2.17 Week 2 Homework Quiz

Kevin Offemaria (username: offemakp)

Attempt 7

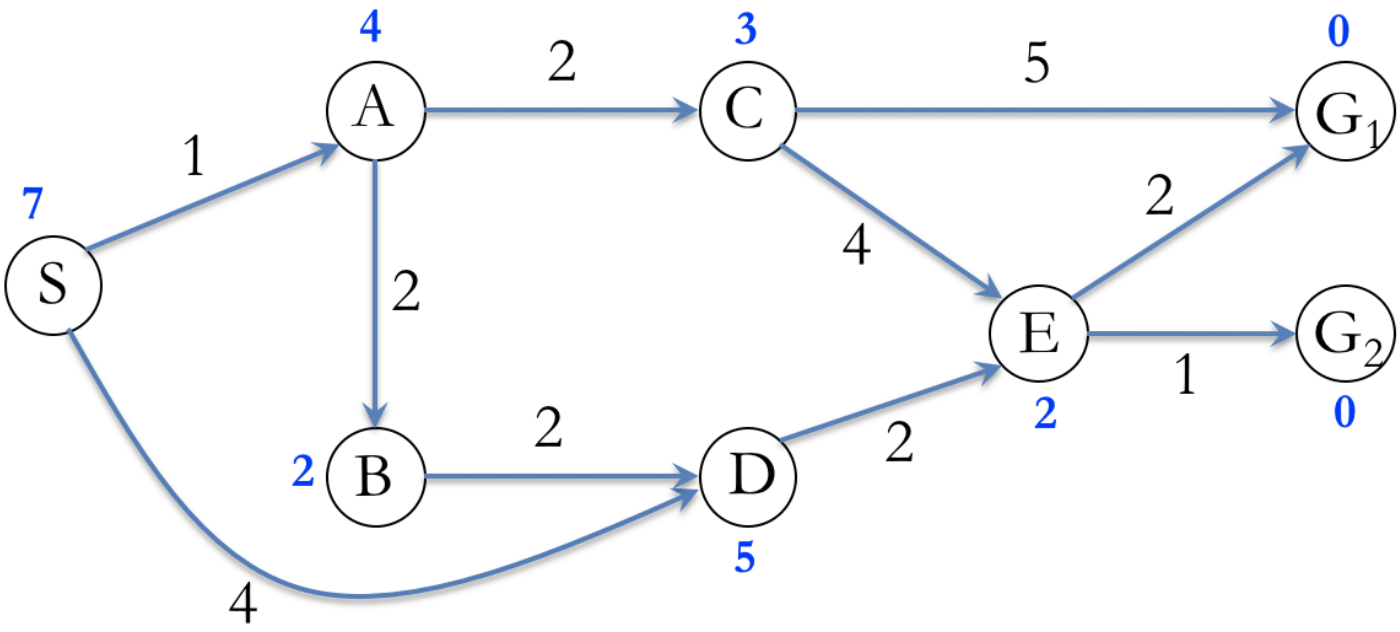
Written: Jan 21, 2023 8:13 AM - Jan 21, 2023 8:16 AM

Submission View

Your quiz has been submitted successfully.

Question 1 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***heuristic depth first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited. In this question, assume that "reached" is NOT used.***

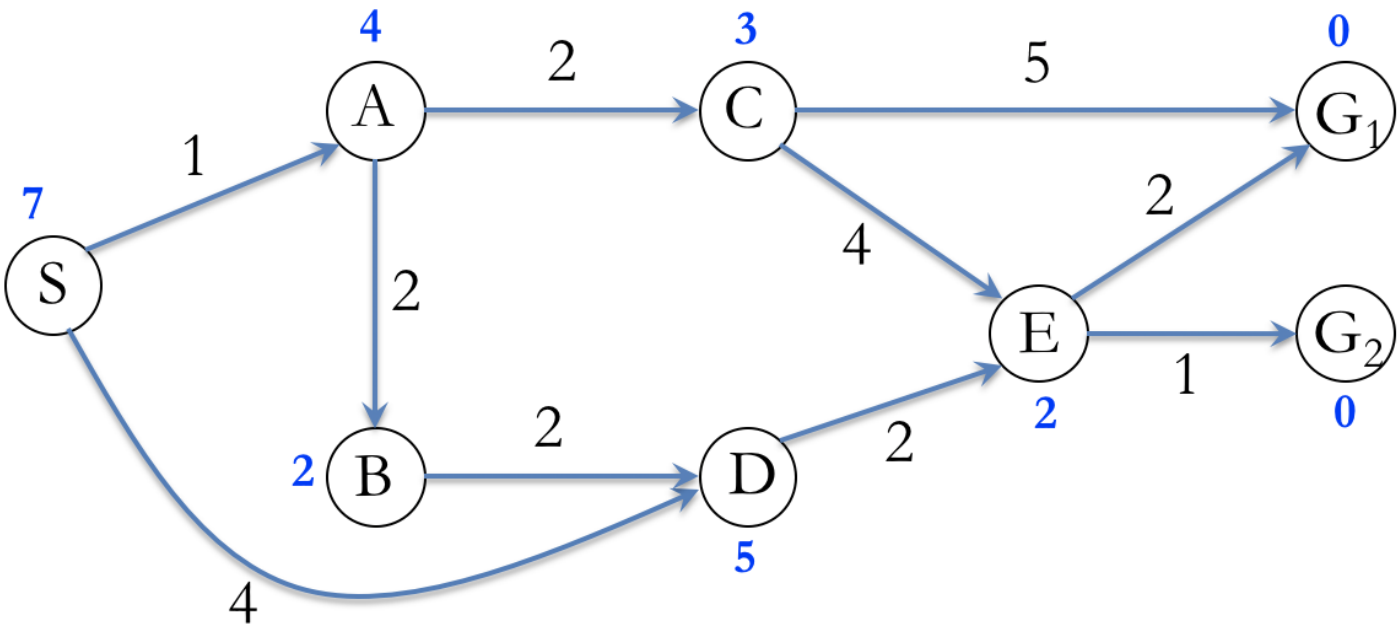


Give the 4th vertex visited.

- S
- A
- B
- C
- D
- E
- G_1
- G_2

Question 2 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***heuristic depth first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited. In this question, assume that "reached" is NOT used.***

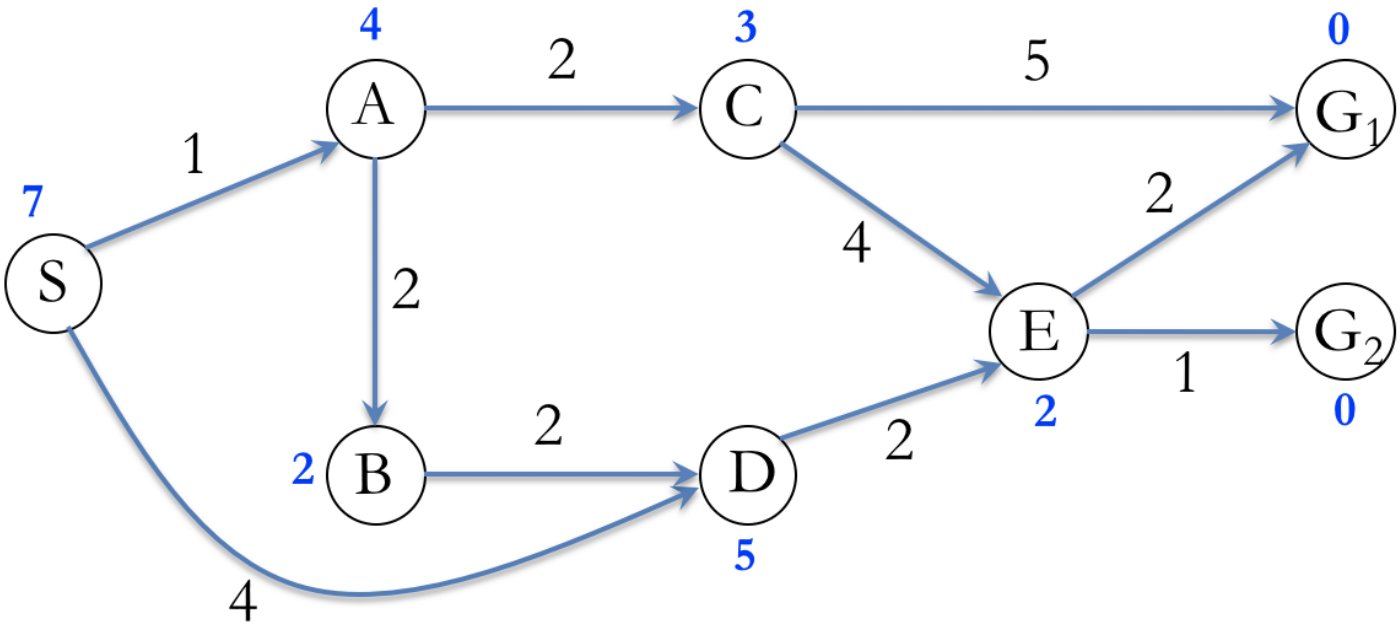


What is the evaluation score of the 4th vertex visited?

