Accredited

AS and A LEVEL

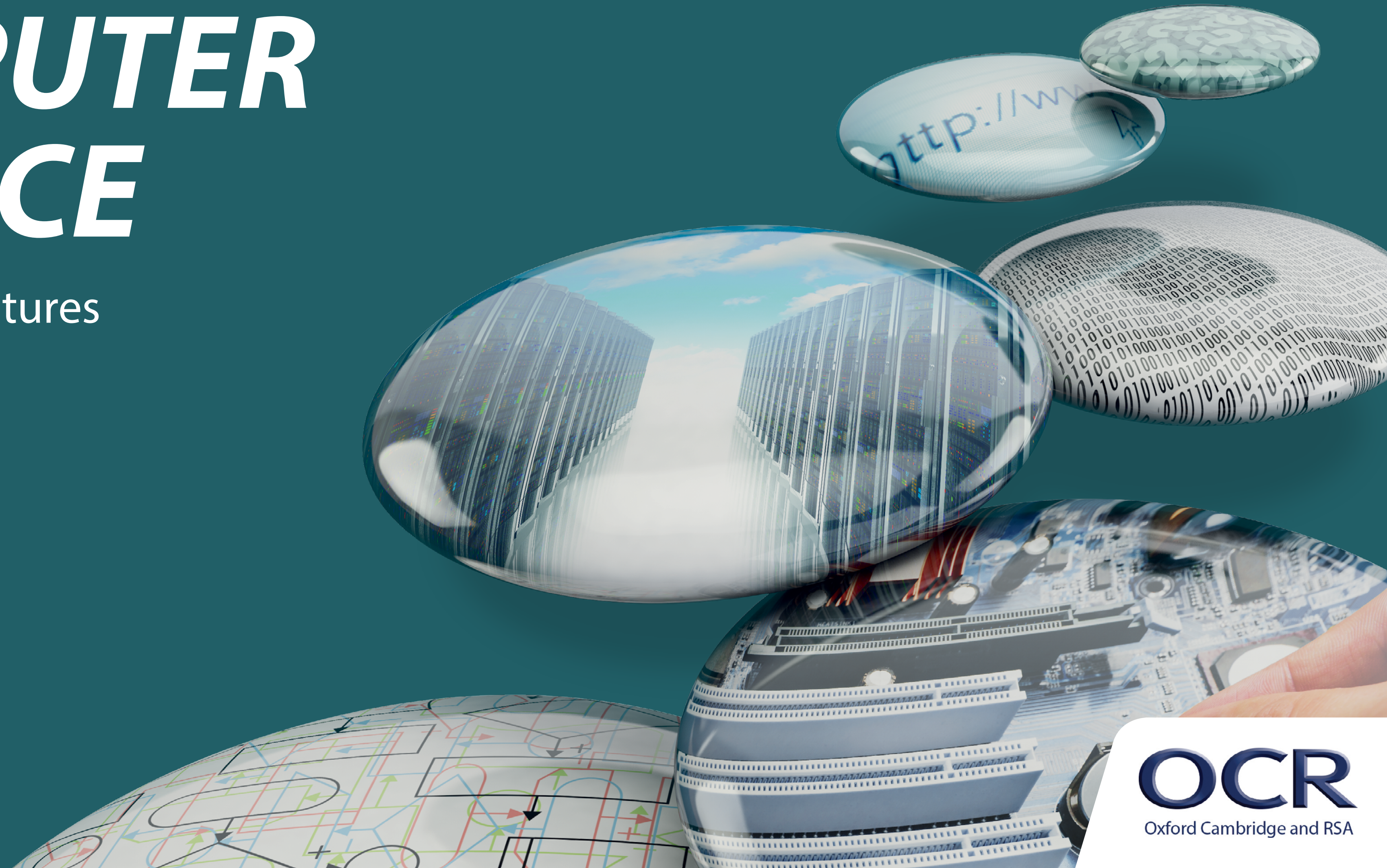
Delivery Guide

H046/H446

COMPUTER SCIENCE

Theme: Data structures

July 2015



OCR
Oxford Cambridge and RSA

Oxford Cambridge and RSA Examinations is a Company Limited by Guarantee.
Registered in England. Registered company number 3484466.

Registered office: 1 Hills Road
Cambridge
CB1 2EU

OCR is an exempt charity.

CONTENTS

Introduction

Delivery guides are designed to represent a body of knowledge about teaching a particular topic and contain:

- Content: A clear outline of the content covered by the delivery guide;
- Thinking Conceptually: Expert guidance on the key concepts involved, common difficulties students may have, approaches to teaching that can help students understand these concepts and how this topic links conceptually to other areas of the subject;
- Thinking Contextually: A range of suggested teaching activities using a variety of themes so that different activities can be selected which best suit particular classes, learning styles or teaching approaches.

If you have any feedback on this Delivery Guide or suggestions for other resources you would like OCR to develop, please email resourcesfeedback@ocr.org.uk.

KEY

Click to view associated resources within this document.

Click here — Click to view external resources

AS Level only — AS Level content only

Curriculum Content

a) Arrays (of up to 3 dimensions), records, lists, tuples.

b) The following structures to store data: linked-list, graph (directed and undirected), stack, queue, tree, binary search tree, hash table.

c) How to create, traverse, add data to and remove data from the data structures mentioned above. (NB this can be either using arrays and procedural programming or an object-oriented approach).

