BELLMAN-FORD

Shortest path Algorithm

SHAH EMAAD : 21CE10061
SASUMANA CHANDANA: 21CE10060

Overview:

1. non-greedy algorithm
2. Time complexity : O(V.E)

Assumptions

- Cycles are non-negative
- All paths are simple
- At most V -1 iterations

When to use

Works for negative edge weights

Dosen't work for negative cycles

← BACK TO AGENDA PAGE

Pseudo code

```
function bellmanFord(G, S)
 for each vertex V in G
  distance[V] <- infinite
   previous[V] <- NULL
 distance[S] <- 0

 for each vertex V in G
  for each edge (U,V) in G
   tempDistance <- distance[U] + edge_weight(U, V)
   if tempDistance < distance[V]
    distance[V] <- tempDistance
    previous[V] <- U

 for each edge (U,V) in G
  If distance[U] + edge_weight(U, V) < distance[V}
   Error: Negative Cycle Exists

 return distance[], previous[]
```

Example

A cab driver in a city want to know which path is most profitable so that he can increase his rides in that particular path. He even choose a source S and wants the most profitable paths

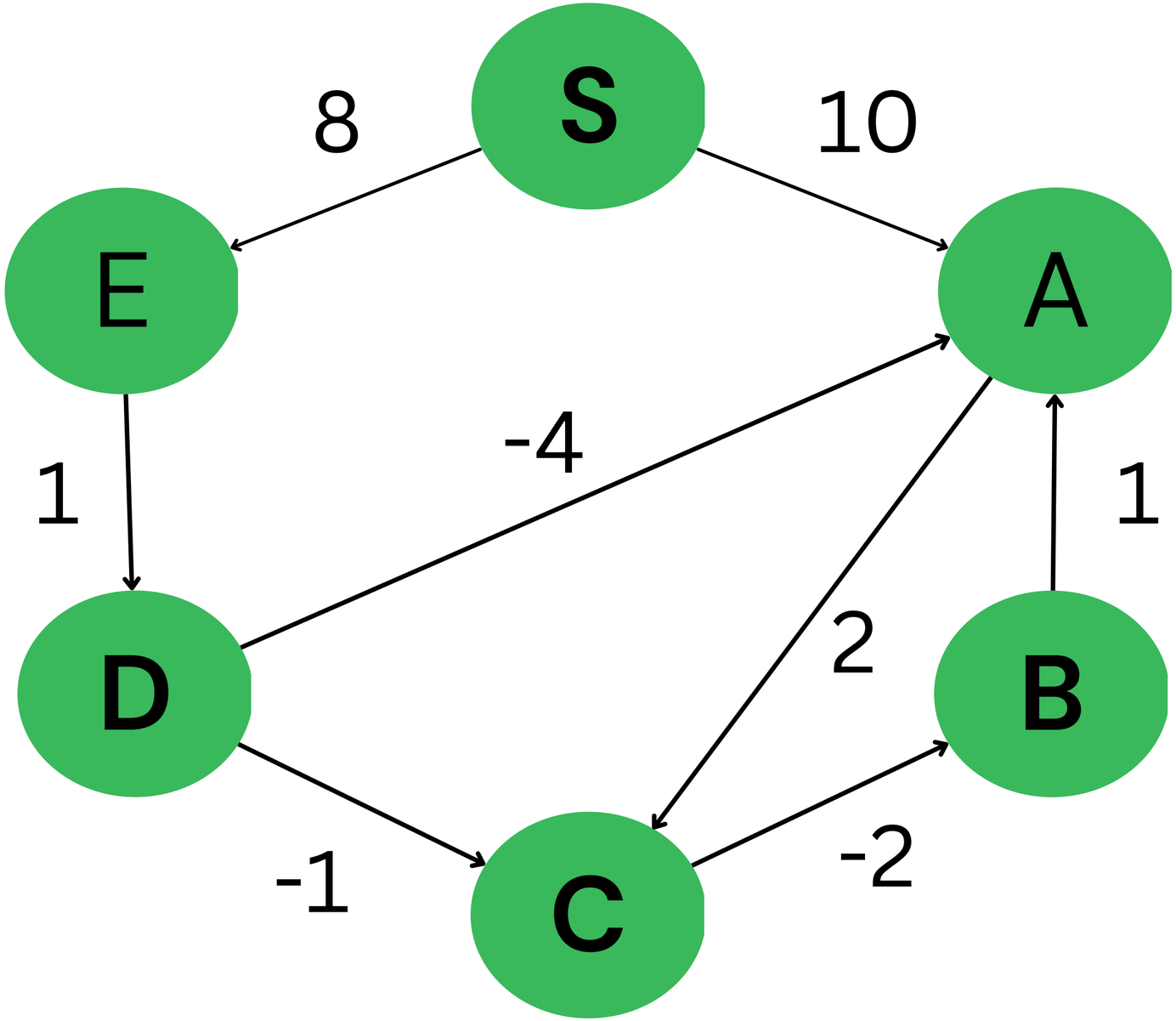
Plan

so to solve this problem we build a graph of possible cost and when ever there is profit we indicate it as negative weight because cost reduce in that particular path