- ○ 0
- ○ 1
- ○ 2
- ○ 3
- ○ 4
- ○ 5
- ○ 6
- ○ 7
- ○ 8
- ○ 9
- ○ 10

Question 3 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***heuristic depth first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited. In this question, assume that "reached" is NOT used.***

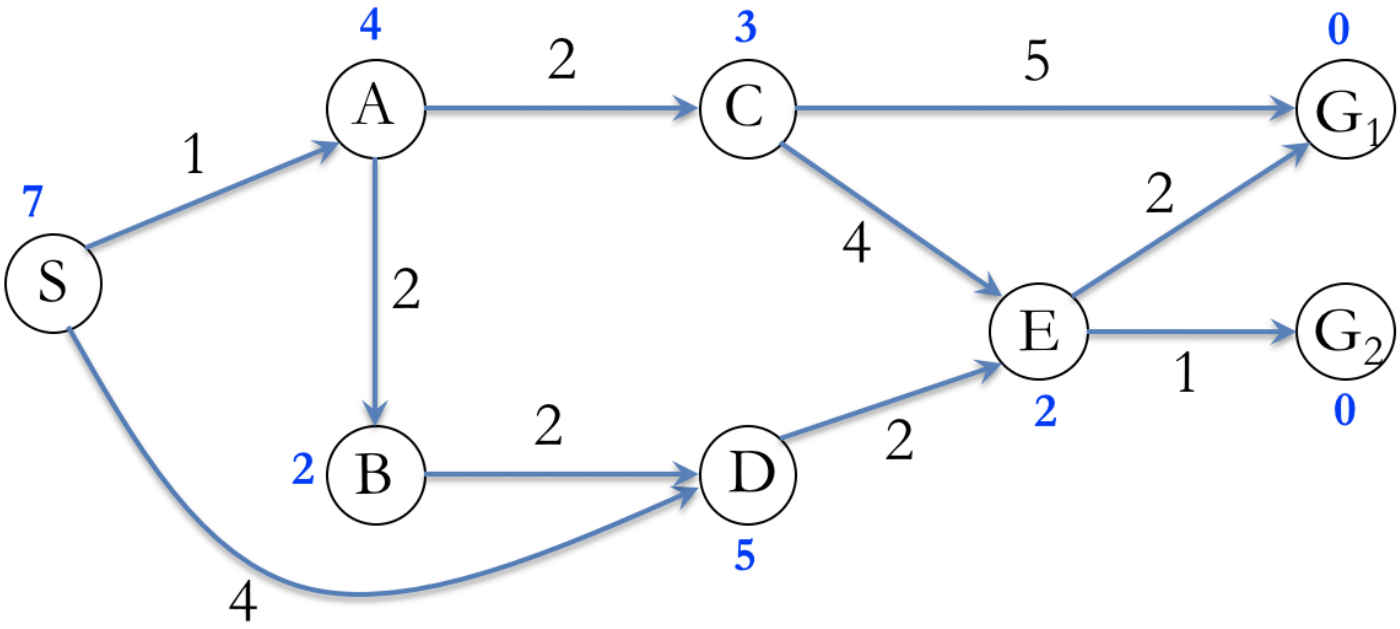


Give the 5th vertex visited.

- S
- A
- B
- C
- D
- E
- G_1
- G_2

Question 4 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***heuristic depth first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited. In this question, assume that "reached" is NOT used.***

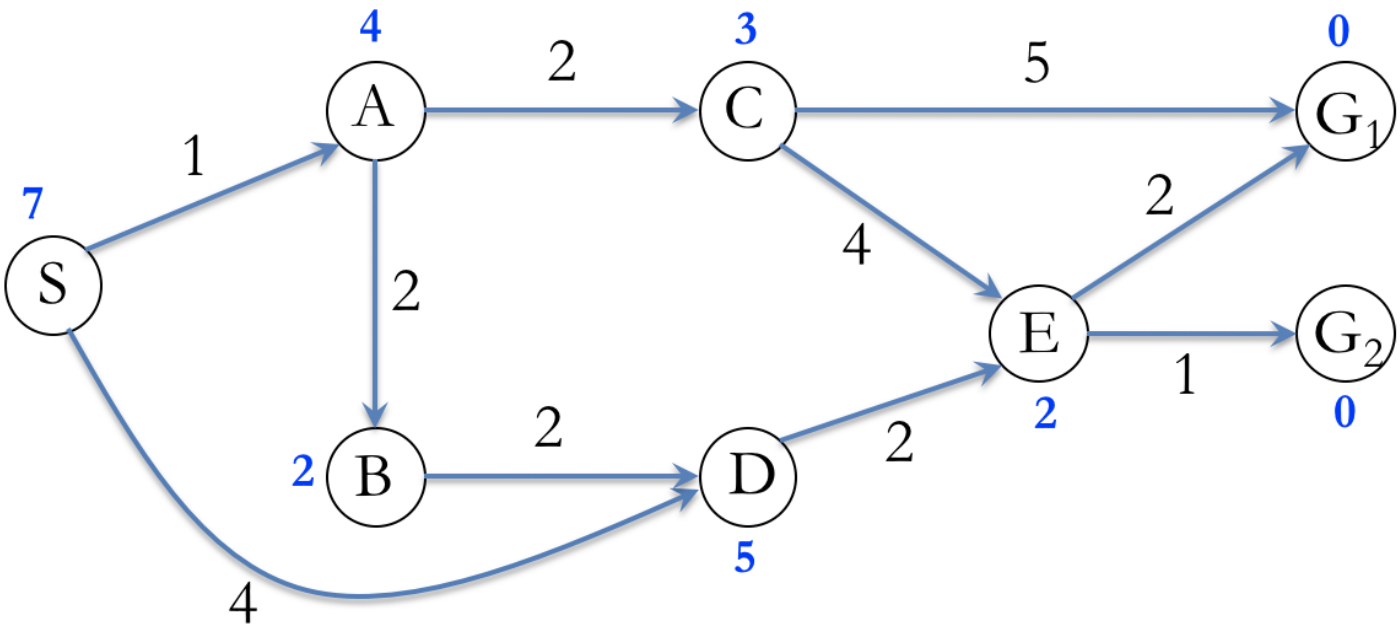


What is the evaluation score of the 5th vertex visited?

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

Question 5 **2 / 2 points**

You are asked to give the vertices that are visited (i.e., checked for goalness) by ***heuristic depth first search***. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited. In this question, assume that "reached" is NOT used.***

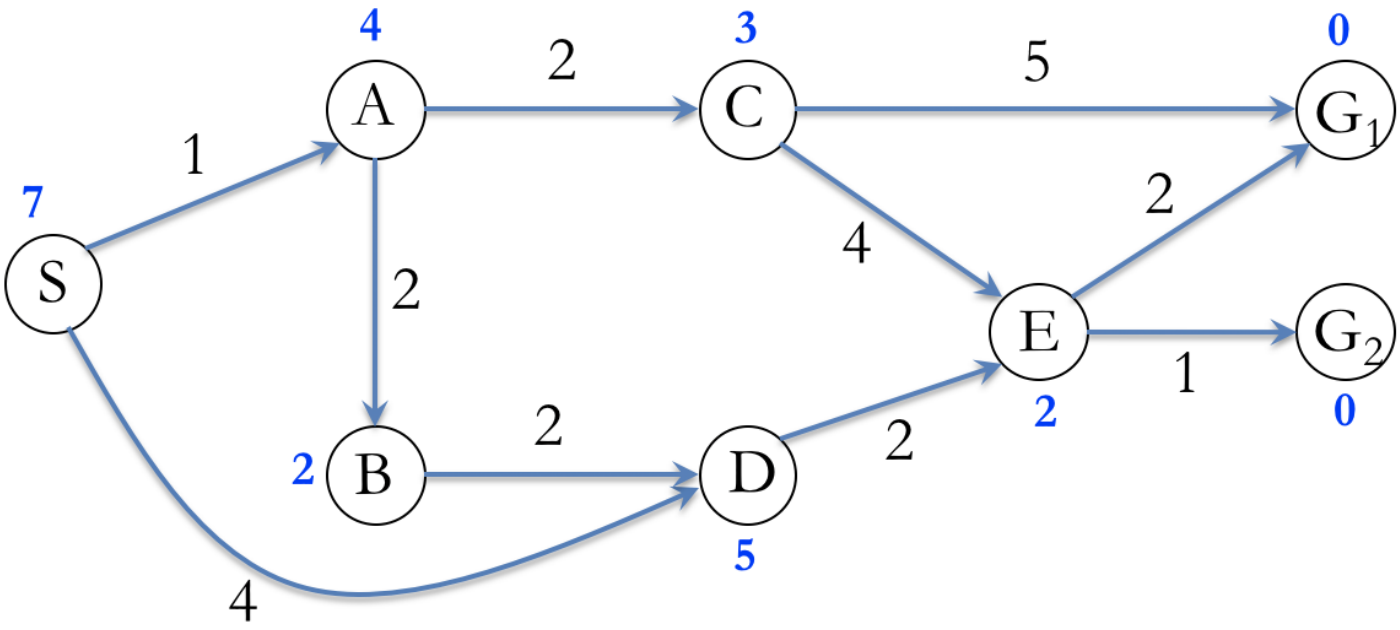


Give the goal, G_1 or G_2, that is found and returned by heuristic depth first search.