The following websites are detailed here as activities to self-teach or as revision aids. Some of the websites also give example questions that have been developed specifically for delivering this knowledge at A-Level standard.

Activities	Resources
Data structures and Algorithms (Sue Sentance and Adam McNicol) http://www.pythonschool.net/category/data-structures-algorithms.html Designed to be part of a day course on data structures in Python. Covers Stacks, Queues, Linked Lists and Binary Trees. Implementations in Python 3	Click here
How to think like a computer scientist: Learning with Python 3 http://www.ict.ru.ac.za/Resources/cspw/thinkcspy3/thinkcspy3_latest/linked_lists.html Covers Linked lists, Stacks, Queues, Trees in lots of detail. Implementations in Python 3	Click here
Problem solving with Algorithms and Data structures http://interactivepython.org/courselib/static/pythonds/index.html Very detailed in-depth look at Lists, Dictionaries, Linked lists, Stacks, Graphs, trees, queues and hashing. Implementations in Python 3	Click here
Data structures (Teach ICT) http://www.teach-ict.com/as_as_computing/ocr/H447/F453/3_3_5/data_structures/theory_datastructures.html Covers lists, linked-lists, stacks, queues, trees and binary trees	Click here

Thinking Conceptually

Data structures in relation to Data Types from 1.4.1 outline the relationships between how data is related and how this data can be manipulated. Typical data structures will have specific ways to add, insert, remove, get the next or last element and to locate elements.

It is useful to draw parallels between data structures and their real-life counterparts. For instance, you could say that

- arrays are a bit like a CD rack that has numbers on
- a stack is a lot like a stack of books in that if you want to get to the books at the bottom you can just 'pop' books off the stack until you reach the bottom. This is thought of as a LIFO system (last in first out), whereas a Queue – just like a real queue – is FIFO (First in first out) just like going on a rollercoaster.
- Trees can be related to the DOM (Document Object Model) hierarchy, where a page contains many HTML elements that are interrelated
- Graphs can be found in Networking data models.
- A linked-list is a lot like a scavenger hunt, where you have to find each clue, and then pick up the next clue that 'points' to the next clue.

Common misconceptions or difficulties students may have

Learning about data structures is similar to learning about data types. For a lot of students these concepts will be brand new and can only be related to computers, but some can be expressed in terms of real life problems such as queues.

Students may find that the differences between lists and tuples are hard to notice, but the main differences are that tuples cannot be altered through insertion, removal or updating elements whereas lists can. Lists use square brackets and tuples use curved brackets.

Students' learning and understanding will benefit through having worked-through examples to reference and being able to implement the data structures themselves.

Conceptual links to other areas of the specification – useful ways to approach this topic to set students up for topics later in the course.

Data structures are usually the next field to study after students have a full grasp of Data Types. It would be unusual to study these in reverse order.

ACTIVITIES

Unless the student has studied computer science before, it is unlikely that they have come across many of the data structures present since we are unlikely to encounter them outside the realm of the subject area.

Activities	Resources
The below activity gets students to 'act out' linear and binary searches as well as using hashing through a battleships style activity where students must work in pairs. This activity would be useful before implementing binary trees and hashing in code. Binary numbers (CSUnplugged) http://csunplugged.org/searching-algorithms Activity is in many different languages along with other interpretations and links to similar activities. **Covers linear search, binary search, hashing**	Click here
Data structures – Questions worksheet Provided is learner resource 1 which gives some questions based on some of the topics that have been covered in order to test students. Links have been given in the 'content' section to enable students to find answers if they get stuck.	Learner Resource 1 Teacher Resource 1

Thinking Contextually

Activities	Resources
Data structure cards Learner resource 2 is given as a set of resources that you can use with students to help them visualise how different data structures work. You could get students to cut out the cards given at the top of the document and then ask for groups of two to spend a minute thinking how they could use the materials to represent that data structure. When you walk around the class, the pair needs to explain what is going on and you can help to address any misconceptions asking questions such as "how would you go about deleting an item from a linked-list"?	Learner Resource 2
Tree Traversal Learner resource 3 is designed to help students understand tree traversal algorithms pre-order, in-order and post-order. Students can construct their own trees using the cut-outs and then write the tree down using the templates beneath. They can then use the table to write down the traversal output for each tree or can use the trick detailed in the worksheet for determining the right order using dots on the nodes. For more information: Tree Traversal (Wikipedia) http://en.wikipedia.org/wiki/Tree_traversal Tree traversal (Prof.Robert C. Holte - University of Ottowa) http://webdocs.cs.ualberta.ca/~holte/T26/tree-traversal.html	Learner Resource 3 Click here Click here

Learner Resource 1 Questions worksheet

See page 7

Data structures

1. What is the difference between static and dynamic data structures?

2. What is the difference between mutable and immutable data structures?

Arrays

3. What type of data structure is an array? (Static/Dynamic?) (Mutable/immutable?)

4. How many elements are in the array Clowns(4,7)?

5. Array Footballteams:

Man united – index 0 in the array	Tottenham	Ipswich
Arsenal	West Ham	Chelsea

a) What is found at the location Footballteams(1)?

b) What is found at the location Footballteams(4)?

c) What is the location of Ipswich in the array?

d) What is the location of Chelsea in the array?