Example

Iteration: 0

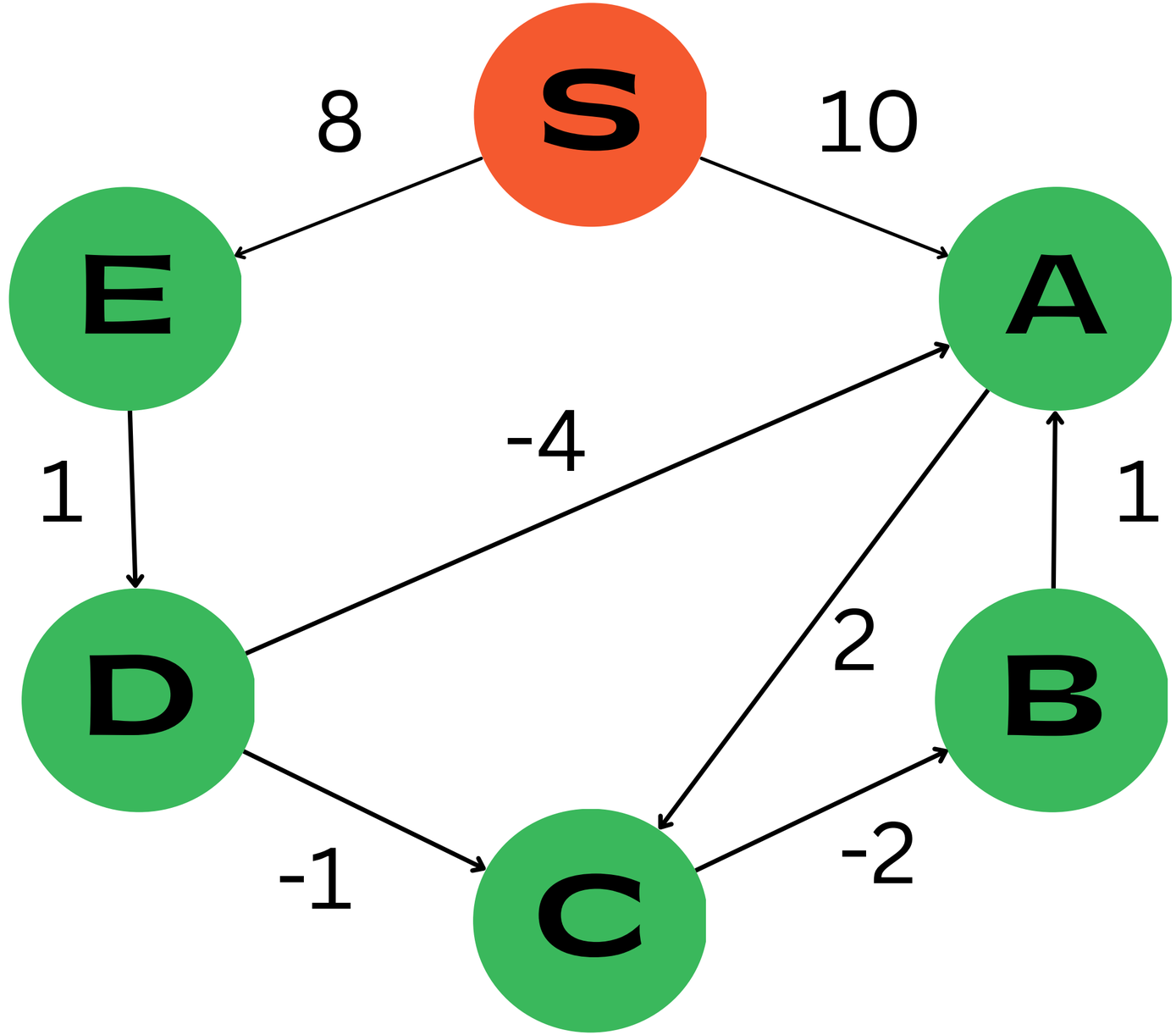
S	0
A	∞
B	∞
C	∞
D	∞
E	∞



← BACK TO AGENDA PAGE

Iteration: 1

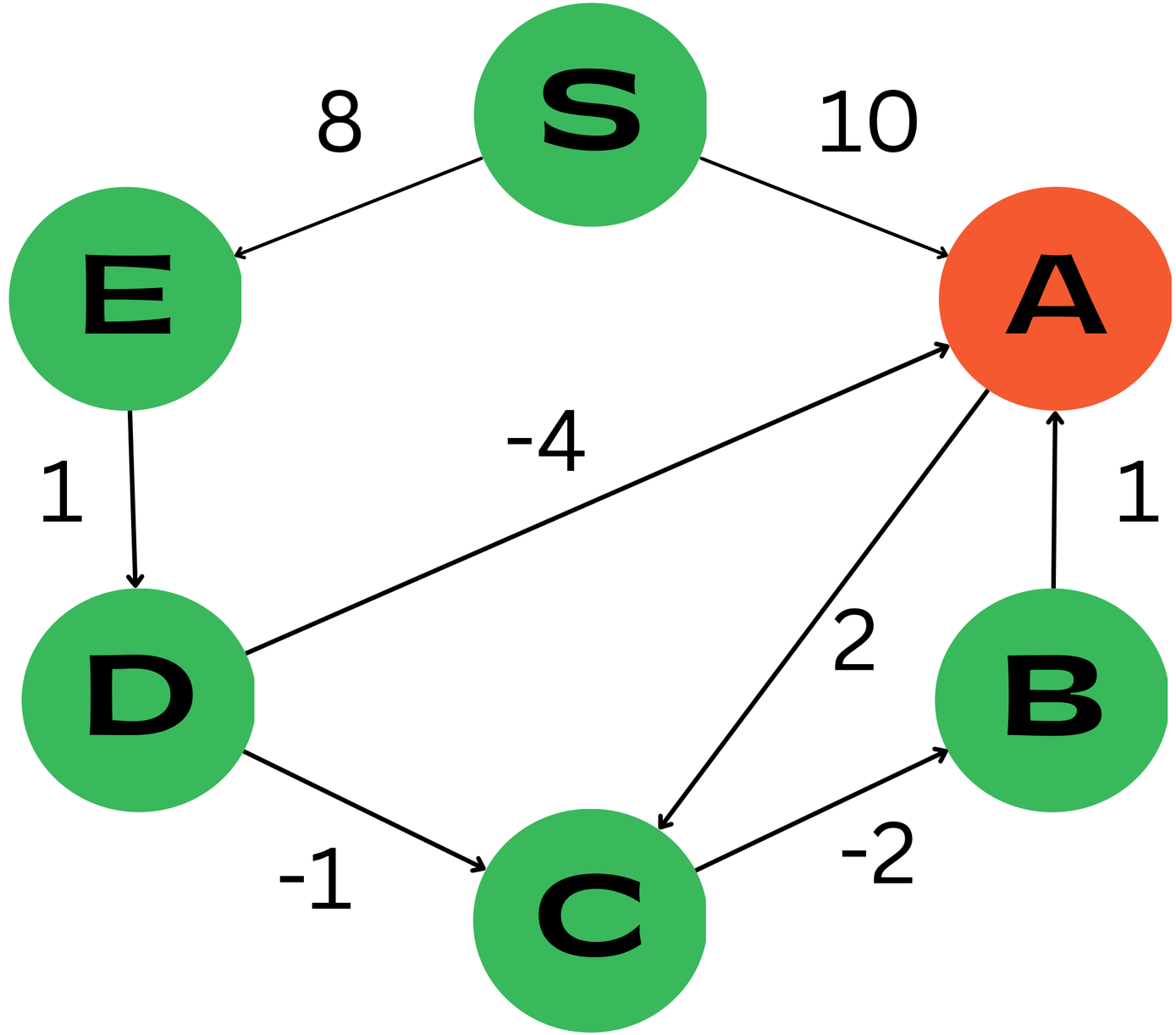
S	0
A	10
B	∞
C	∞
D	∞
E	8



← BACK TO AGENDA PAGE

