Home > Dijkstra: trace 1

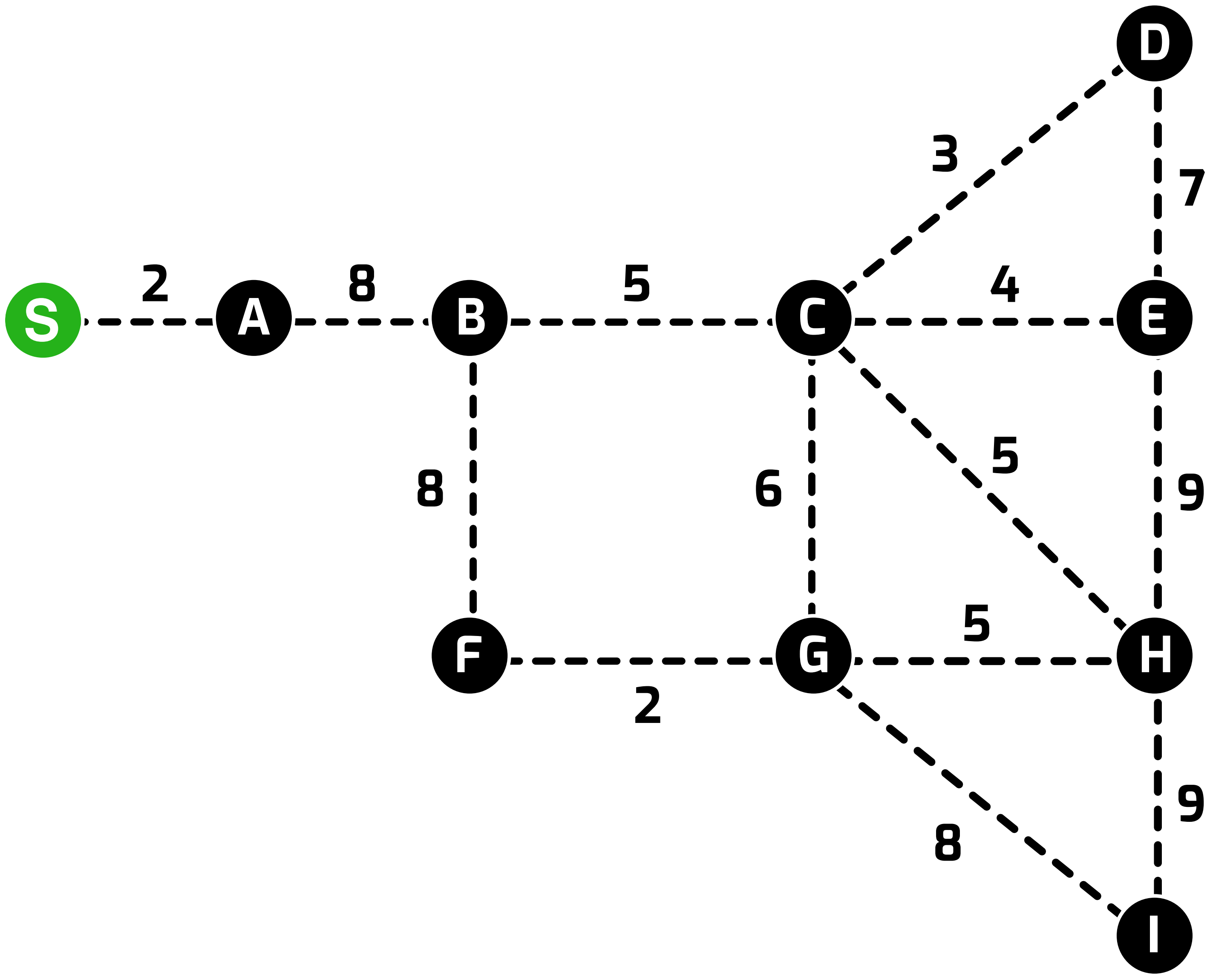
Dijkstra: trace 1

A Level Advanced

P P P P

This graph represents the network on a university campus. The nodes are computers and the weights show how long it takes for a signal to get from one computer to the other (in milliseconds).

Using Dijkstra's algorithm, you can find the quickest path to all of the nodes on the network from your starting node (highlighted in green).



Graph representing a university campus

The structured English that follows defines the main steps of the algorithm.

```
While the unvisited list is not empty:
   Set current node to the node with the lowest cost from the unvisited list
   Copy cost and previous values for current node from the unvisited list to t
   Remove the current node from the unvisited list

   For each neighbour of current node:
      If neighbour is not in the visited list
          Calculate new cost = weight of edge + cost of current node
```

```
            If new cost is less than neighbour's cost in unvisited list
                Update the neighbour's cost to become the new cost
                Update the neighbour's previous node to become the current node
```

Using pen and paper, follow the algorithm for the graph shown and then answer the following questions:

Part A

What is the shortest path to G? Write the order of the nodes without any spaces between them, e.g. SABCD

Part B

What is the cost of the shortest path to G?