Lists and linkedlists

6. What do we mean by a static data structure?

7. What do we mean by a 'dynamic list'?

8. What are some drawbacks on having a non-dynamic list?

9. What information needs to be in each item of a linked list?

10. Draw the stack after each of the following commands, starting with an empty stack. What does the stack achieve:

1. Push 'Tim'	State 1:
2. Push 'Jack'	State 2:
3. Push 'Peter'	State 3:
4. Push 'Tom'	State 4:
5. Pop	State 5:
6. Pop	State 6:
7. Pop	State 7:

11. What would the commands be to get from:

Bagels		Bun
Sandwiches	To	Panini
Donut		Donut

Queues

12. If a stack is LIFO, what is a queue?

13. Sometimes if your computer is busy the things you are typing do not appear on the screen right away. What might this have to do with a queue?

14. Why would you want to use a circular queue as opposed to a linear queue?

Learner Resource 2 Data structure cards

See page 8

Core items

Food (ID: 01)	Fruit (ID: 02)	Vegetable (ID: 03)
Grape (ID: 04)	Banana (ID: 05)	Orange (ID: 06)
Apple (ID: 07)	Cox Apple (ID: 08)	Granny Smith Apple (ID: 09)
Carrot (ID: 10)	Onion (ID: 11)	Broccoli (ID: 12)
Bean (ID: 13)	Runner bean (ID: 14)	Broad bean (ID: 15)

Duplication of cards

Food (ID: 01)	Fruit (ID: 02)	Vegetable (ID: 03)
Grape (ID: 04)	Banana (ID: 05)	Orange (ID: 06)
Apple (ID: 07)	Cox Apple (ID: 08)	Granny Smith Apple (ID: 09)
Carrot (ID: 10)	Onion (ID: 11)	Broccoli (ID: 12)
Bean (ID: 13)	Runner bean (ID: 14)	Broad bean (ID: 15)

Pointer arrows (Used with Queue, Circular queue and Stack)

Arrays and hash tables

Arrays

Have a go at placing items into the grid below and how the numbering system works. If you remove an item, this item will be left blank.

Hash tables

The same table can be used to represent hash tables. Simply add together the two digits of the item ID and place them into the table at that location.

Queues

A queue uses a pointer to the front of the queue and the next free memory location. It does not reuse memory.

Circular queues

Circular queues use 3 pointers; One to the start of the buffer, one to the start of the data, and one to the end of the data.

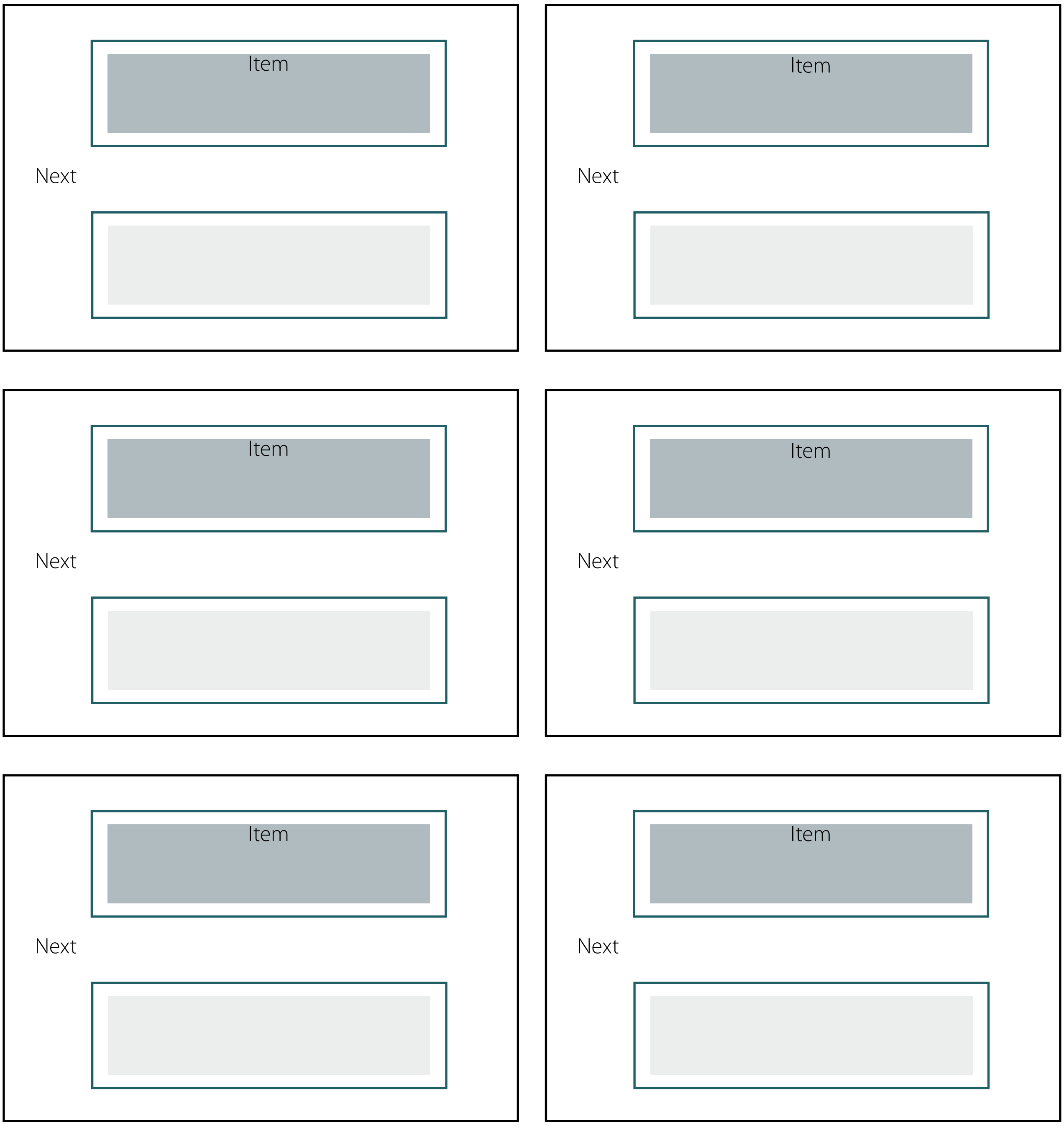
		Pointer
0		
1		
2		
3		
4		
5		
6		
7		
8		
9		

