Stack: pop algorithm

Practice 2

The following statements, in structured English, form the algorithm for the **pop** operation on a stack. Put them into the correct order. You do not need to indent any of the statements.

Available items

- If stack empty, display error and return / halt
- Check if the stack is empty
- Decrement stack_pointer by one
- Else return the data item indexed by stack_pointer

Queue reversal

Challenge 2

A local highways agency is developing a program to simulate a new section of road. A queue will be used to store vehicles that are queuing at a junction. The new road will be quite narrow and if there is a blockage at the junction the vehicles will need to turn round one at a time starting at the end of the queue. This will create a new queue (heading in the opposite direction) that will be a reverse of the original queue. For example:

Original queue:

Car 1	Car 2	Van 1	Car 3
FRONT			REAR

Reversed queue:

Car 3	Van 1	Car 2	Car 1
FRONT			REAR

A stack can be used to reverse a queue and an algorithm has been developed to do this.

```
SIZE_OF_QUEUE = 4

PROCEDURE reverse(queue, stack)
   FOR i = 0 TO SIZE_OF_QUEUE - 1
      item  = [          ]
      [          ]
   NEXT i

   FOR i = 0 TO SIZE_OF_QUEUE - 1
      item = [          ]
      [          ]
   NEXT i
ENDPROCEDURE
```

Items:

dequeue() push(item) pop() enqueue(item)

Quiz:
STEM SMART Computer Science Week 47

Graph: representation 5

Practice 2

A graph can be represented using an adjacency matrix or an adjacency list. Under which circumstances would it be more appropriate to use an adjacency list (instead of an adjacency matrix)? Select **three** correct statements.

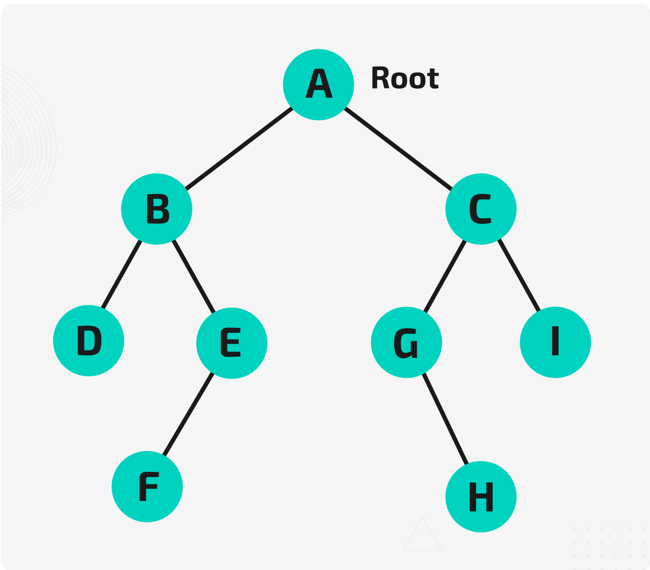
- [ ] When the presence or absence of edges will be tested frequently
- [ ] When edges will be removed frequently
- [ ] When nodes will be removed frequently
- [ ] When there are relatively few edges compared to the number of nodes in the graph and the amount of memory space is a potential issue
- [ ] When new nodes will be added frequently
- [ ] When new edges will be added frequently

Tree: rooted tree traversal 2

Challenge 2

Akram is designing a content management system (CMS) application. He is using a file system to organise the files (that hold the content) in folders so that each file can be retrieved easily. To do that, the folders are structured as a **rooted tree**.

A diagram of the tree is presented in the image below. For example, the folder assigned to node G is stored inside the folder assigned to node C which is stored in the root folder assigned to node A.



A tree data structure with nine nodes.

Drag and drop the given terms in the correct spaces to complete the following text.

Akram is exploring how the nodes of the rooted tree will be processed using the pre-order, post-order, and in-order tree traversals.

- In a pre-order traversal, each node is visited ______ either of the node's subtrees are visited
- In an in-order traversal, each node is visited ______ each of its subtrees
- In a post-order traversal, each node is visited ______ both of its subtrees are visited

In this example, the tree in the image above is traversed from left to right.

- With a pre-order traversal, the order that the nodes of the tree are visited is ______
- With an in-order traversal, the order that the nodes of the tree are visited is ______

- With a post-order traversal, the order that the nodes of the tree are visited is

Items:

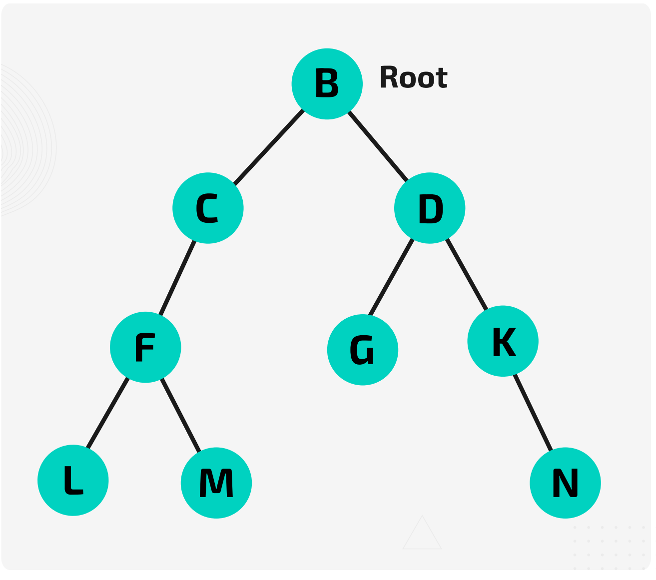
before | between | after | ABDEFCGHI | DBFEAGHCI | DFEBHGICA

Quiz:
[STEM SMART Computer Science Week 47]

Binary tree: array of records 2

Challenge 2

The image below shows a binary tree with nine nodes: B, C, D, F, G, K, L, M, and N. The root of the tree is node B.