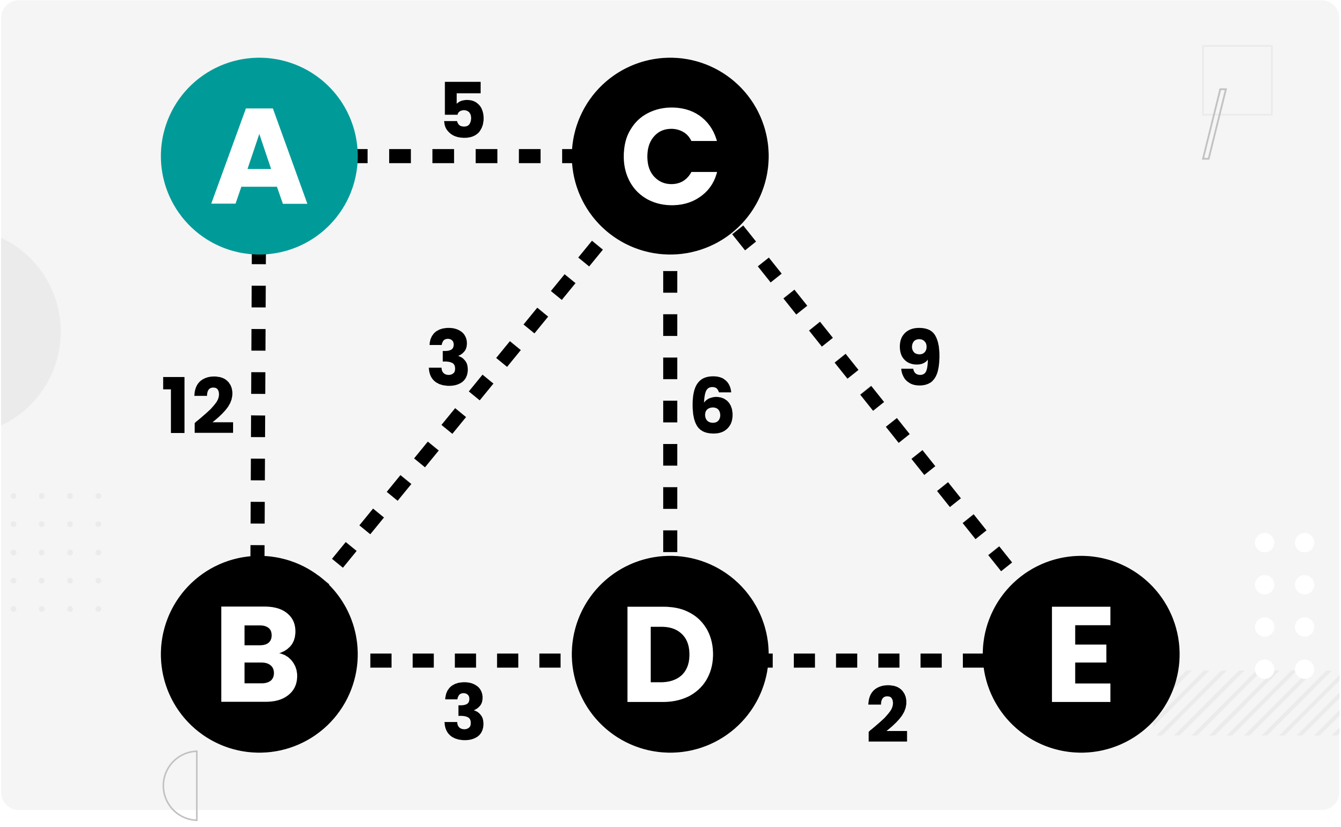
Dijkstra: trace 2

A Level Advanced

P P P P

Ali is planning to visit his friend Eli. The graph shown below illustrates the connections between Ali's house (node A) and that of Eli (node E).

Ali will use Dijkstra's algorithm to find the shortest path between the two houses. The algorithm selects the next current node by picking the node from the unvisited list with the **lowest cost**.



Ali's map.

Part A

Which of the following options shows the final visited list that is produced by running the algorithm?

○

Visited list		
Node	**Cost (from start)**	**Previous**
A	0	none
C	5	A
B	8	C
D	11	B
E	13	D

○

Visited list		
Node	**Cost (from start)**	**Previous**
A	0	none
B	12	A
C	5	A
D	11	C
E	13	D

○

Visited list		
Node	**Cost (from start)**	**Previous**
A	0	none
B	8	C
C	5	A
D	11	C
E	13	D

○

Visited list		
Node	**Cost (from start)**	**Previous**
A	0	none
C	5	A
B	8	C
D	11	C
E	13	D

Part B

Using the visited list that is produced by following the algorithm, which option correctly shows the **route** of the shortest path from A to E?