Stacks

A stack is a Last In First Out system (LIFO)

- Pushing: Adds a new specified item to the top of the stack
- Popping: Removes the item from the top of the stack.

		Pointer
0		
1		
2		
3		
4		
5		
6		
7		


Item
Next
Item
Next
Item
Next
Item
Next
Item
Next
Item
Next

Tree and binary tree

Queues and stacks are linear lists and so each data item only points to the one before or after it. They are in order, but they do not imply there is a relationship between the items

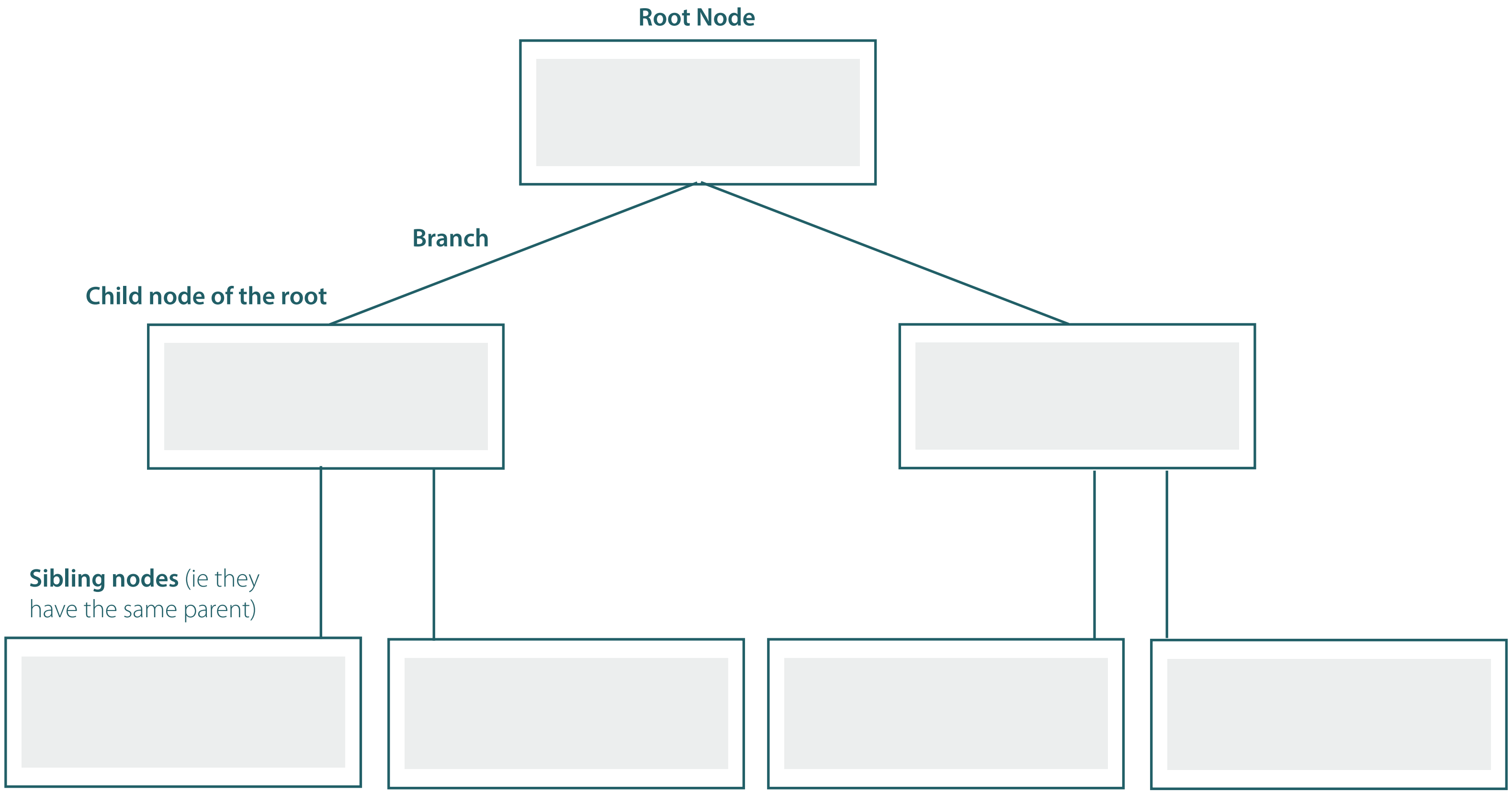
A tree data structure is used to represent relationships between items.

Tree

A tree or 'general tree' allows for any number of child nodes for a given node. If a node has no children it is called a 'leaf'. The 'root' is the very first node at the top of the tree.