Iteration: 1

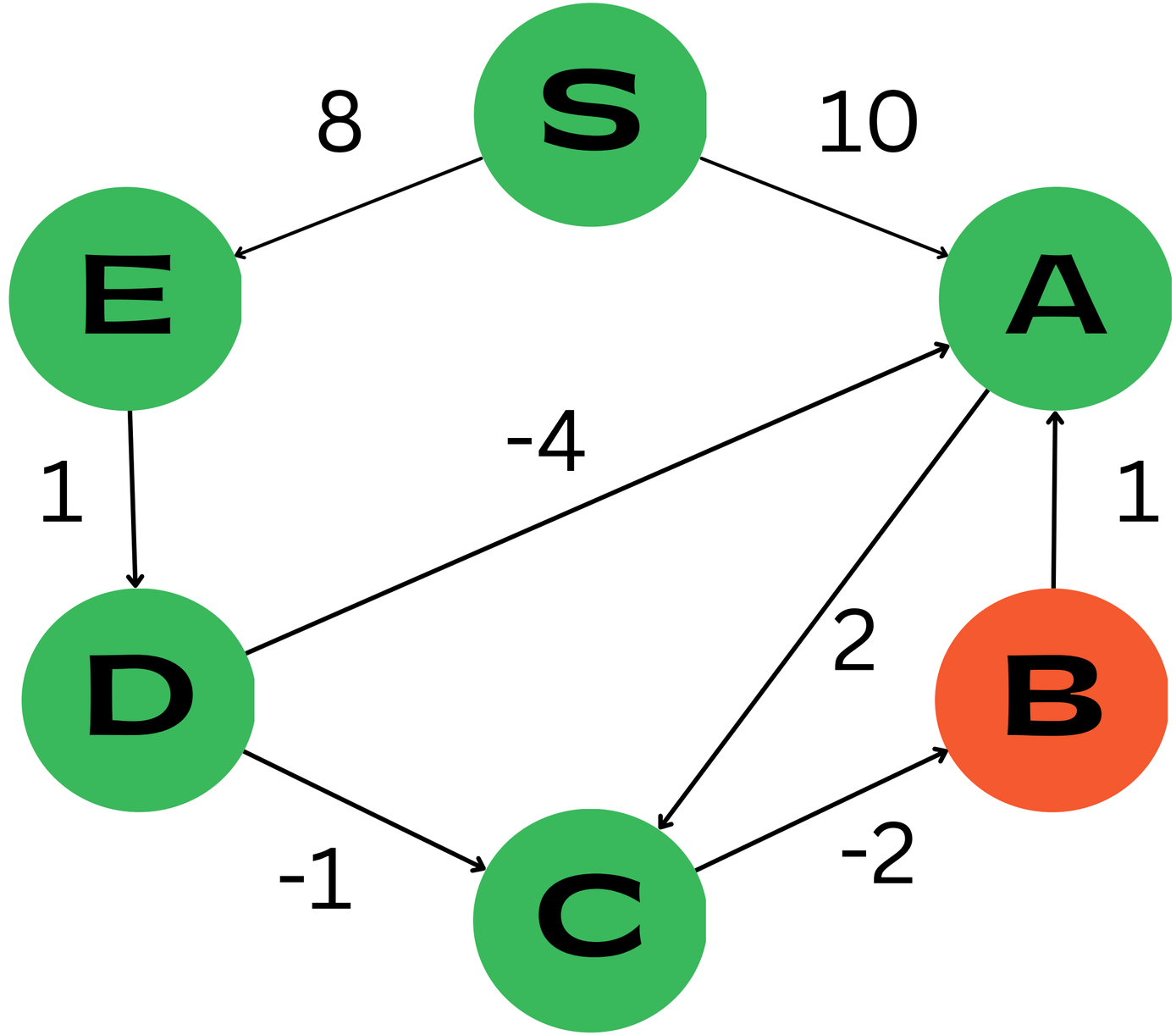
S	0
A	10
B	∞
C	12
D	∞
E	8



← BACK TO AGENDA PAGE

Iteration: 1

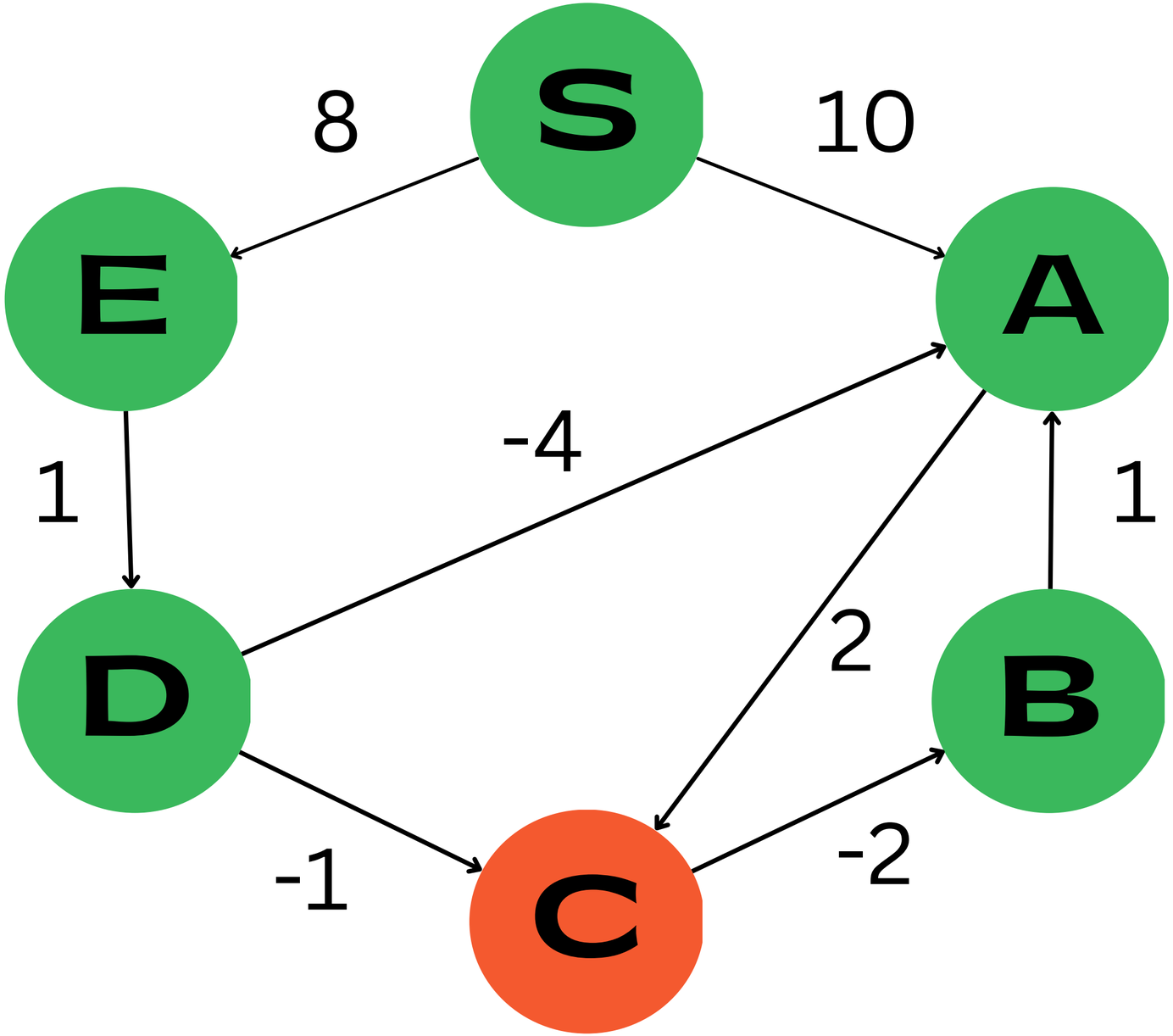
S	0
A	10
B	∞
C	12
D	∞
E	8



← BACK TO AGENDA PAGE

Iteration: 1