- G_1
- G_2

Question 6 **2 / 2 points**

You are asked to give vertices that are visited (i.e., checked for goalness) by ***heuristic depth first search***. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited. In this question, assume that "reached" is NOT used.***

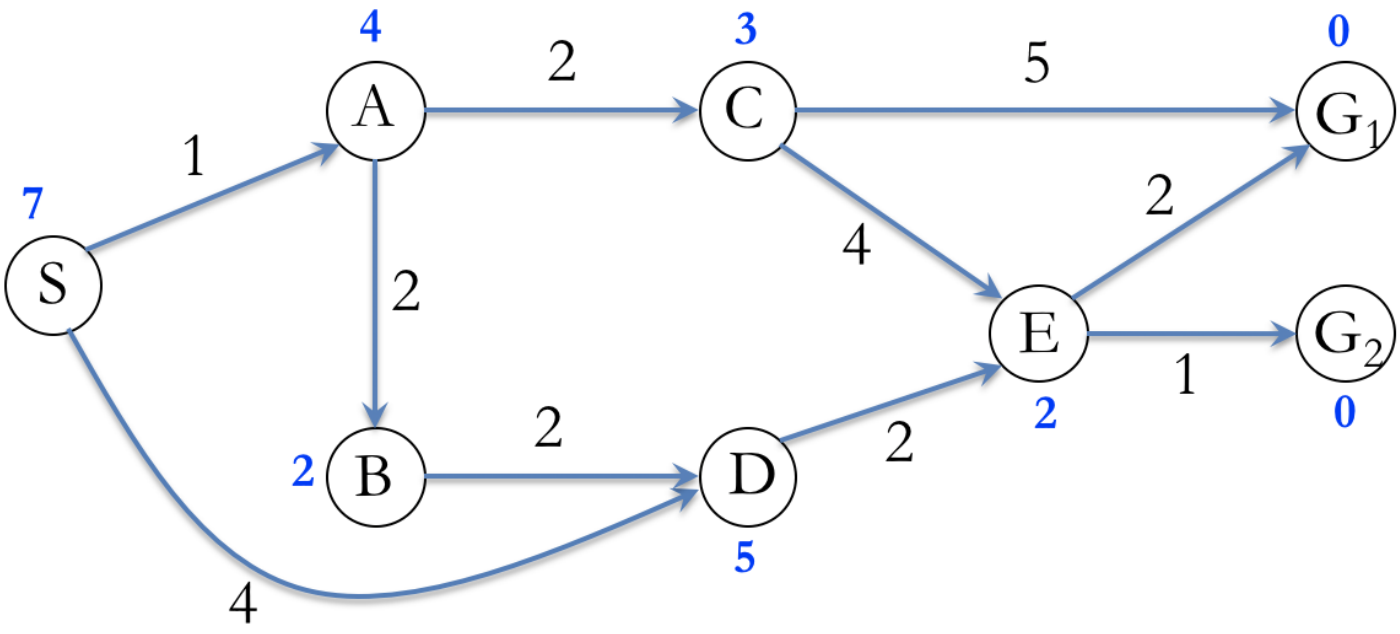


What is the total **path cost** to the goal that is found?

- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12

Question 7 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***greedy best-first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

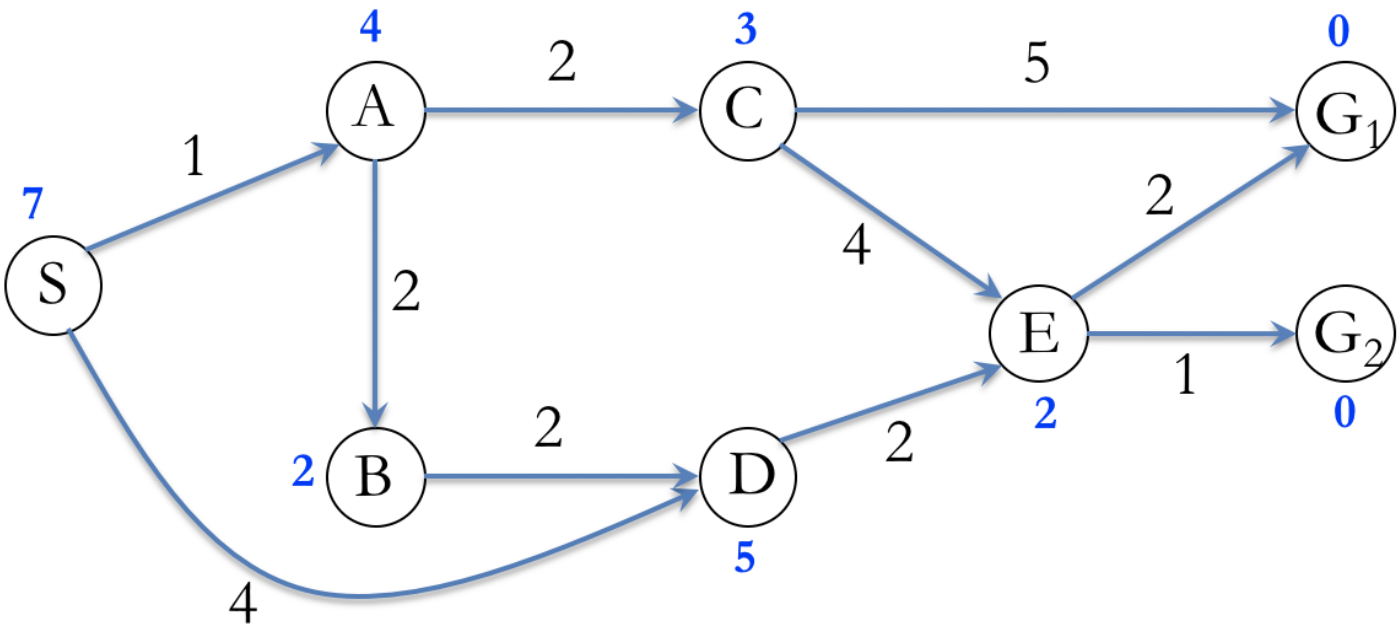


Give the 4th vertex visited.

- S
- A
- B
- C
- D
- E
- G_1
- G_2

Question 8 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***greedy best-first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

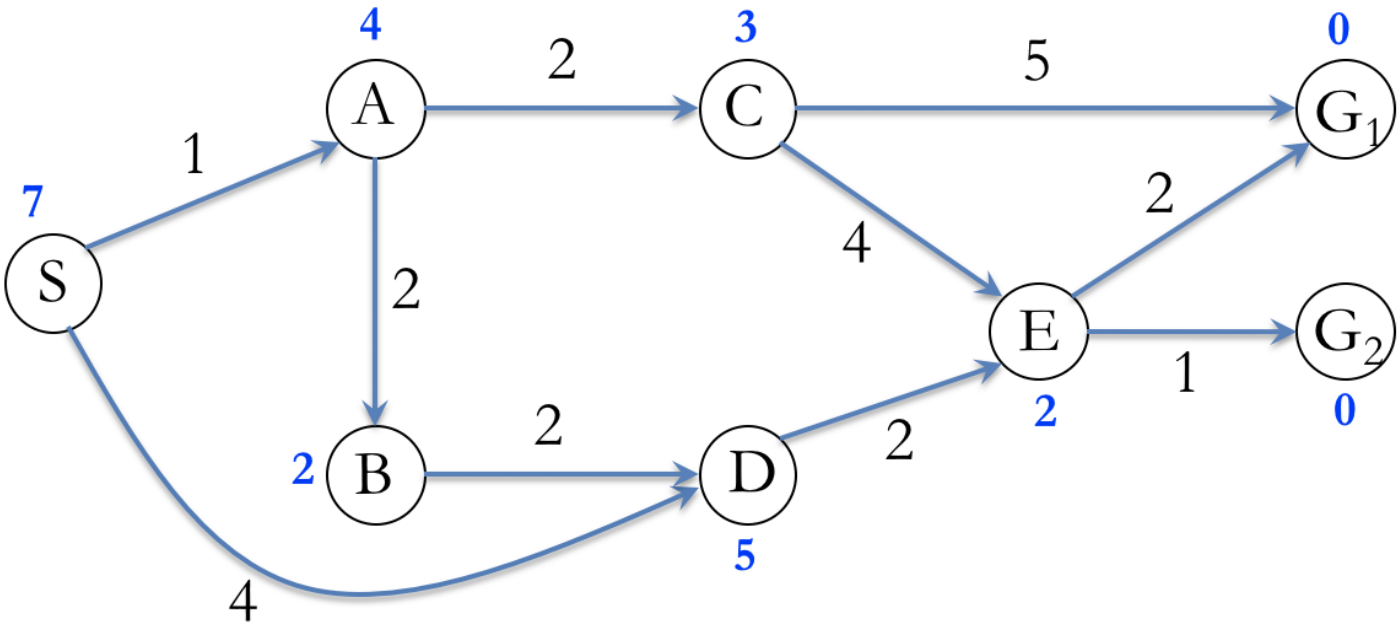


Give the evaluation score of the 4th vertex visited.

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

Question 9 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***greedy best-first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

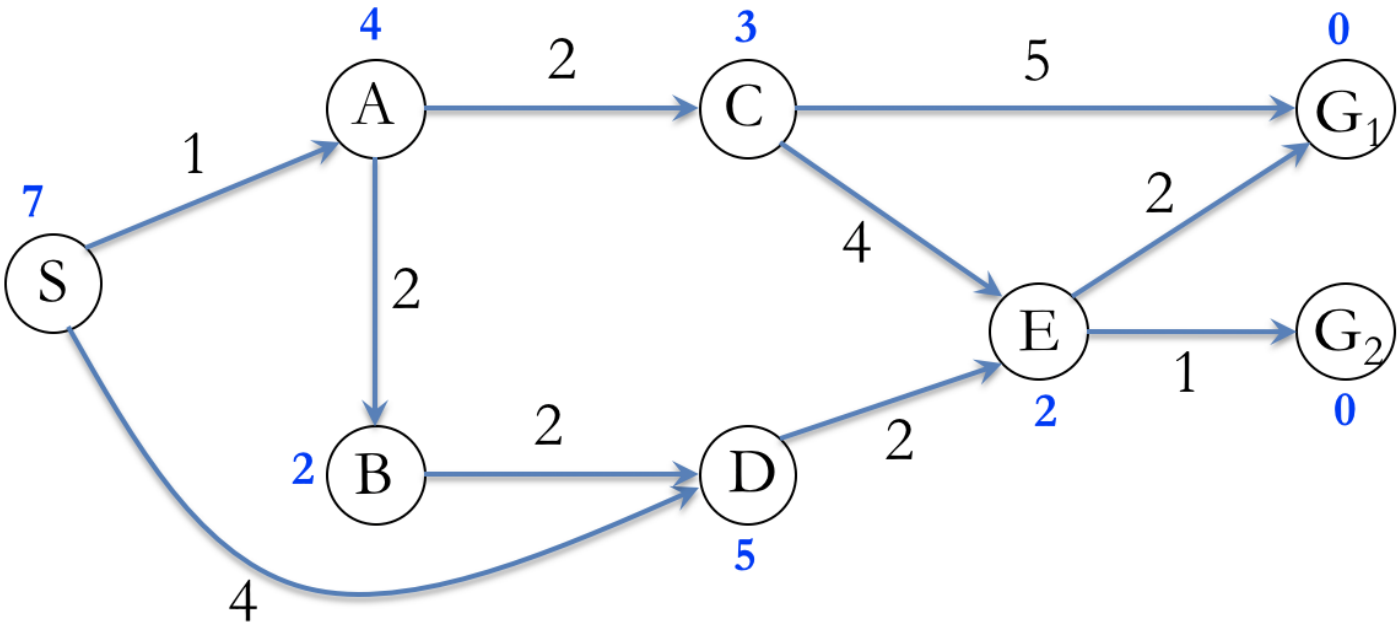


Give the 5th vertex visited.