Binary tree

Binary trees can only have 2 children per 'parent' node. These are referred to as the left and right sub-trees (or left/right children).



Learner Resource 3 Tree Traversal

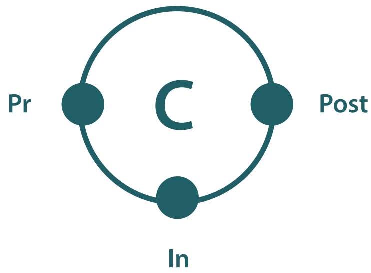
See page 8

Traversing a tree

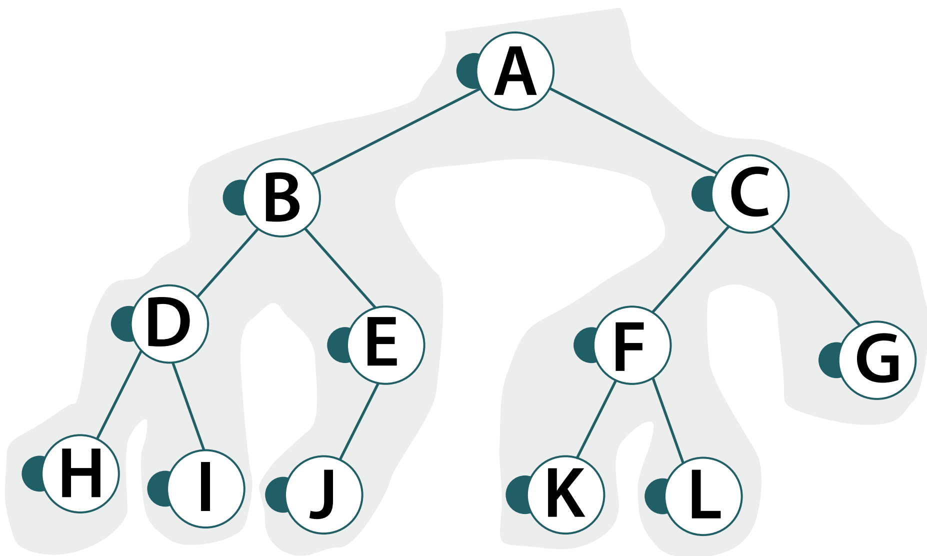
Pre-order	In-order	Post-order
Root	Traverse to Left sub-tree	Traverse to Left sub-tree
Traverse to Left sub-tree	Root	Traverse to Right sub-tree
Traverse to Right sub-tree	Traverse to Right sub-tree	Root

Helpful trick

If you put a dot corresponding to the location on the diagram below on all of the nodes and trace a line around the tree, marking off each dot as you go, this will give you the output of the tree traversal.



For instance, on the below tree, if we use the pre-order traversal then we place the dot on the left hand side