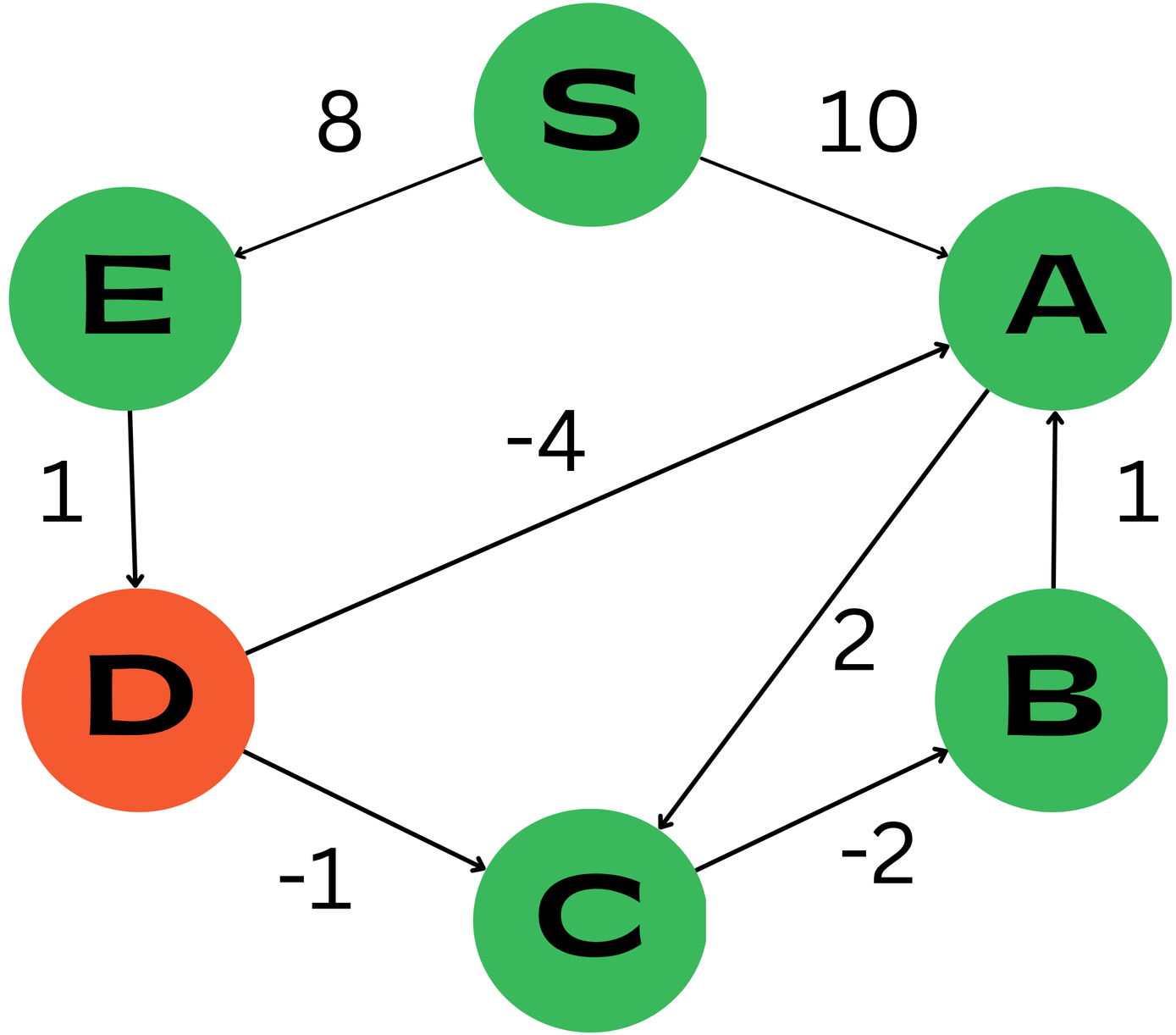
S	0
A	10
B	10
C	12
D	∞
E	8



← BACK TO AGENDA PAGE

Iteration: 1

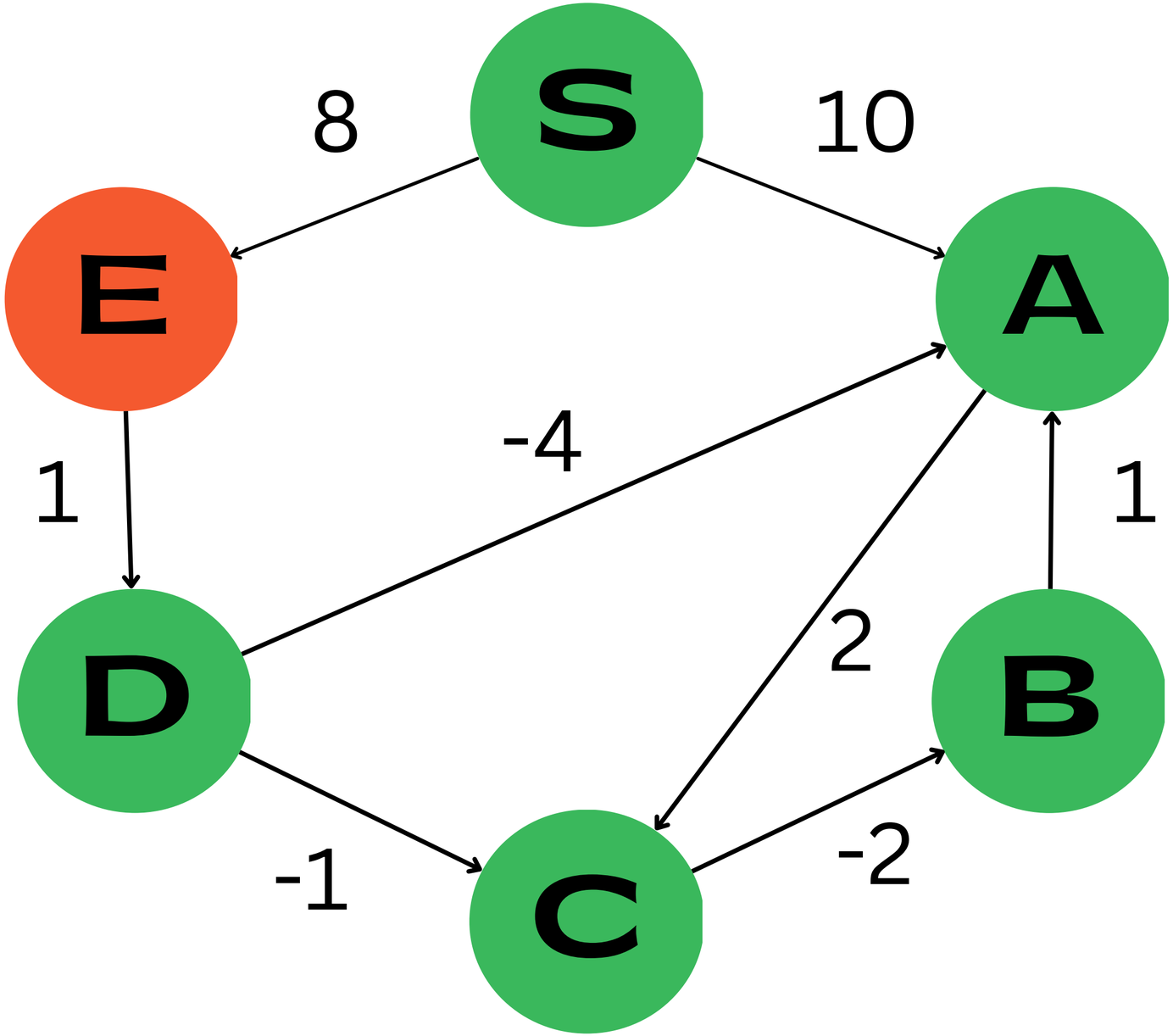
S	0
A	10
B	10
C	12
D	∞
E	8



← BACK TO AGENDA PAGE

Iteration: 1