- S
- A
- B
- C
- D
- E
- G_1
- G_2

Question 10 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***greedy best-first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

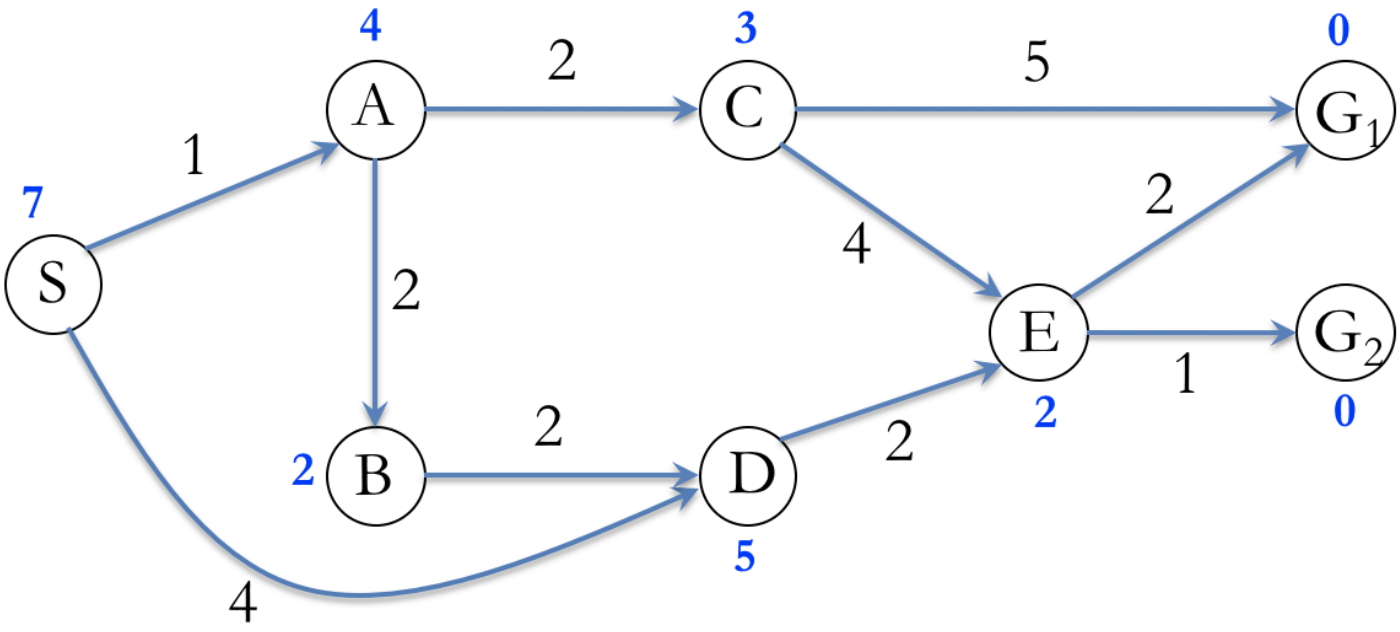


Give the evaluation score of the 5th vertex visited.

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

Question 11 **2 / 2 points**

You are asked to give vertices that are visited (i.e., checked for goalness) by ***greedy best-first search***. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

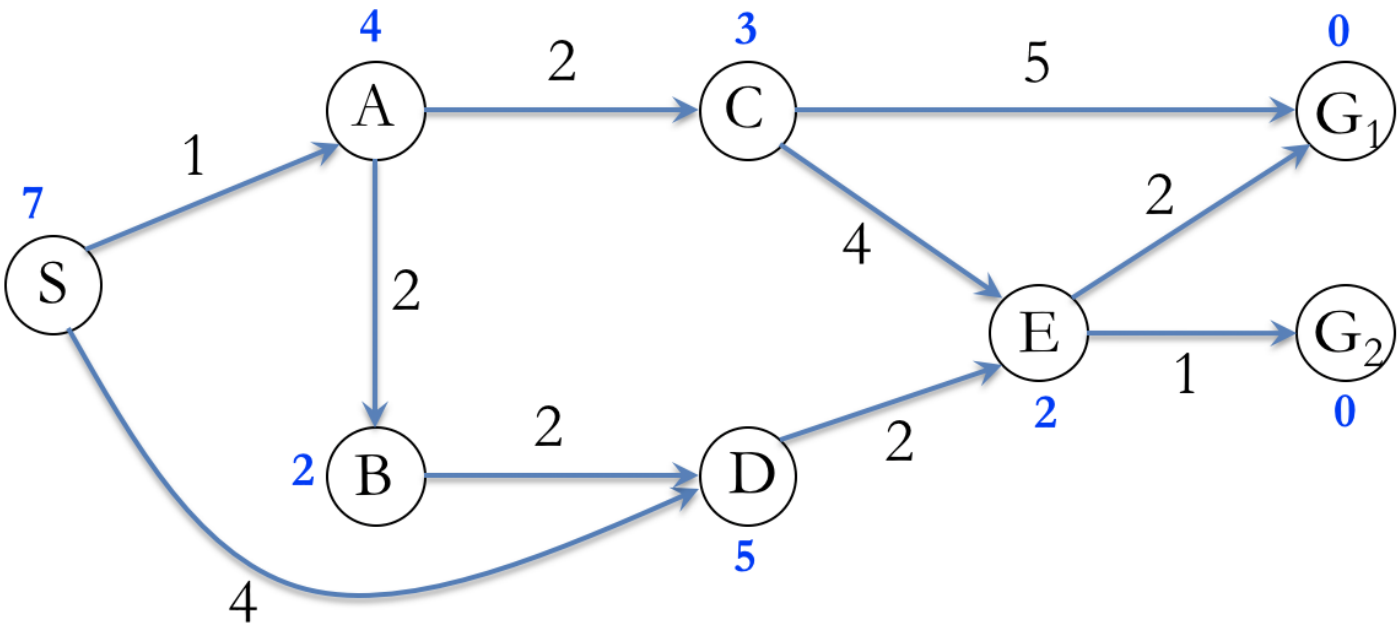


Give the goal, G1 or G2, that is found and returned by greedy best-first search.

- G_1
- G_2

Question 12 **2 / 2 points**

You are asked to give vertices that are visited (i.e., checked for goalness) by ***greedy best-first search***. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***



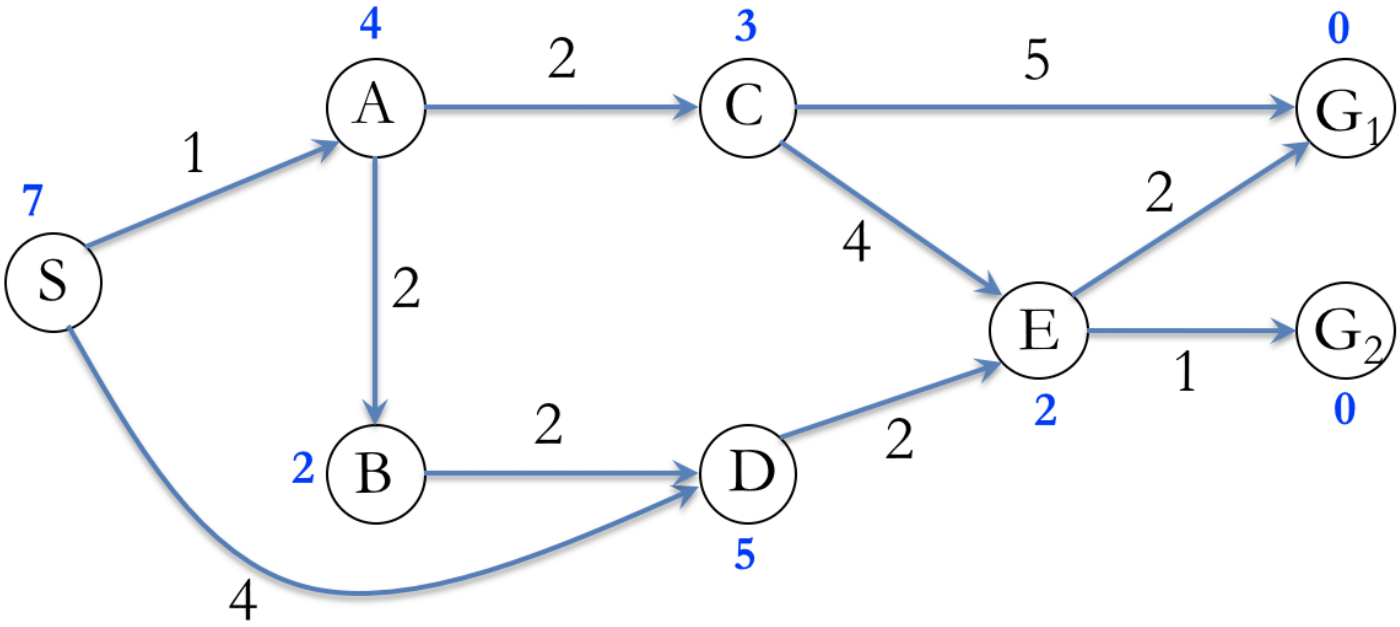
What is the total **path cost** to the goal that is found?