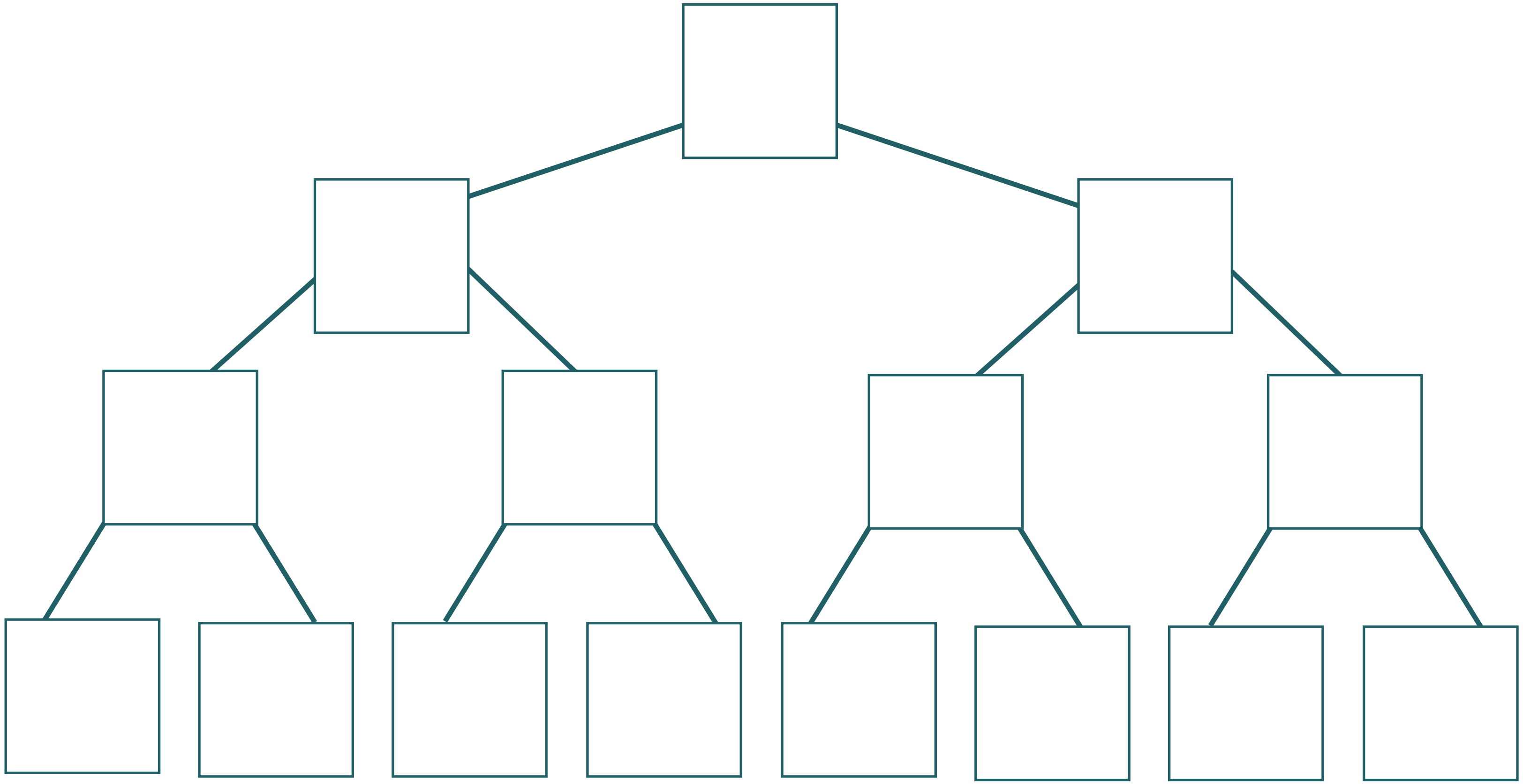


The output would therefore be A, B, D, H, I, E, J, C, F, K, L, G

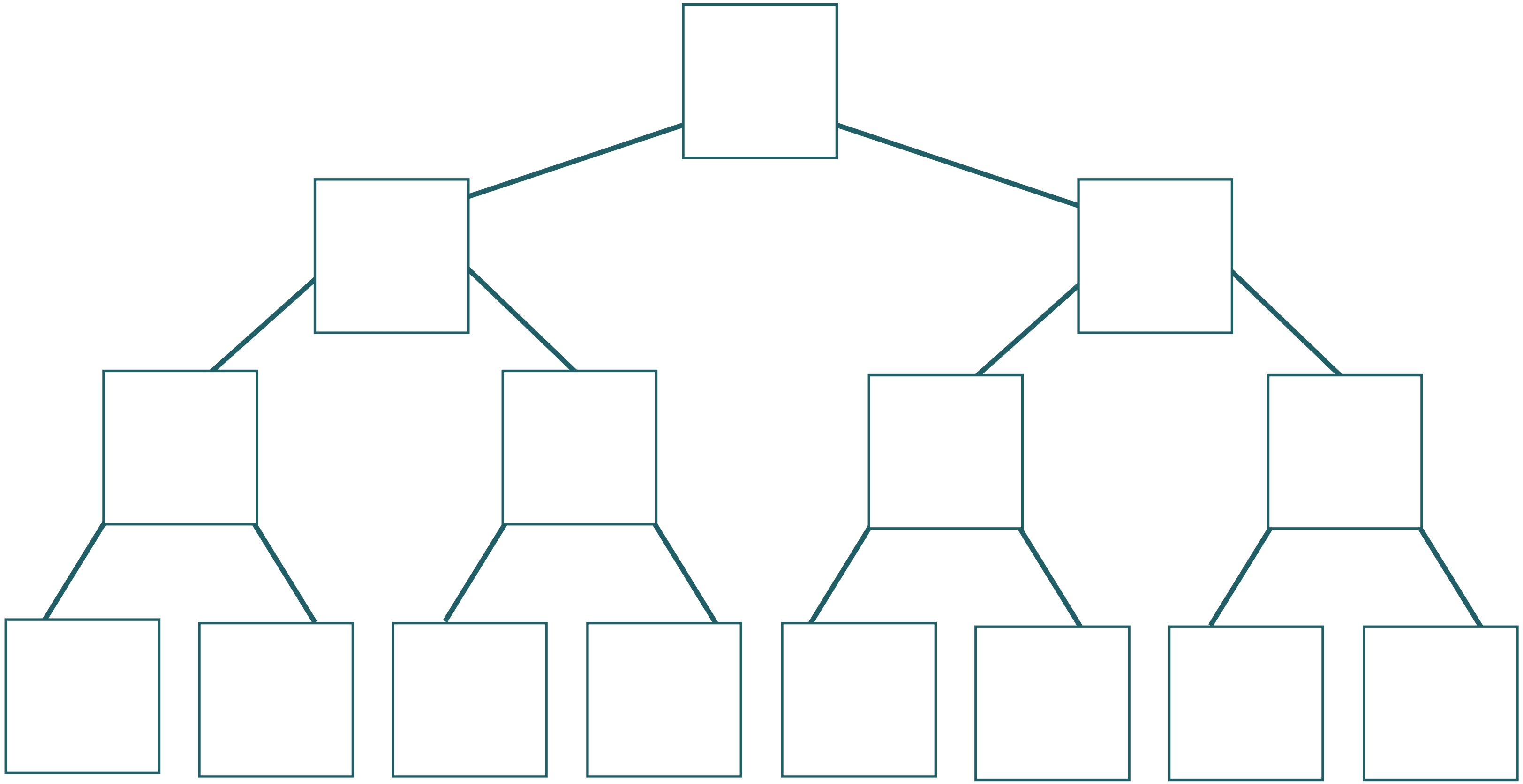
Traversing a tree

A	B	C	D
E	F	G	H
I	J	K	L

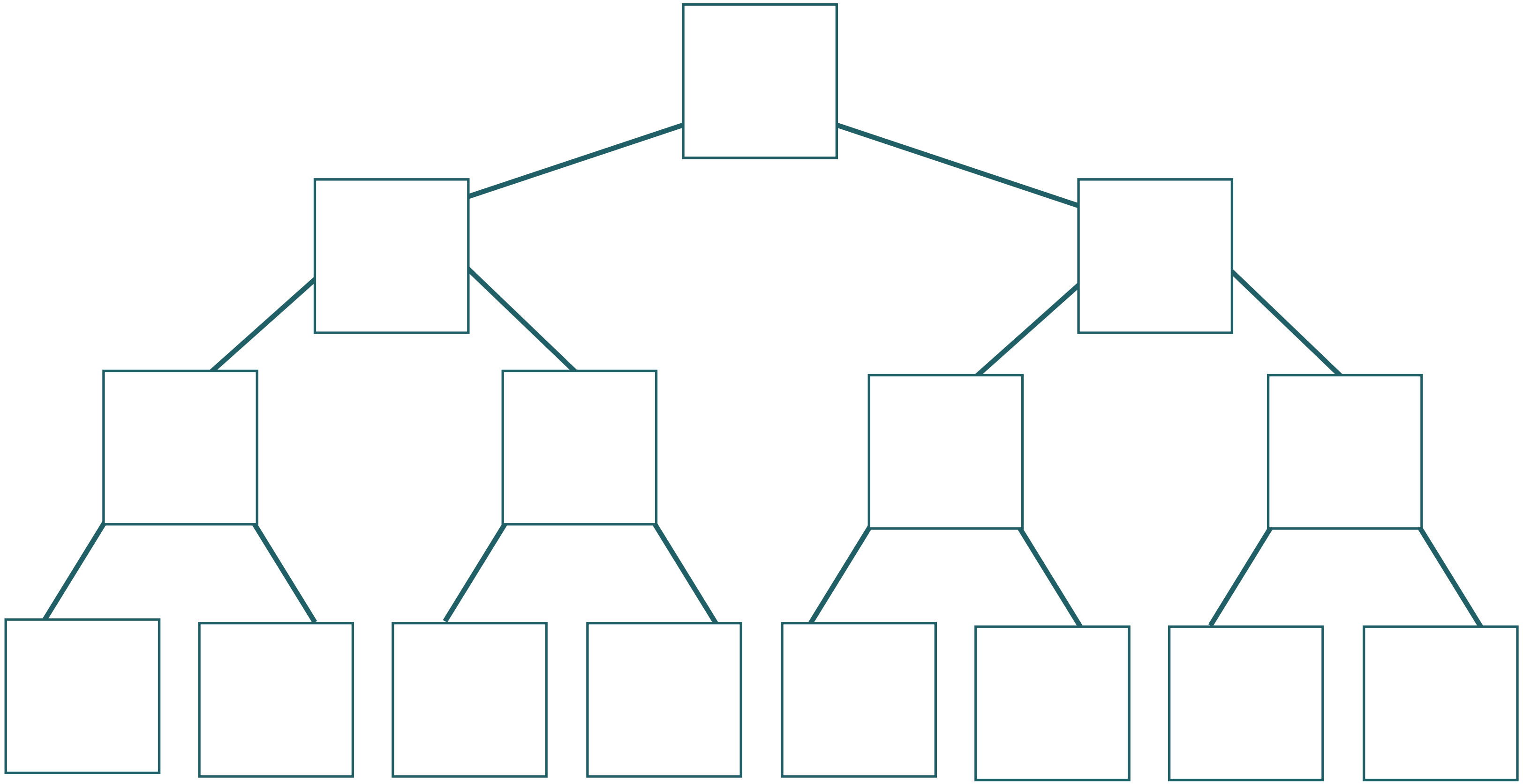
Practice trees



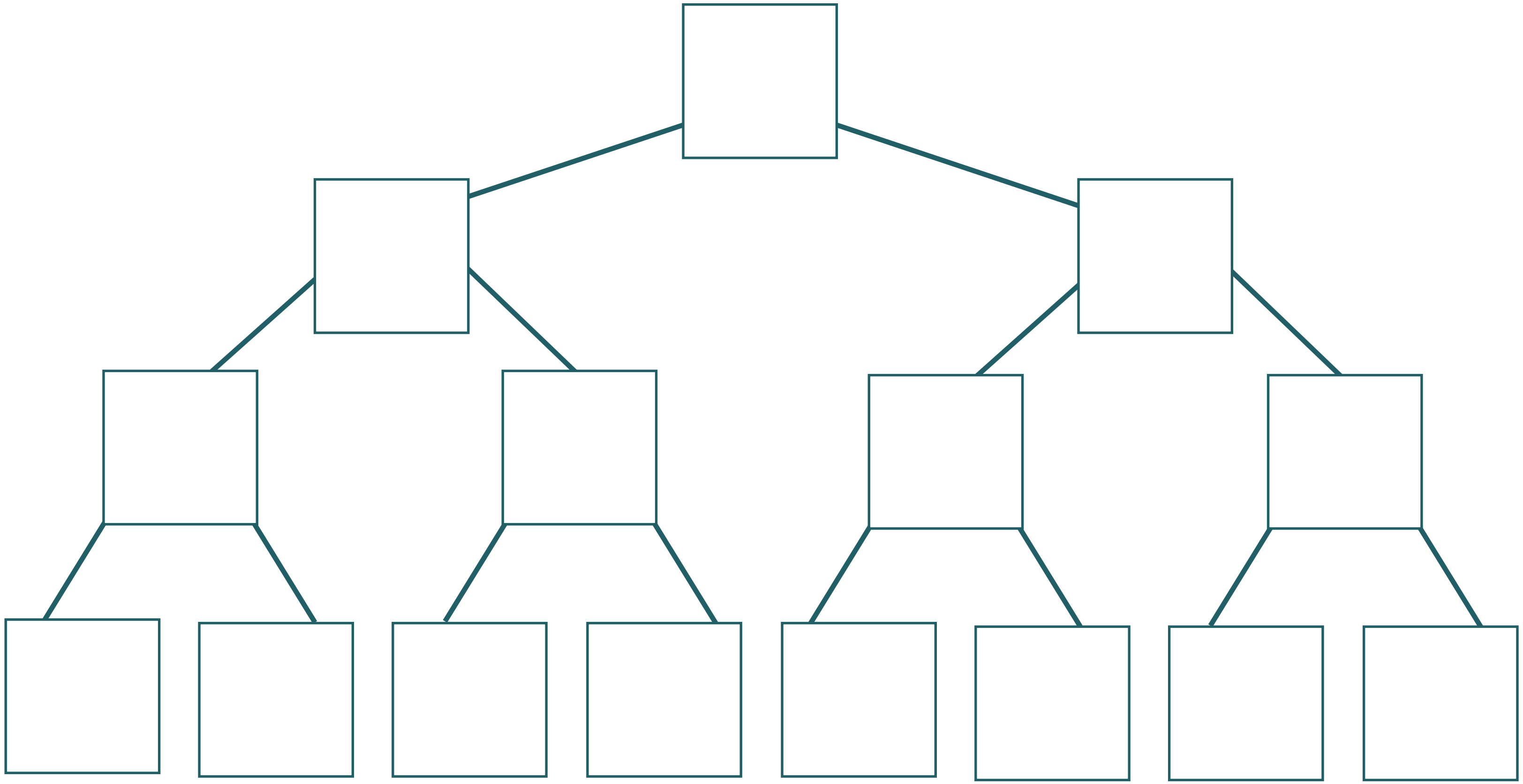
Practice trees 1



Practice trees 2



Practice trees 3



Teacher Resource 1 Questions worksheet

See page 7

Data structures

1. What is the difference between static and dynamic data structures?

A static data structure is fixed in size, but dynamic structures can increase of decrease in size.

2. What is the difference between mutable and immutable data structures?

A mutable structure allows data to be changed (such as deletion, insertion etc)

3. What type of data structure is an array? (Static/Dynamic?) (Mutable/immutable?)

An array is static and mutable. It cannot change size but items can be added, edited moved or deleted.

4. How many elements are in the array Clowns(4,7)?

4*7 = 28

5. Array Footballteams:

Man united – index 0 in the array	Tottenham	Ipswich
Arsenal	West Ham	Chelsea

a) What is found at the location Footballteams(1)? Tottenham

b) What is found at the location Footballteams(4)? West Ham