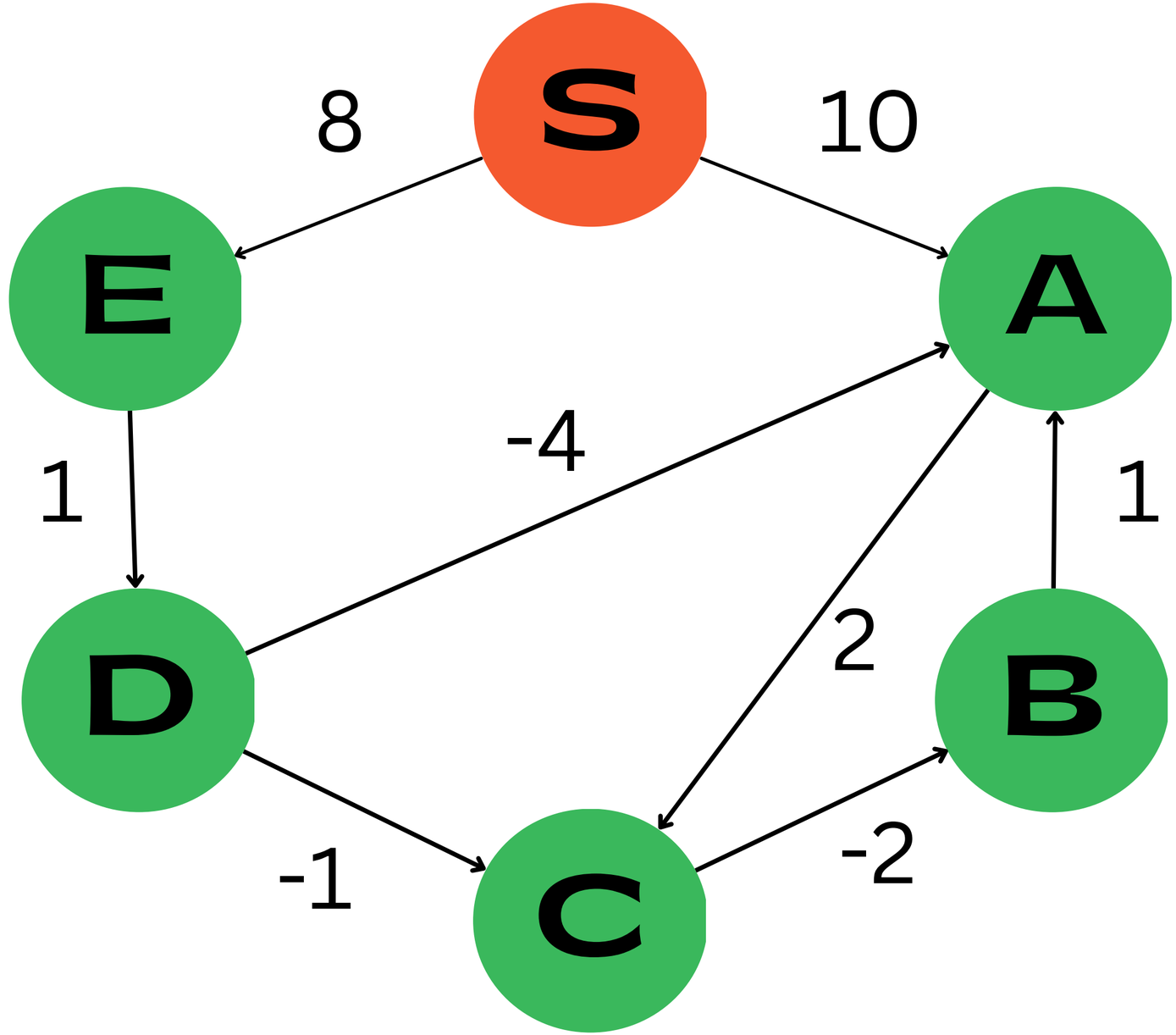
S	0
A	10
B	10
C	12
D	9
E	8



← BACK TO AGENDA PAGE

Iteration: 2

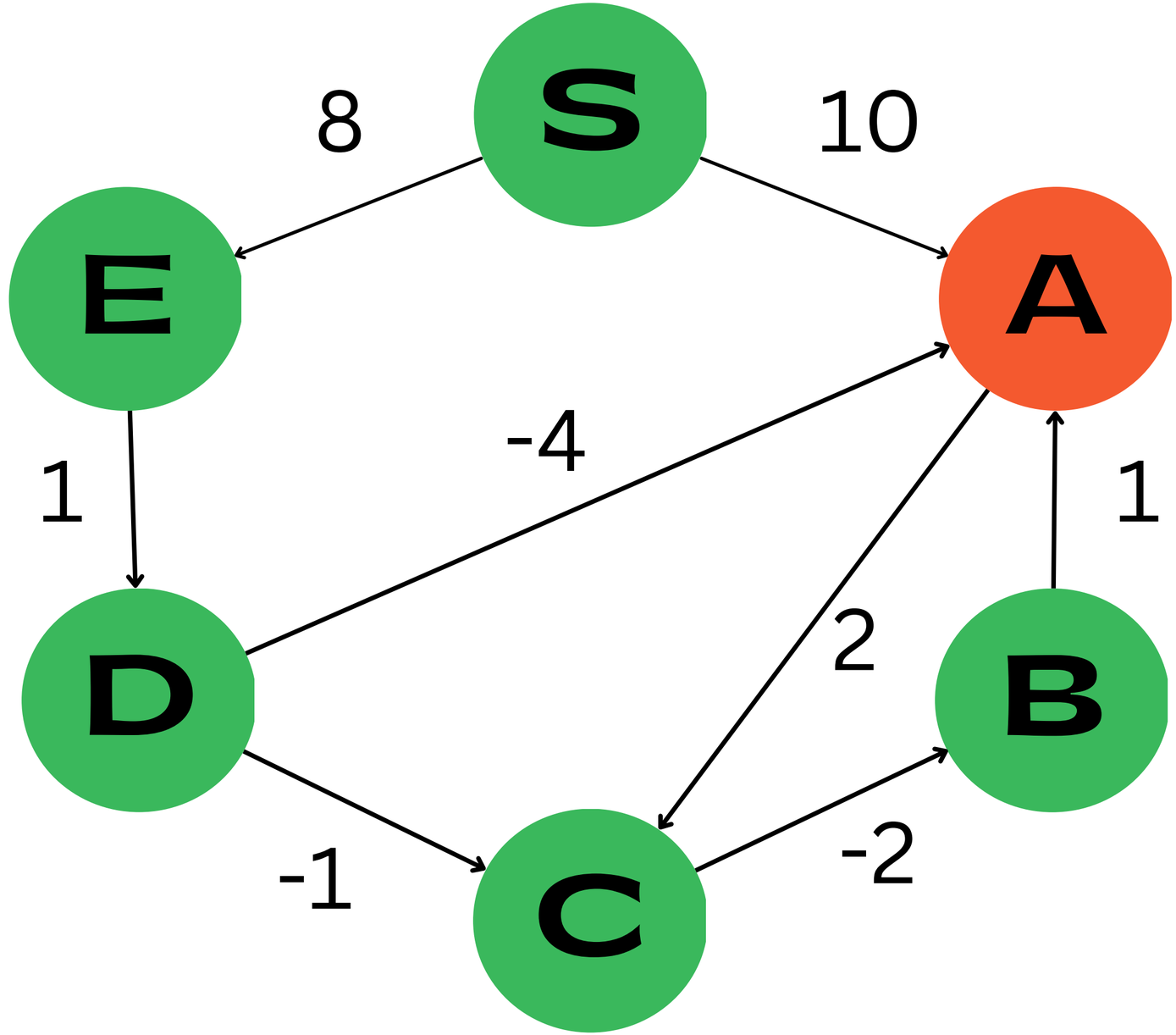
S	0
A	10
B	10
C	12
D	9
E	8



← BACK TO AGENDA PAGE

Iteration: 2

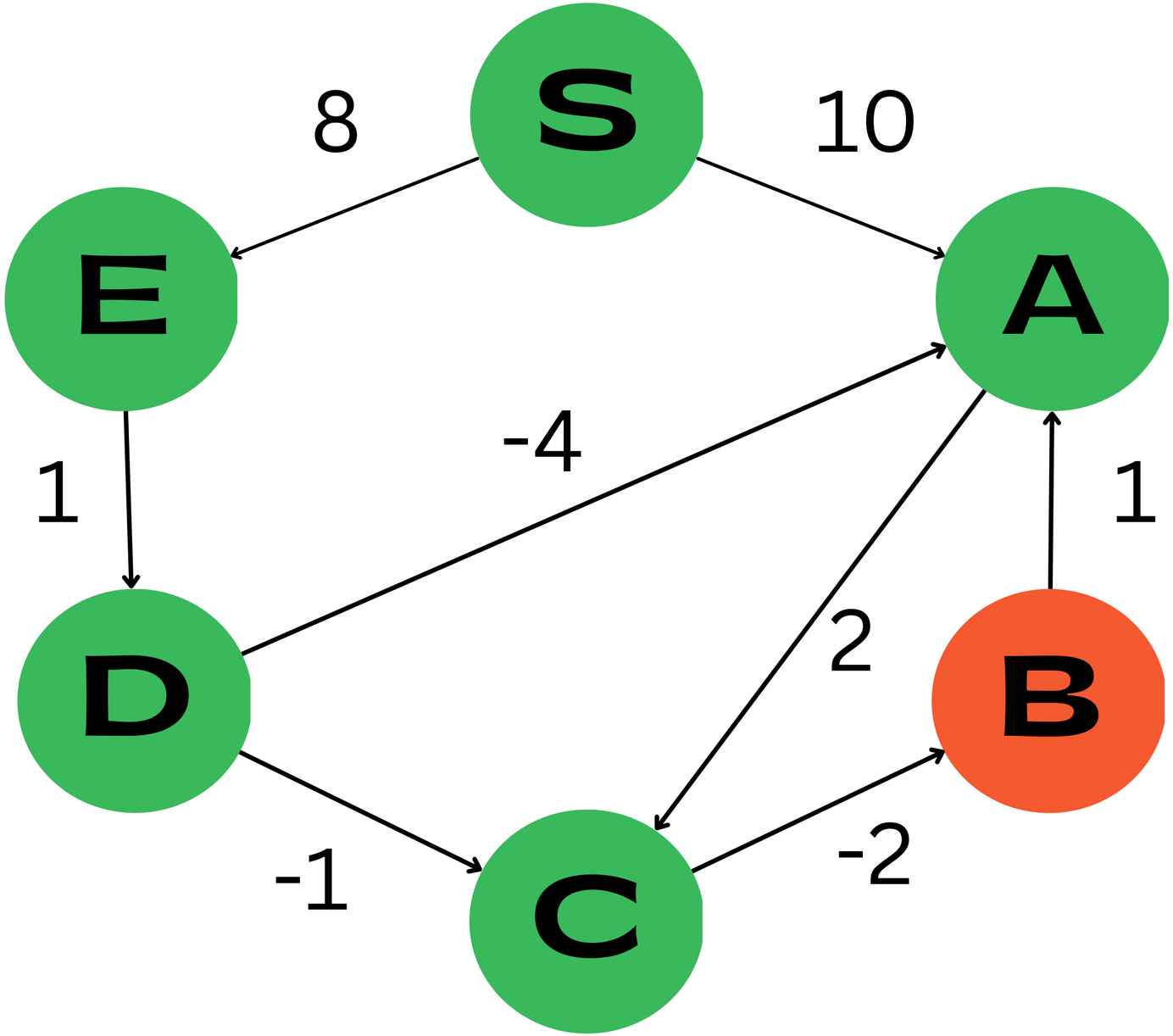