- 5
- 6
- 7
- 8
- 9
- 10

Question 13 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***lowest-cost-first search*** (aka uniform-cost search, Dijkstra's algorithm). You are also asked to give the evaluation score

(g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***



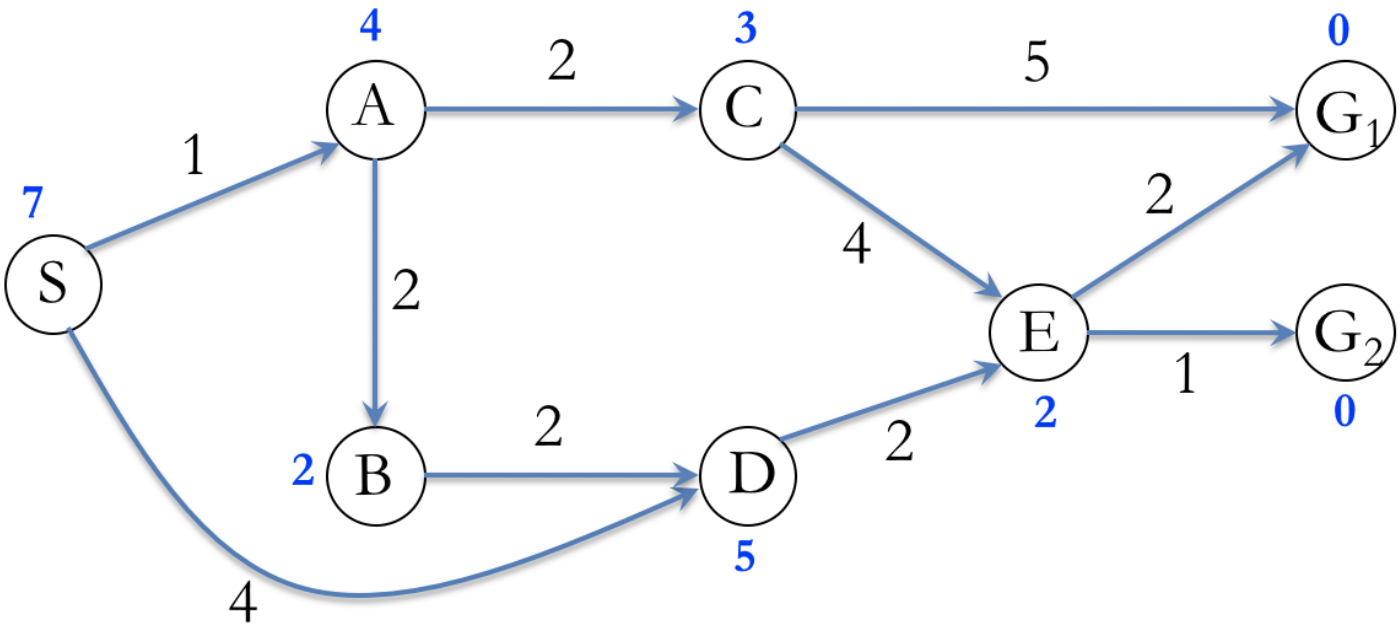
Give the 4th vertex visited.

- S
- A
- B
- C
- D
- E
- G_1
- G_2

Question 14 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***lowest-cost-first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled

in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***



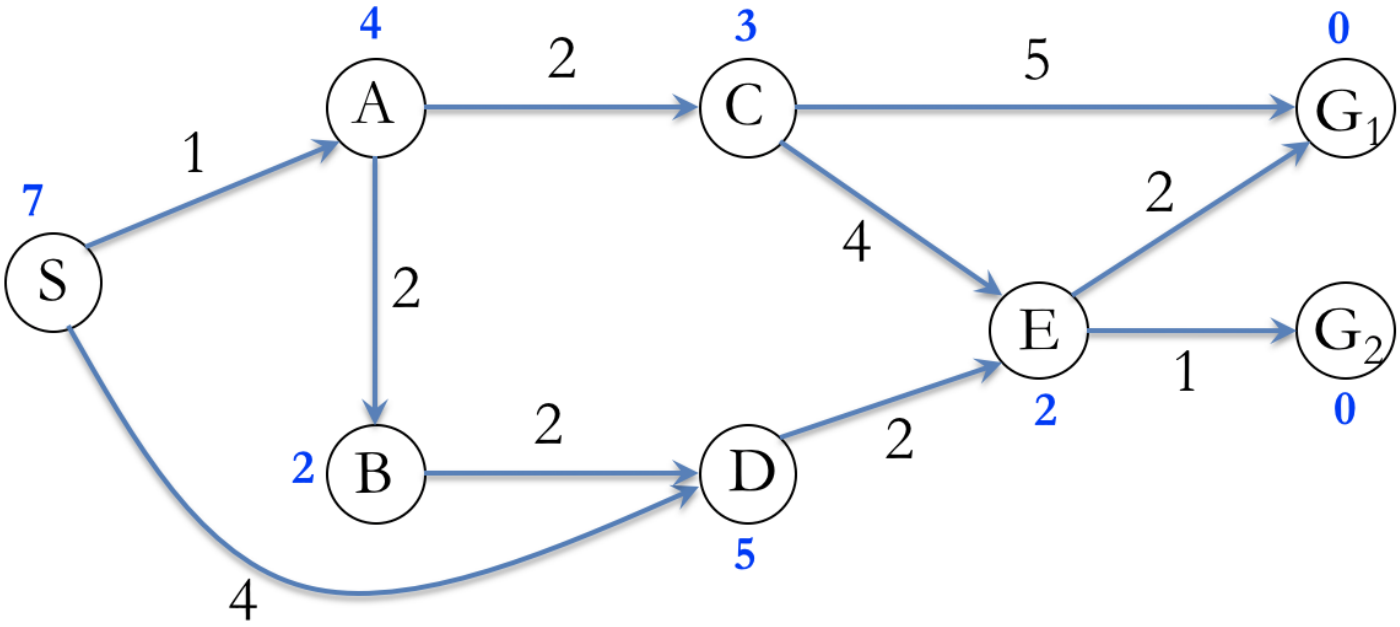
Give the evaluation score of the 4th vertex visited.

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

Question 15 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***lowest-cost-first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated

by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***



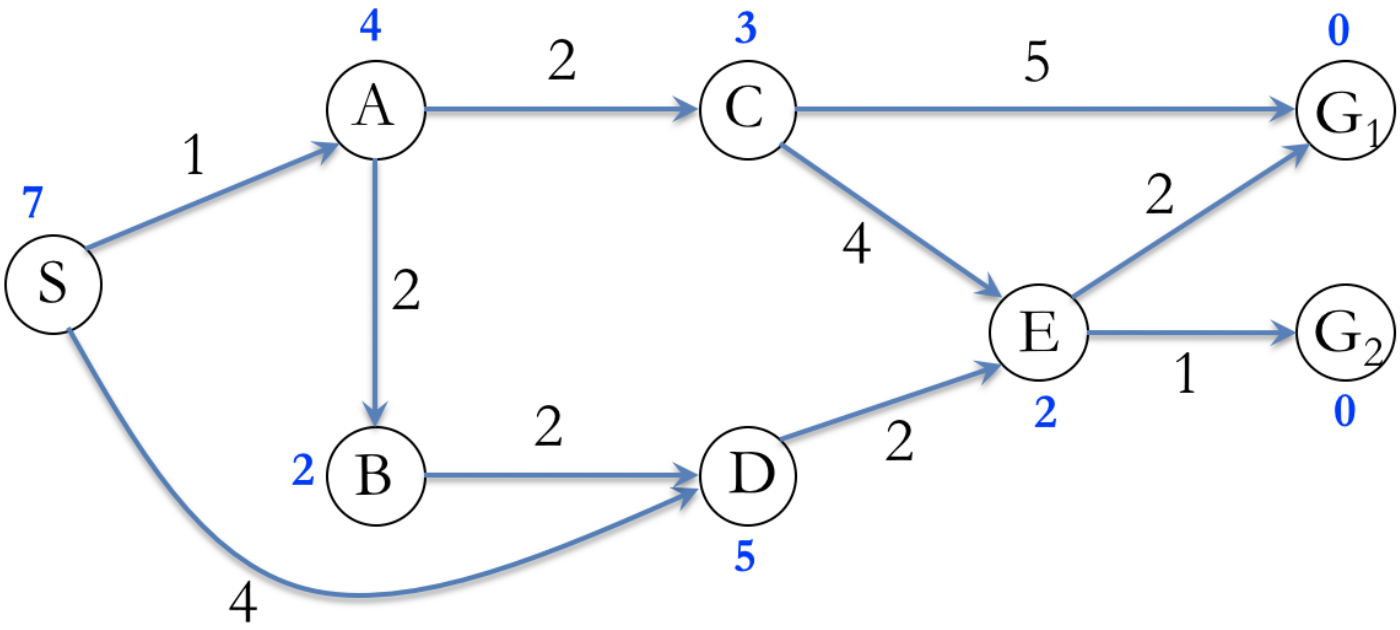
Give the 5th vertex visited.