c) What is the location of Ipswich in the array? 2

d) What is the location of Chelsea in the array? 5

Lists and linkedlists

6. What do we mean by a static data structure?

A static data structure is one that has a fixed size and cannot change at run time

7. What do we mean by a 'dynamic list'?

A dynamic list is able to adapt to accommodate the data inside it and so it does not waste as much space.

8. What are some drawbacks on having a non-dynamic list?

If your list is full you cannot add any more elements, and if the list is empty or not completely full, you are wasting the space that you are not using.

9. What information needs to be in each item of a linked list?

The data and a pointer to the next item

10. Draw the stack after each of the following commands, starting with an empty stack. What does the stack achieve:

1. Push 'Tim'
2. Push 'Jack'
3. Push 'Peter'
4. Push 'Tom'
5. Pop
6. Pop
7. Pop

State 1:	Tim			
State 2:	Jack	Tim		
State 3:	Peter	Jack	Tim	
State 4:	Tom	Peter	Jack	Tim
State 5:	Output Tom			
State 6:	Output Peter			
State 7:	Output Jack			

11. What would the commands be to get from:

Bagels		Bun
Sandwiches	To	Panini
Donut		Donut

Pop, Pop,Push 'Panini', Push 'Bun'

Queues

12. If a stack is LIFO, what is a queue?

FIFO

13. Sometimes if your computer is busy the things you are typing do not appear on the screen right away. What might this have to do with a queue?

The keyboard uses a queue buffer so that no 'keystrokes' are 'lost' if the computer happens to be busy at any moment

14. Why would you want to use a circular queue as opposed to a linear queue?

If the end of the available spaces are reached then the next item will use any free spaces at the start of the queue, which is more memory efficient.

We'd like to know your view on the resources we produce. By clicking on the 'Like' or 'Dislike' button you can help us to ensure that our resources work for you. When the email template pops up please add additional comments if you wish and then just click 'Send'. Thank you.

If you do not currently offer this OCR qualification but would like to do so, please complete the Expression of Interest Form which can be found here: www.ocr.org.uk/expression-of-interest

OCR customer contact centre
General qualifications
Telephone 01223 553998
Facsimile 01223 552627
Email general.qualifications@ocr.org.uk

ISO 9001 UKAS MANAGEMENT SYSTEMS 001

A DIVISION OF CAMBRIDGE ASSESSMENT