A binary tree with nine nodes

A binary tree can be implemented as an array of records. The following table represents the array of records for the binary tree in the image.
Drag and drop the given values in the correct spaces to complete the array. Each value can be used more than once.

Index	Data	Left	Right
0	B	1	2
1	C	3	
2	D		
3	F		7
4	G	Null	Null
5	K	Null	
6	L	Null	Null
7	M	Null	Null
8	N		

Items:

4 Null 8 5 6

Quiz:
STEM SMART Computer Science Week 47

Linear queue: operations 2

Practice 2

Sam's aunty wants to make sure that everyone at his party gets a piece of cake before they go for a second slice.

Sam's aunty makes the friends queue up for a piece of cake. Brandon joins the queue first, followed by Millie, Dorris, Adele, Robin, and Joe. Sam takes his place at the rear of the queue as he is very well mannered! If somone has eaten their cake and wants another piece, they join the queue once more.

The diagram in Figure 1 below represents a **linear queue** that is used to manage the process. The cake has 16 pieces so there are 16 positions (elements) in the queue. The **front pointer** points to the start of the queue. The **rear pointer** points to the end of the queue. At the start, the queue is empty so the front and rear pointers both point to the first element which is numbered 0.

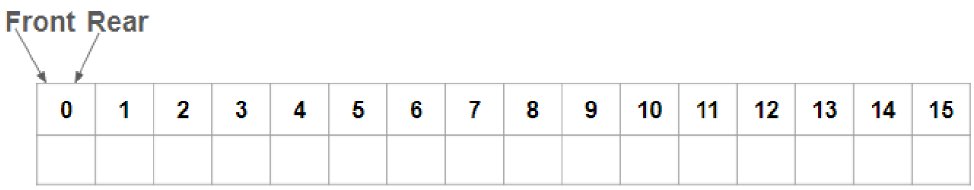


Figure 1: The cake queue

Part A Front pointer 1

To which position will the **front pointer** point after everyone has joined the queue for their first piece of cake?

Part B Rear pointer 1

To which position will the **rear pointer** point once everyone (including Sam) has joined the queue for their first piece of cake?

Part C Front pointer 2

To which position will the **front pointer** point after Brandon and Millie (who joined the queue first) have been given their first piece of cake?

Part D Rear pointer 2

To which position will the **rear pointer** point after Brandon and Millie have finished their cake and have rejoined the queue for a second slice, while everyone else is still waiting for their first piece of cake?

Part E Rear pointer 3

To which position will the **rear pointer** point after everyone has joined the queue for a second piece of cake and been given another slice?

Tree traversals

Challenge 2

A robotics control system uses binary trees to organise the decision-making process for robot movements. The decision structures can be traversed using a variety of methods.

Describe the following methods of traversal: inorder, post-order, and pre-order.

Part A Inorder

Describe an inorder traversal.

[2 marks]

Part B Post-order

Describe a post-order traversal.

[2 marks]

Part C Pre-order

Describe a pre-order traversal.

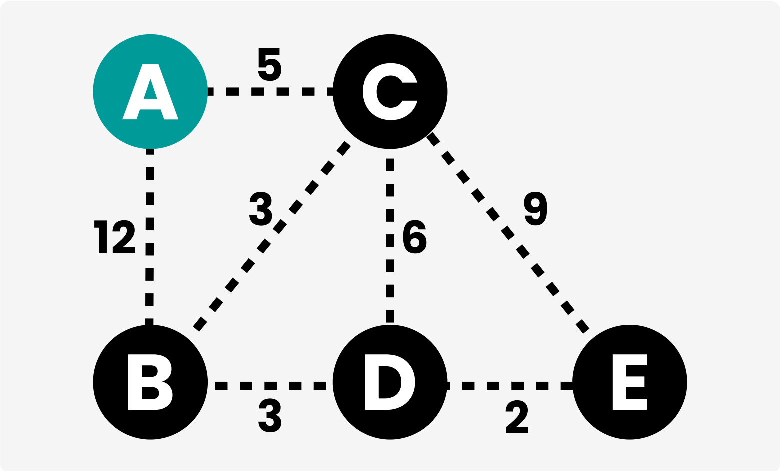
[2 marks]

Dijkstra: trace 2

Practice 2

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 Dijkstra's shortest path algorithm?

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

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	8	C
C	5	A
D	11	C
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

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 - B - D - E
- A - C - E
- A - C - D - E

Dijkstra: trace 4

Practice 2

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.

A*: trace 2

Challenge 2

A computer game uses the A* pathfinding 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 A* 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 node will the A* algorithm choose as the next node to visit?

- P
- H
- C
- N