S	0
A	10
B	10
C	12
D	9
E	8



← BACK TO AGENDA PAGE

Iteration: 2

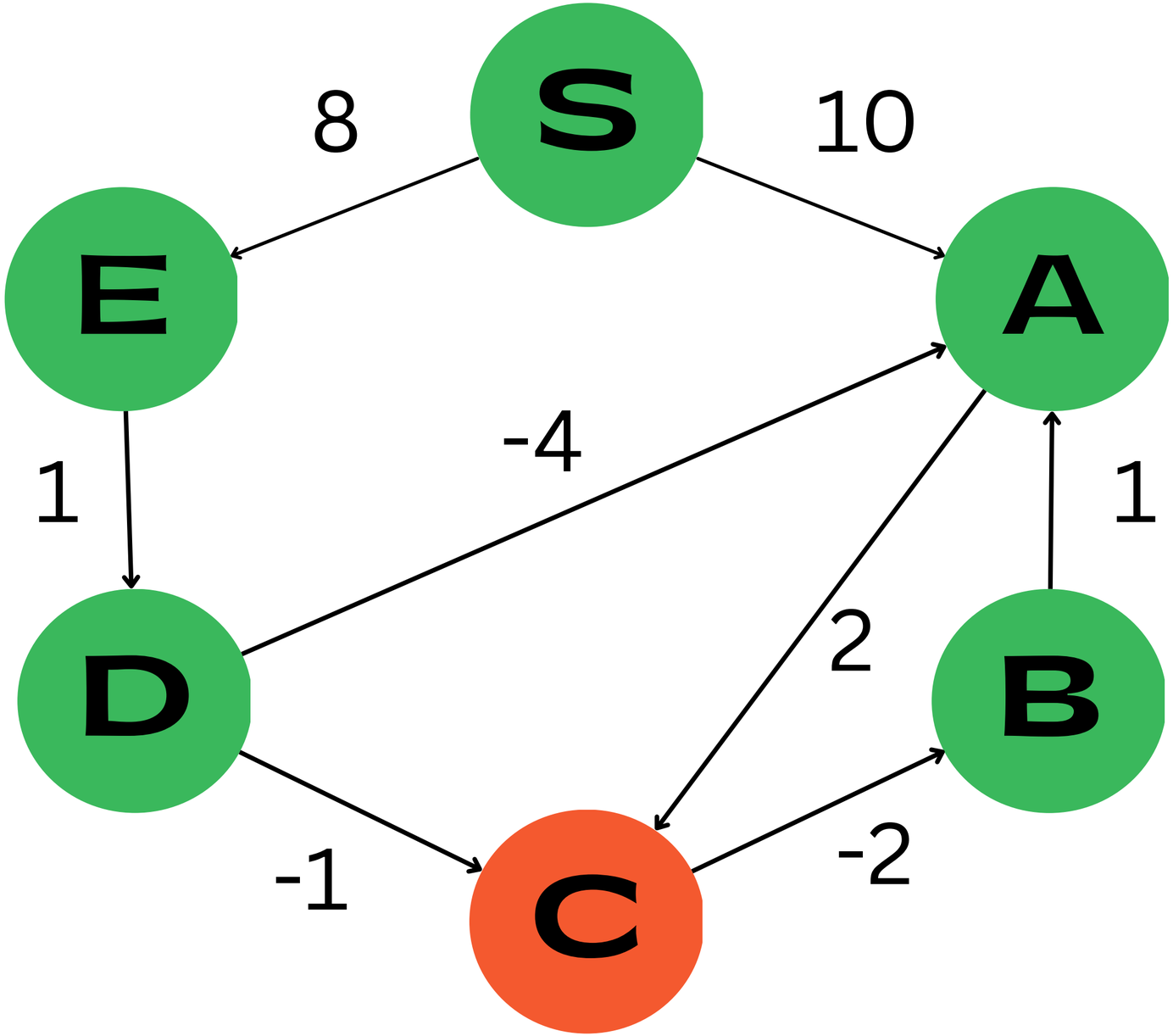
S	0
A	10
B	10
C	12
D	9
E	8



← BACK TO AGENDA PAGE

Iteration: 2

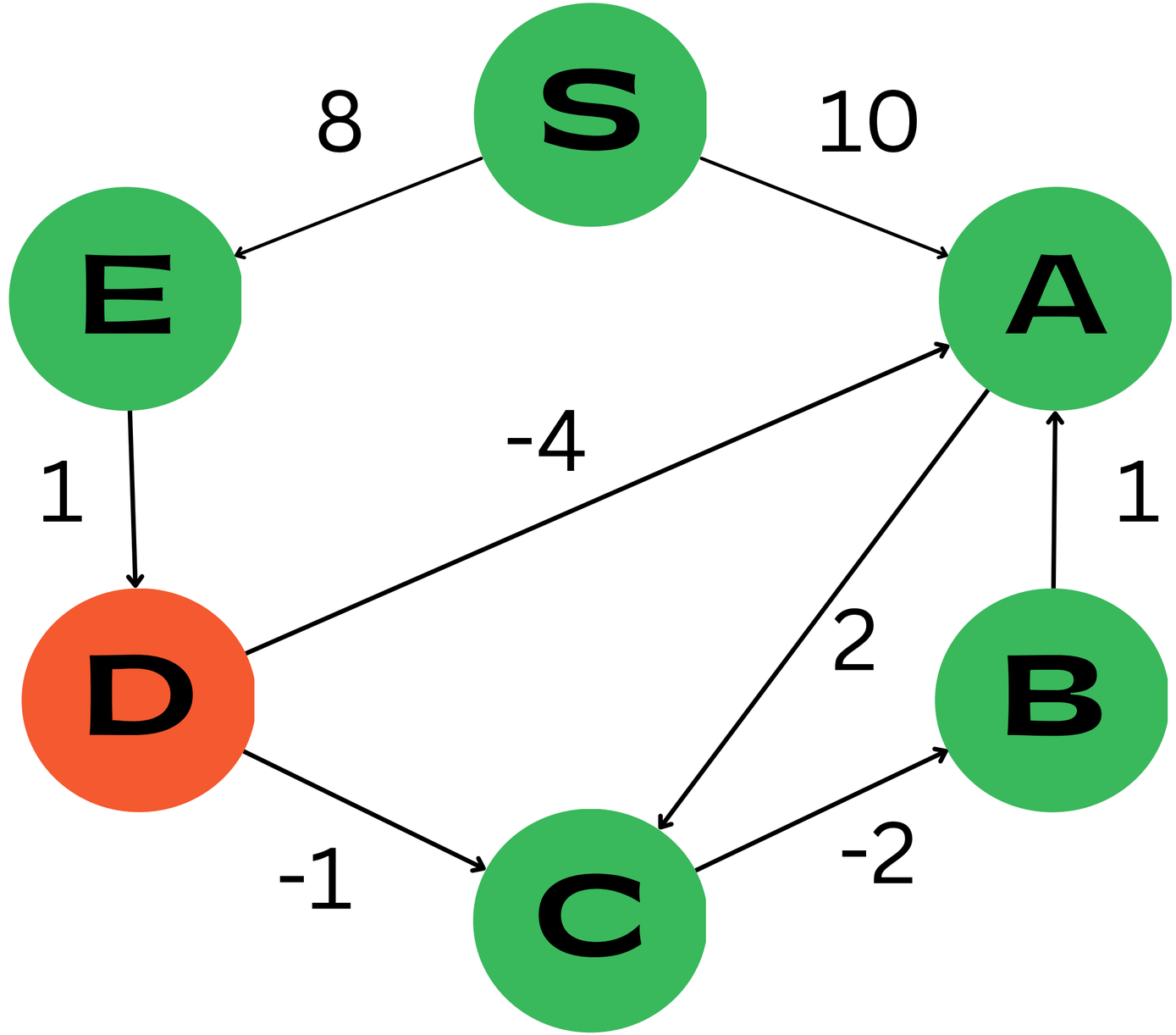