- S
- A
- B
- C
- D
- E
- G_1
- G_2

Question 16 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***lowest-cost-first search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled

in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

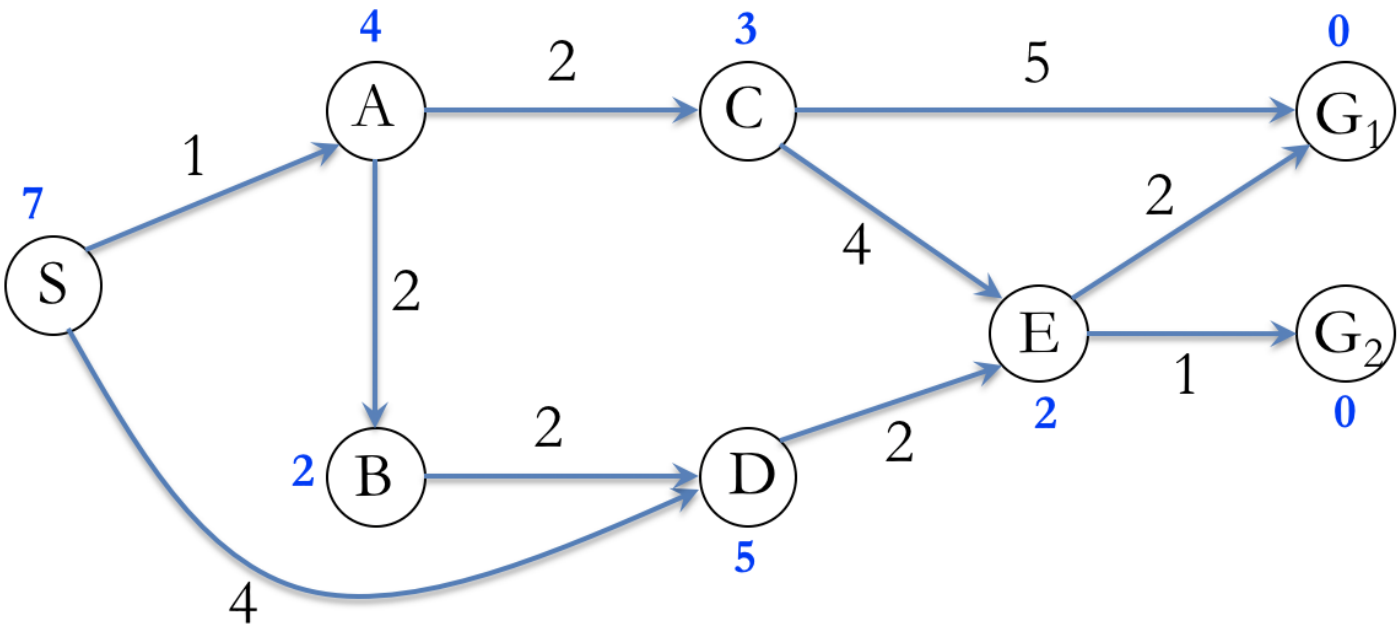


Give the evaluation score of the 5th vertex visited.

- 4
- 5
- 6
- 7
- 8
- 9
- 10

Question 17 **2 / 2 points**

You are asked to give vertices that are visited (i.e., checked for goalness) by ***lowest-cost-first search***. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

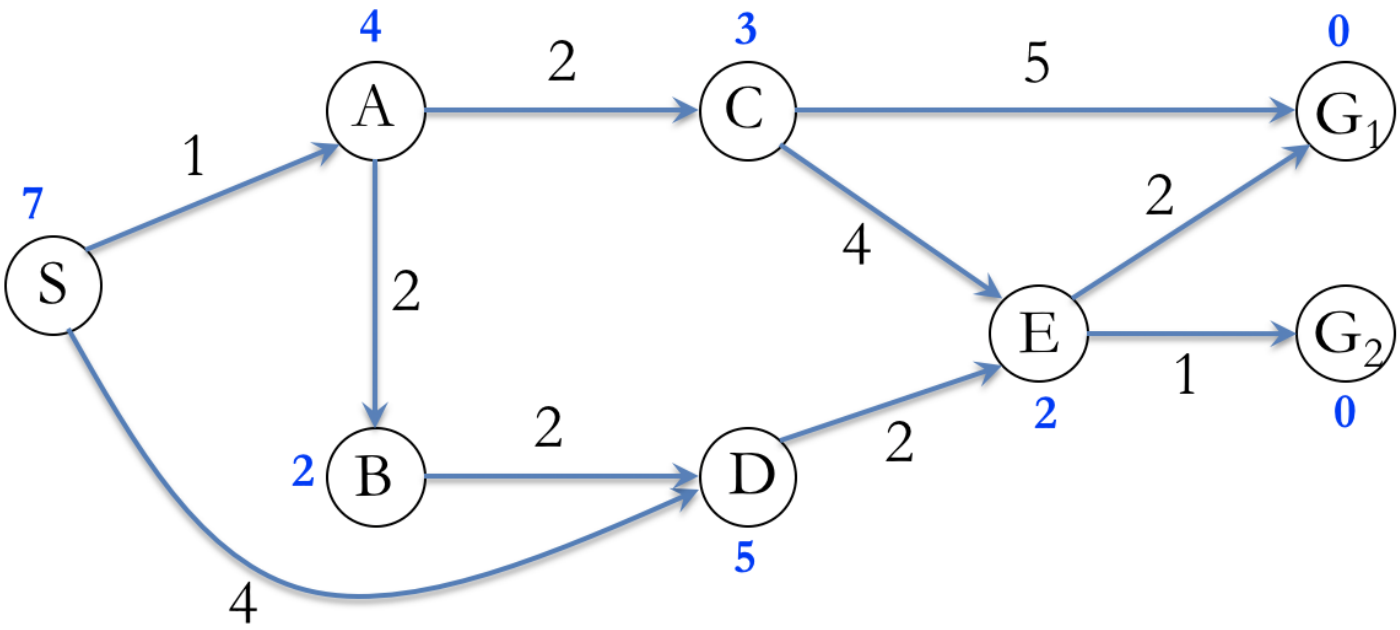


Give the goal, G_1 or G_2, that is found and returned by lowest-cost-first search.

- G_1
- G_2

Question 18 **2 / 2 points**

You are asked to give vertices that are visited (i.e., checked for goalness) by ***lowest-cost-first search***. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

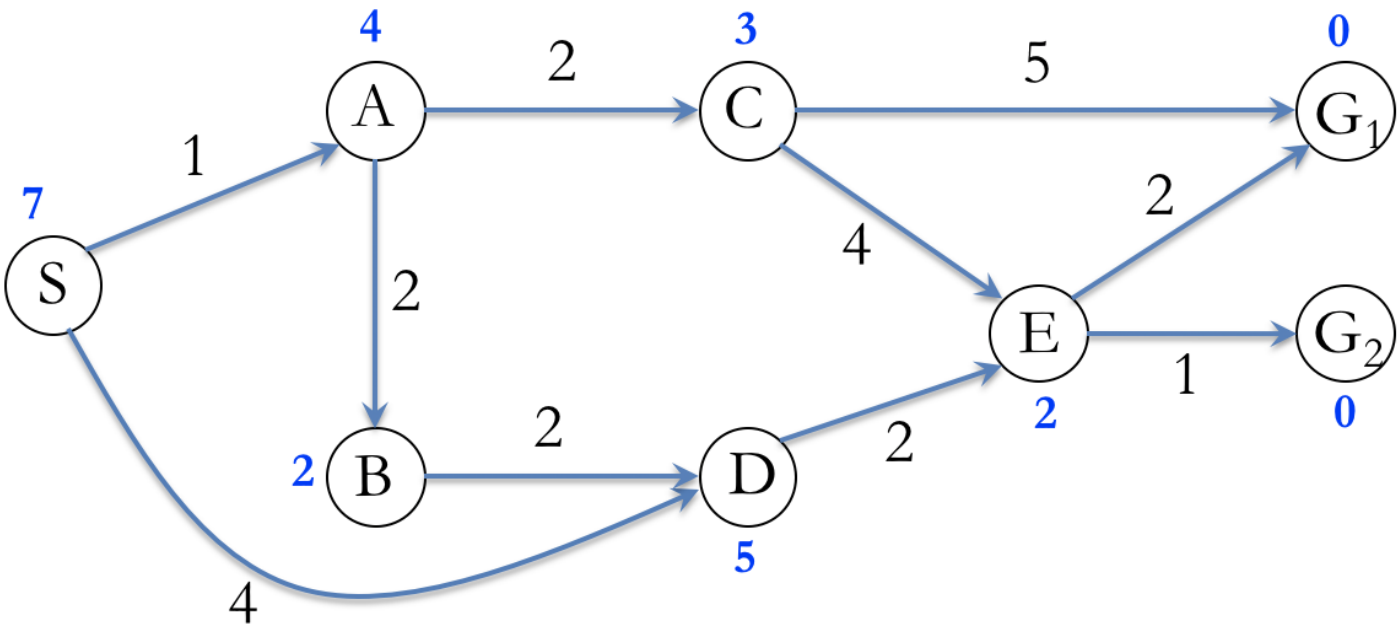


What is the total **path cost** to the goal that is found?