○ A - C - B - D - E

○ A - C - E

○ A - B - D - E

○ A - C - D - E

Dijkstra: trace 4

A Level Advanced
P P P P

The big sales start tomorrow morning at the shops, and Queenie wants to get there as quickly as possible to bag a bargain.

The route data is stored in an adjacency matrix. The nodes are road junctions and the weights represent the time in minutes between junctions. Queenie's home is node Q and the town centre is T.

	A	L	Q	R	S	T	V
A		4	3	8			2
L	4			3		5	
Q	3			6			
R	8	3	6			4	
S						2	2
T		5		4	2		
V	2				2		

Part A

Using Dijkstra's algorithm, find the length of the quickest route ("shortest path") from her home (Q) to the town centre (T). Type the answer as a number.

Part B

What is the route? Type the letters without spaces.

Home > Dijkstra: trace 3

Dijkstra: trace 3

A Level Advanced
P P P P

The data for a graph is stored in an **adjacency list** as shown below.

Node	Neighbours (node, weight)
A	E,1 ; H,4
B	D,4 ; F,2 ; H,2
C	G,1 ; H,1
D	B,4
E	A,1 ; G,2
F	B,2 ; G,2
G	C,1 ; E,2 ; F,2
H	A,4 ; B,2 ; C,1

Dijkstra's algorithm will be run to fnd the shortest path between the start node **E** and all other nodes. The algorithm selects the next current node by picking the node from the unvisited list with the **lowest cost**.

Part A

What is the cost of the shortest path from E to D? Enter your answer as an integer value.

Part B

What is the route of the shortest path from **E** to **B**? Select your answer from the options shown below.

○ E - A - H - B

○ E - G - F - B

○ E - G - C - H - B

○ E - G - H - B

A*: heuristics

A Level P P Advanced P P

The A* path finding algorithm improves on the efficiency of Dijkstra's algorithm by introducing a heuristic element.

Which of the following statements correctly describes the heuristic function?

○ The heuristic function provides an estimate of the cost of the path between an interim node in the graph and the target node, and must not over-estimate the costs.

○ The heuristic function provides an estimate of the cost of the path between an interim node in the graph and the target node, and must not under-estimate the costs.

○ The heuristic function provides an estimate of the cost of the path between the start node and an interim node in the graph, and must not over-estimate the costs.

○ The heuristic function provides an estimate of the cost of the path between the start node and an interim node in the graph, and must not under-estimate the costs.

A*: trace 2

A Level Advanced

C C C C

A computer game uses the A* path finding algorithm to determine the shortest path between the enemy and its target. All of the relevant game components are stored as nodes in a graph and the weights represent the distances between them, which are kept up to date as the game play progresses.

The path finding algorithm maintains a list of nodes yet to be visited; this is called the unvisited list. At a particular point in the game, the state of the list is as shown in the table below.

Unvisited list

node	g-score	f-score	previous
C	75	80	B
H	30	140	E
L	90	95	B
N	60	75	K
P	35	85	K
R	85	90	J

Which will it choose as the next node to visit?

○ P

○ H

○ C

○ N

A*: trace 1

A Level Advanced

C C C C

A delivery company makes uses of the A* algorithm to determine the fastest route for a delivery van to reach a depot.

- Depots are stored as nodes in a graph.
- Road junctions are also stored as nodes and the weights represent the cost of fuel to drive from node to node.
- The fuel consumption and cost is updated frequently by information collected from each van's navigation system.

The company has 50 depots and hundreds of vans; the graph data below shows a small subset of the graph data in the form of an adjacency list.

Node	Neighbours (node, weight)		Heuristic
A	E,20 ; H,40		30
B	D,45 ; F,20 ; H,60		35
C	G,105 ; H,150		40
D	B,45; G,10		10
E	A,20 ; G,70		50
F	B,20 ; G,60		45
G	C,105 ; D,10; E,70 ; F,60		0
H	A,40 ; B,60 ; C,150		100

A van needs to collect an item from **Depot H** and drop it at **Depot G**. The value returned by the heuristic function provides an estimate of the cost. The target node (G) is also shown in the table.

Part A

What is the least amount of fuel needed for the driver to complete the journey based on the current information?

Enter your answer as a single integer value. It might be helpful to draw the graph rather than work directly from the adjacency list.

Part B

What is the route of the shortest path?. Type your answer as a sequence of letters separated by commas with no spaces (e.g. A,B,C)