S	0
A	10
B	10
C	12
D	9
E	8



← BACK TO AGENDA PAGE

Iteration: 2

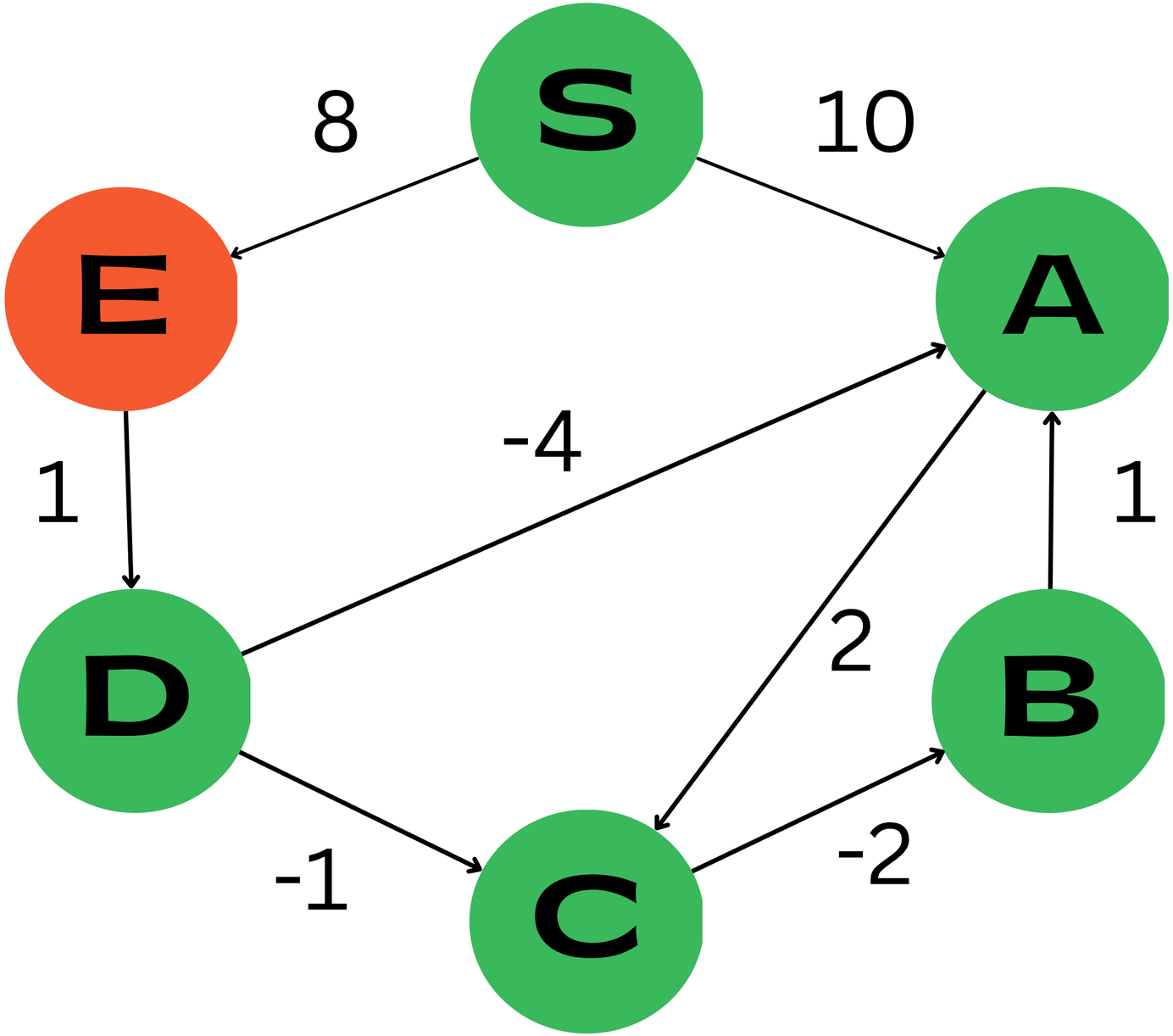
S	0
A	5
B	10
C	8
D	9
E	8



← BACK TO AGENDA PAGE

Iteration: 2

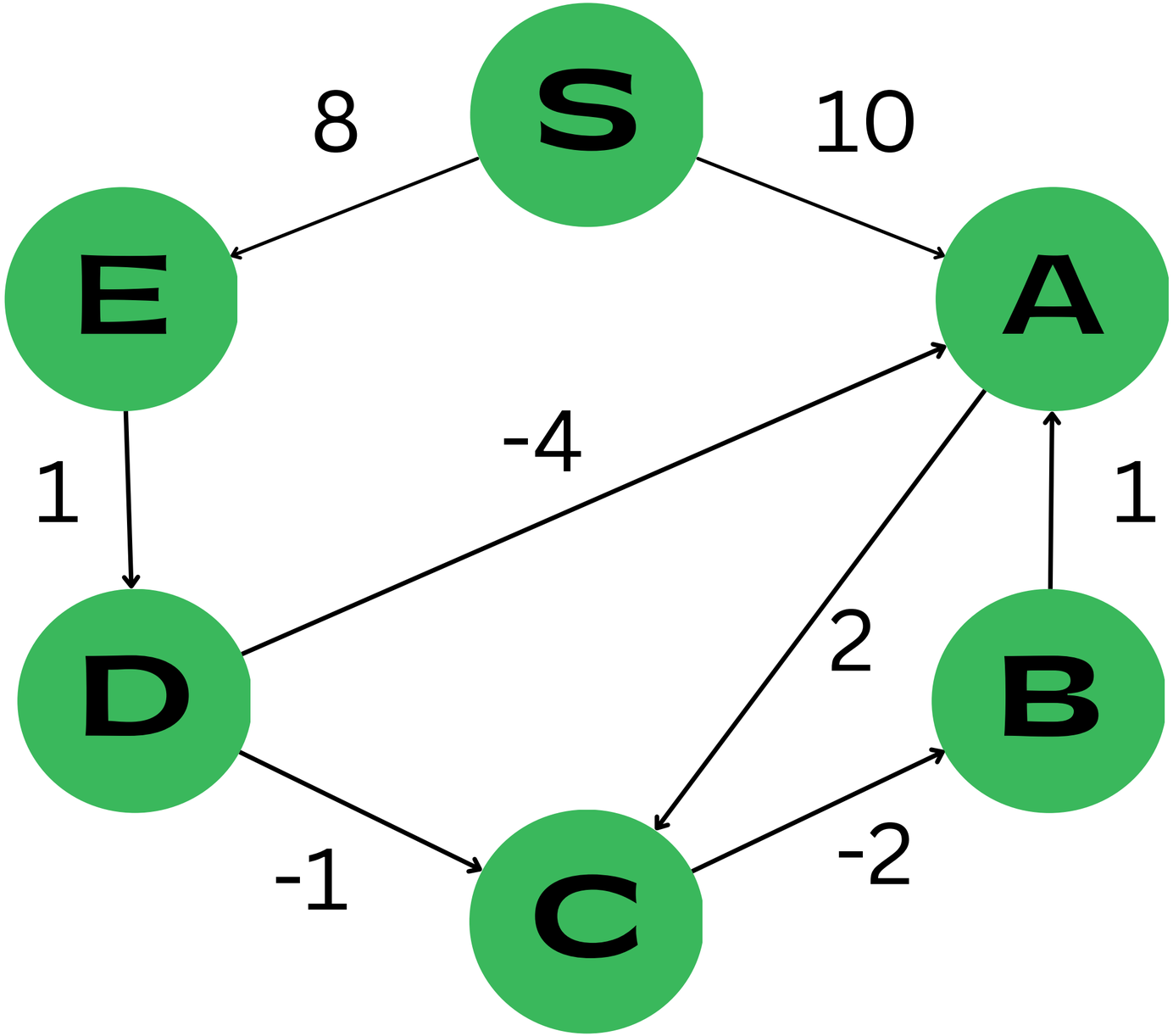
S	0