- 4
- 5
- 6
- 7
- 8
- 9
- 10

Question 19 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***A* search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

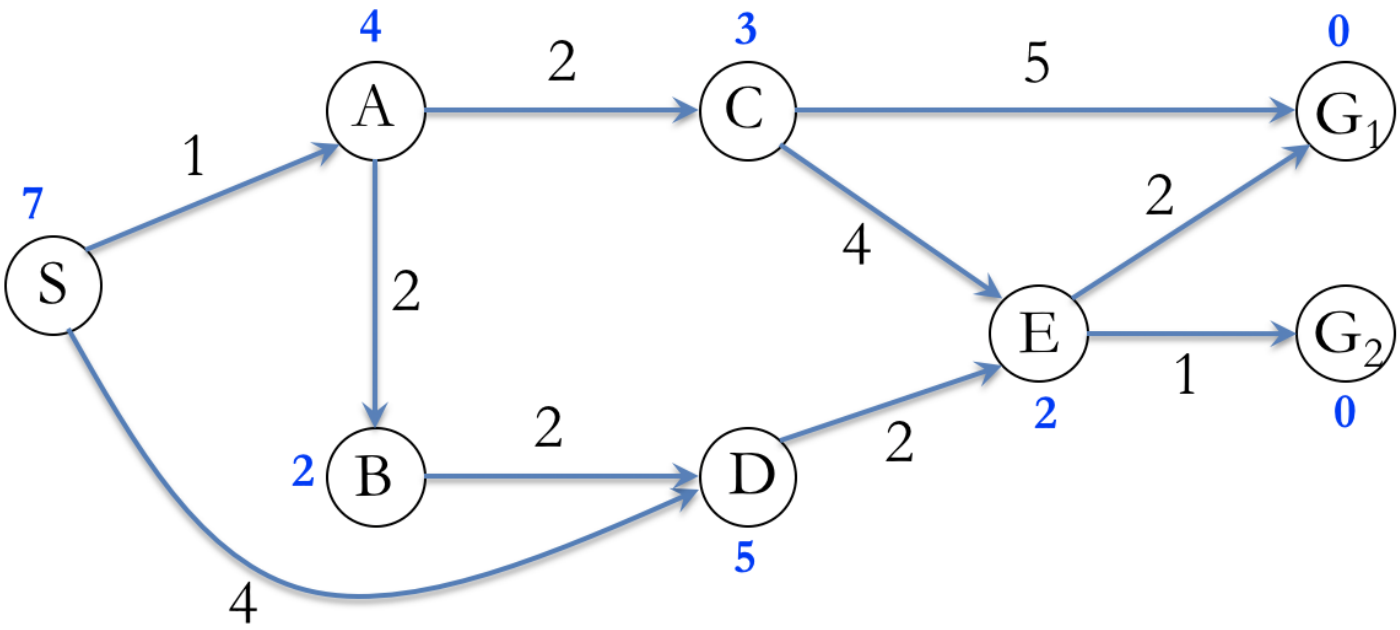


Give the 4th vertex visited.

- S
- A
- B
- C
- D
- E
- G_1
- G_2

Question 20 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***A* search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***



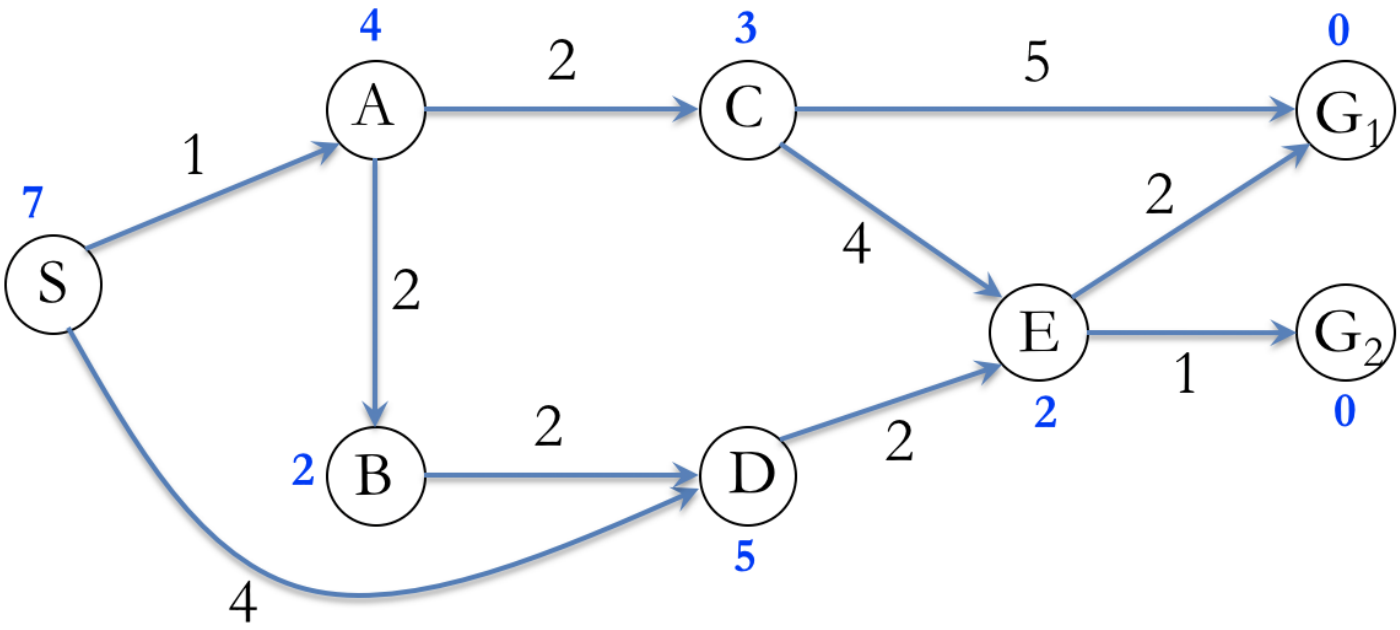
Give the evaluation score of the 4th vertex visited.

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

Question 21 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***A* search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on

the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

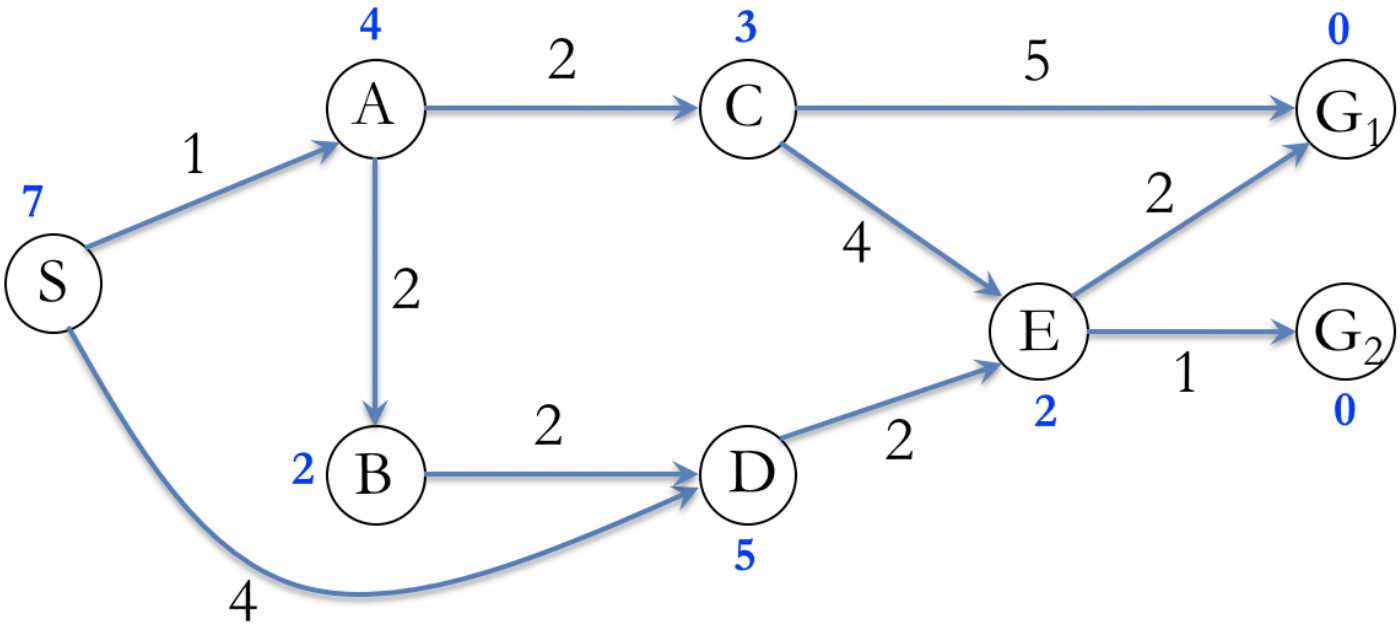


Give the 5th vertex visited.

