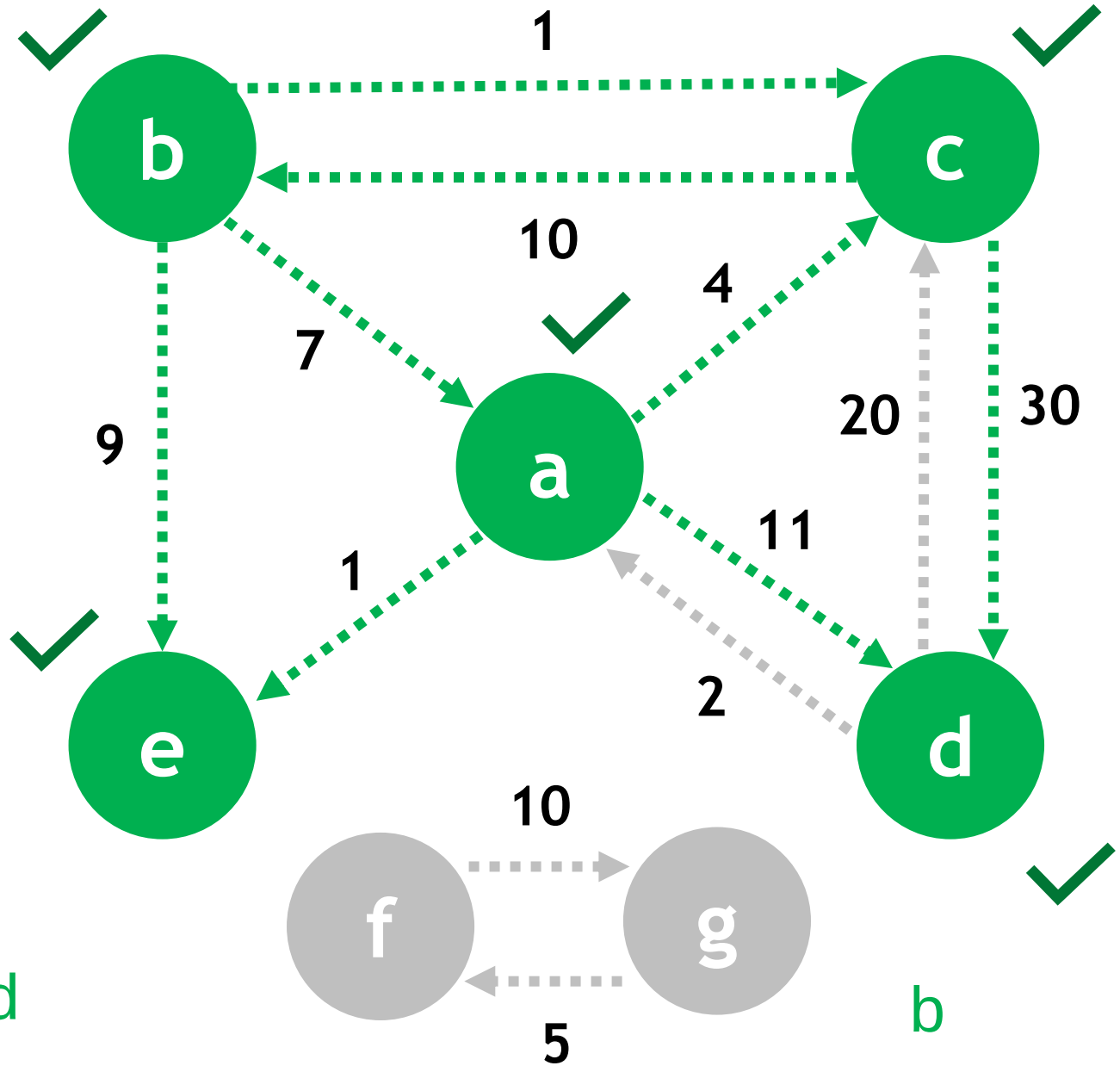
Print: b,a,c,e,d

b

d

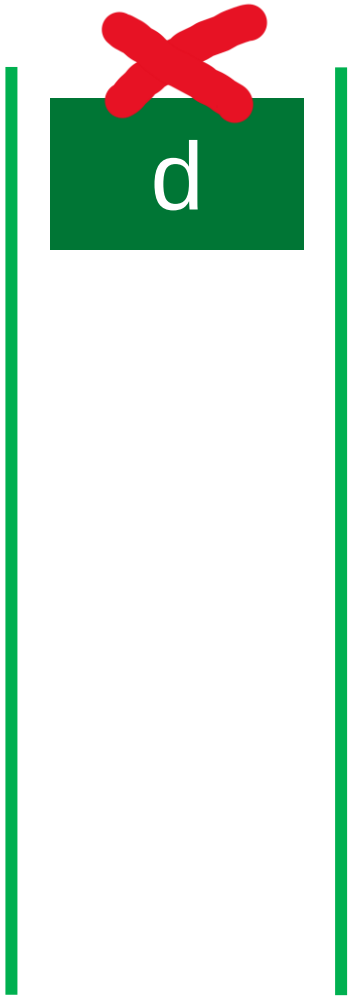
Fronta

Začiatok prehľadavánia z vrcholu b



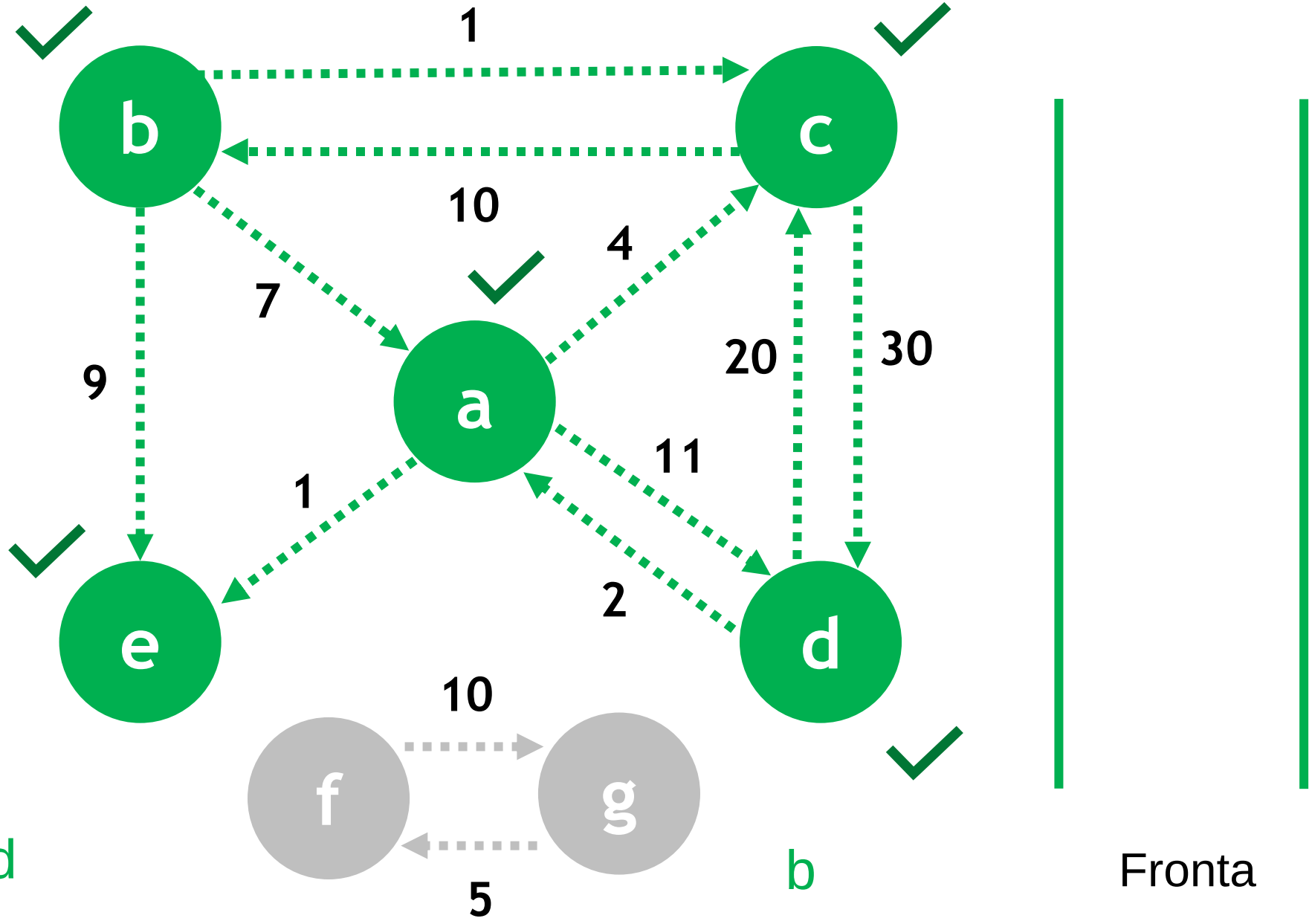
Print: b,a,c,e,d

b



Fronta

Začiatok prehľadavánia z vrcholu b



Print: b,a,c,e,d

b

Fronta

Ďalšie operácie v grafe

- Pridanie vrcholu/hrany
- Odstránenie vrcholu/hrany
- Zistenie, či existuje hrana medzi dvomi vrcholmi
- Zistenie, či existuje cesta medzi dvoma vrcholmi
- Vstupný/výstupný stupeň vrcholu
- Polomer/priemer grafu
- Najkratšia cesta zo zvoleného vrcholu k ostatným vrcholom (Dijkstra)

Vzorová implementácia v C/C++