A	5
B	10
C	8
D	9
E	8



← BACK TO AGENDA PAGE

Iteration: 3

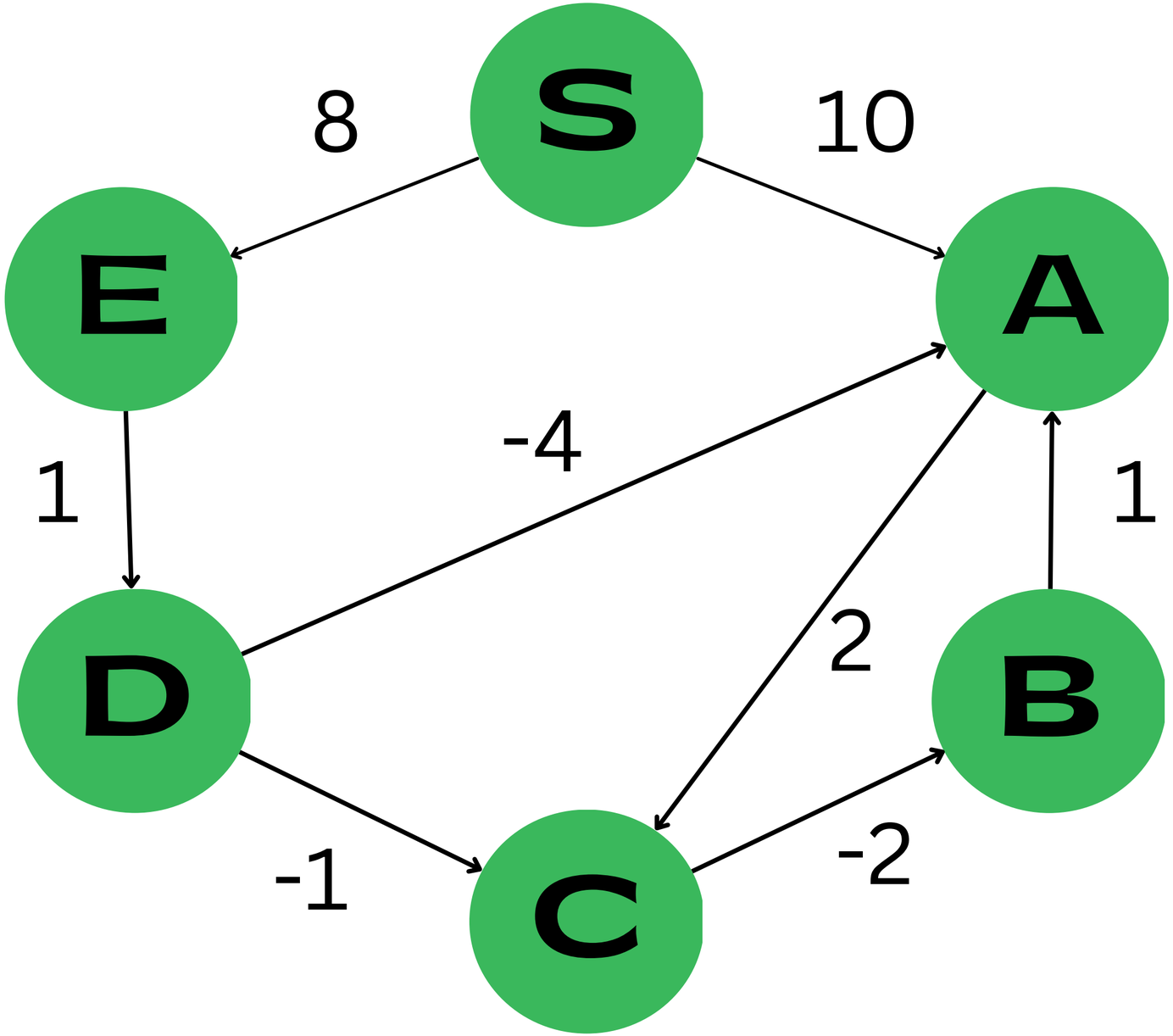
S	0
A	5
B	5
C	7
D	9
E	8



← BACK TO AGENDA PAGE

Iteration: 4

S	0
A	5
B	5
C	7
D	9
E	8



← BACK TO AGENDA PAGE

Thank You