- S
- A
- B
- C
- D
- E
- G_1
- G_2

Question 22 **2 / 2 points**

You are asked to give the Nth vertex that is visited (i.e., checked for goalness) by ***A* search***. You are also asked to give the evaluation score (g-score, h-score, or a combination as dictated by the search strategy) for the Nth vertex. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2. ***In all cases, S (the start vertex) is the 1st vertex visited.***

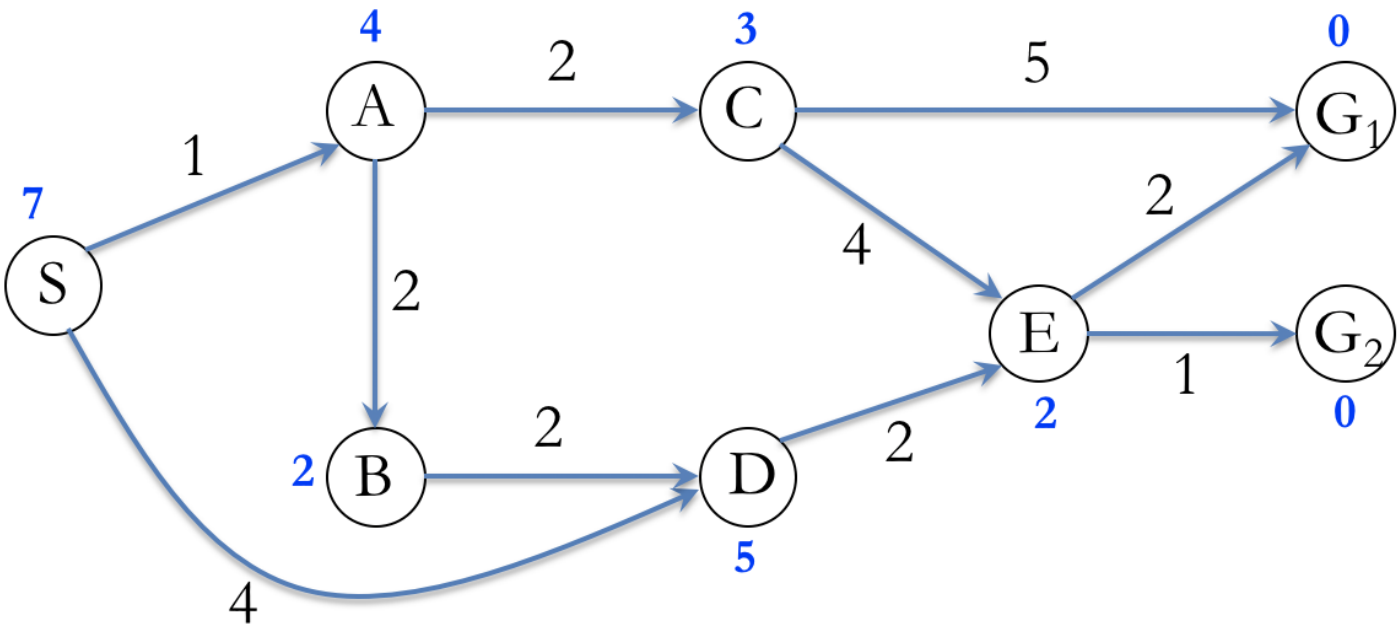


Give the evaluation score of the 5th vertex visited.

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

Question 23 **2 / 2 points**

You are asked to give vertices that are visited (i.e., checked for goalness) by ***A* search***. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2.

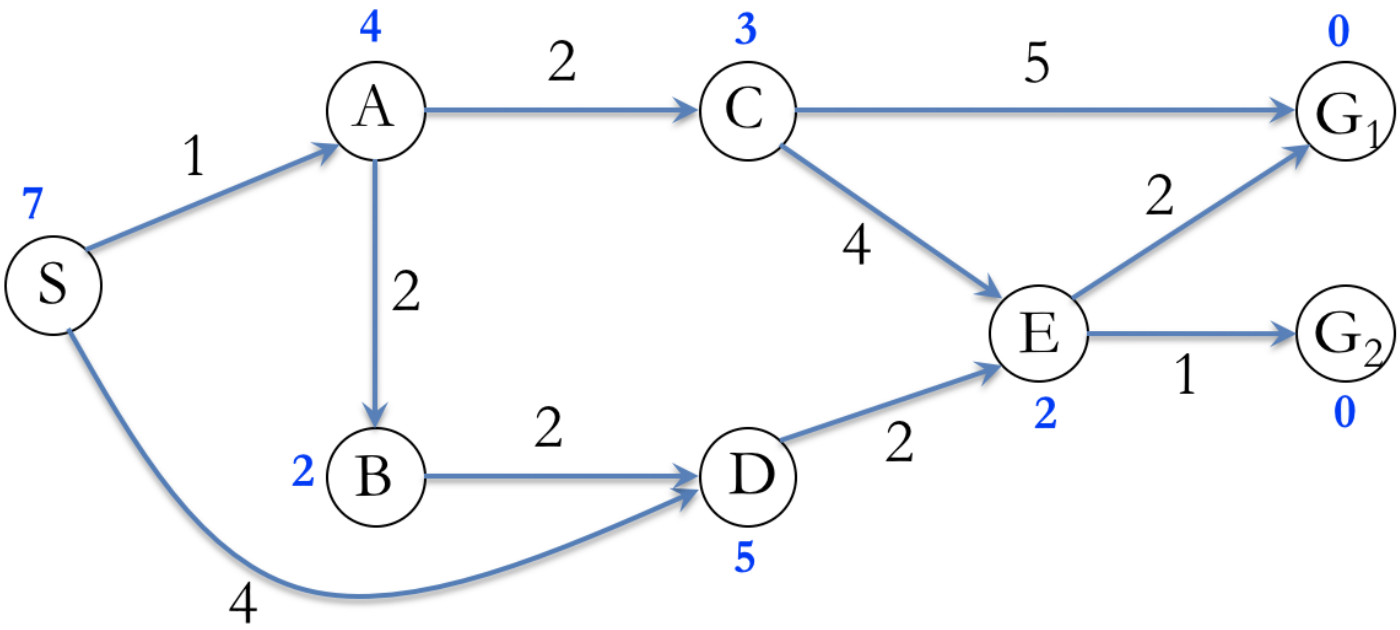


Give the goal, G_1 or G_2, that is found and returned by A*.

- G_1
- G_2

Question 24 **2 / 2 points**

You are asked to give vertices that are visited (i.e., checked for goalness) by ***A* search***. In the case of two or more vertices with the same evaluation score on the frontier, break the tie by visiting the vertices in alphabetical order as labeled in the example graph. G_1 is alphabetically before G_2.

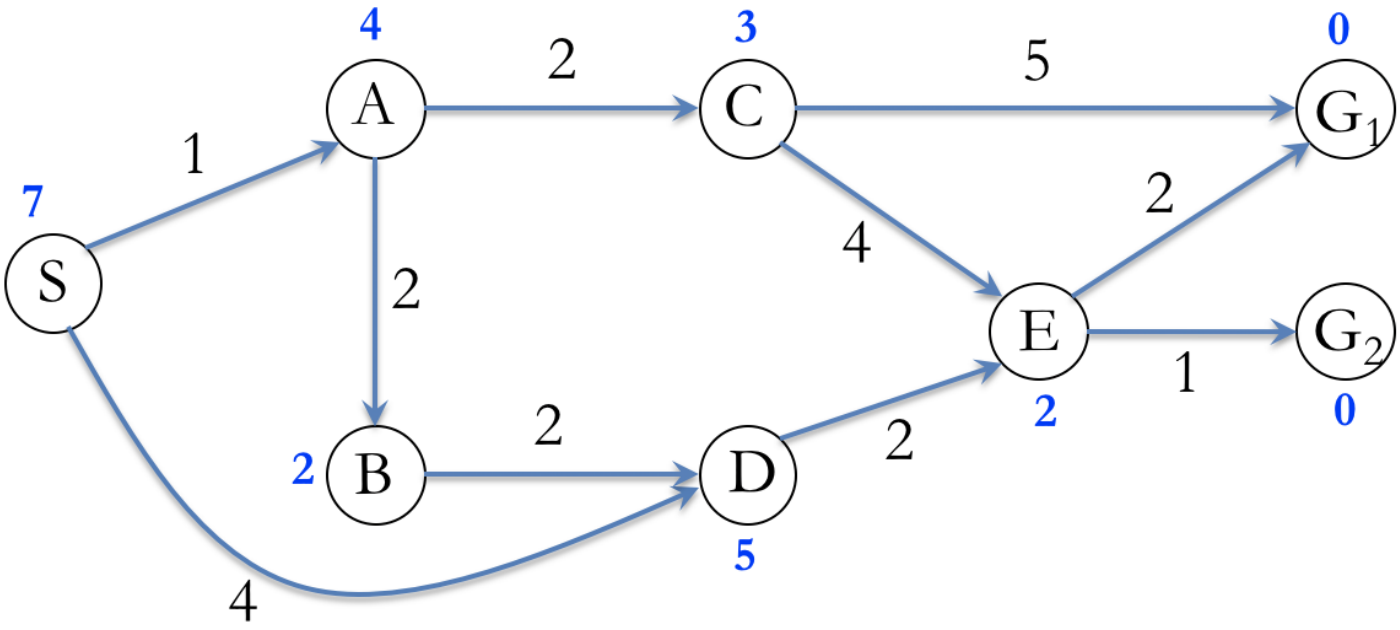


What is the total path cost to the goal that is found?

- 4
- 5
- 6
- 7
- 8
- 9
- 10

Question 25 **3 / 3 points**

Which vertices would be checked for goalness on the first iteration of ***iterative deepening A****? Assume that S is visited on the first iteration, **but others are as well.** S's f-cost determines the initial cutoff for the first iteration. Select all that apply.



- [ ] A
- [ ] B
- [ ] C
- [ ] D
- [ ] E
- [ ] G_1
- [ ] G_2

Question 26 **2 / 2 points**

Anytime search continues to search for solutions after finding the first solution.

- ( ) True
- ( ) False

Question 27 **2 / 2 points**

The most challenging task environments for AI are fully observable, single agent, deterministic, episodic, and discrete.

- ( ) True
- ( ) False

Question 28 **2 / 2 points**

The ***runtime cost*** of a depth-bounded depth-first search is O(B*D), where B is the branching factor and D is the depth bound.

- ( ) True
- ( ) False

Question 29 **2 / 2 points**

Increased heuristic accuracy effects search costs by reducing the effective branching factor of search.

- ( ) True
- ( ) False

Question 30 **2 / 2 points**

The path between a start state and a descendant state M can be recovered through M's SearchNode parent link, and subsequent ancestor links -- this mitigates redundancy in path storage.

- True
- False

Question 31 **2 / 2 points**

Nondeterministic algorithms can be slow due to search, but they can be elegant and simply stated too, and machine learning can speed them up.

- True
- False

Question 32 **2 / 2 points**

In contrast to offline search, online search interleaves computation and action.

- True
- False

Attempt Score: 100 %

Overall Grade (last attempt): 77 %

Done