

OCR Computer Science A Level

1.4.2 Data Structures

Flashcards



What is an array?



What is an array?

An ordered, finite set of elements of a single type.



What type of array is a linear array?



What type of array is a linear array?

A one-dimensional array.



How can you visualise a two-dimensional array?



How can you visualise a two dimensional array?

As a spreadsheet or table.



How can you visualise a three dimensional array?



How can you visualise a three dimensional array?

A three dimensional array can be visualised as a multi-page spreadsheet.



What is a record also known as?



What is a record also known as?

A row in a file



What is a record made up of?



What is a record made up of?

A record is made up of fields.



How can you select a field from a record using pseudocode?



How can you select a field from a record using pseudocode?

```
recordName.fieldName
```



What is the definition of a list?



What is the definition of a list?

A data structure consisting of a number of items in which the items can occur more than once.



What are the main differences between arrays and lists?



What are the main differences between arrays and lists?

- Lists can store data non-contiguously whereas arrays store data in order.
- Lists can store data of more than one data type.



What is a tuple?



What is a tuple?

An immutable, ordered set of values of any type.



What is the difference between a tuple
and an array?



What is the difference between a tuple and an array?

Tuples are initialised using regular brackets instead of square brackets.



What is a linked list?



What is a linked list?

A dynamic data structure used to hold an ordered set of items which are not stored in a contiguous locations.



What is the name given to the items in a linked list?



What is the name given to the items in a linked list?

Nodes



What does each item in a linked list contain?



What does each item in a linked list contain?

It contains a data field and another address field called a link/pointer



What is a data field in a linked list?



What is a data field in a linked list?

A field that stores the actual data.



What is a pointer field in a linked list?



What is a pointer field in a linked list?

A field that contains the address of the next item in the list.



What is a graph?



What is a graph?

A graph is a data structure consisting of a set of vertices/nodes connected by edges/arcs.



What is a directed graph?



What is a directed graph?

A graph in which the edges can only be traversed in one direction.



What is an undirected graph?



What is an undirected graph?

A graph in which the edges can be traversed in both directions.



What is a weighted graph?



What is a weighted graph?

A graph in which the arcs/edges have a cost to traverse



Give two ways of representing graphs so that they can be understood by computers.



What two data types can a computer understand a graph?

Adjacency Matrix, Adjacency List



What are the advantages of using an adjacency matrix?



What are the advantages of using an Adjacency Matrix?

- Convenient to work with
- Easy to add new nodes to



What are the advantages of using an Adjacency List?



What are the advantages of using an Adjacency Matrix?

Space efficient for large sparse networks



What is a stack?



What is a stack?

A LIFO ('last in first out') data structure, where items can only be removed from and added to the top of the list.



Give an example of where stacks may be used.



Give an example of where stacks may be used.

- Back button in a web page
- Undo buttons



What is a queue?



What is a queue?

A FIFO ('first in first out') data structure, where items are added to the end of the queue and removed from the front of the queue.



What is a tree?



What is a tree?

A data structure which has a root node, and child nodes connected with branches.



What is a node?



What is a node?

A node is any item in a tree



What is an edge?



What is an edge?

An edge is the connection between two nodes



What is the root node?



What is the root node?

The node which doesn't have any incoming nodes, at the top of the tree



What is a child?



What is a child?

Any node which has an incoming edge



What is a parent?



What is a parent?

Any node which has outgoing edges



What is a subtree?



What is a subtree?

A section of the tree which consists of a parent and all the children of that parent



What is a leaf?



What is a leaf?

A leaf is a node which has no children



What is a binary tree?



What is a binary tree?

A type of tree in which each node has a maximum of two children



What is the purpose of a binary tree?



What is the purpose of a binary tree?

A binary tree is used to search for values quickly.



What is a hash table?



What is a hash table?

A hash table is an array which is coupled with a hash function



What is a collision? (hashing)



What is a collision? (hashing)

A collision is where two inputs result in the same hashed value



What properties does a good hashing algorithm have?



What properties does a good hashing algorithm have?

A good hashing algorithm must have a low chance of collision and must be fast.



What is pre-order traversal?



What is a pre-order traversal?

Traversal algorithm in which you traverse the root node, followed by the left subtree then the right subtree.



What is in-order traversal?



What is in-order traversal?

Traversal algorithm in which you traverse the left subtree, the root node, then the right subtree.



What is post-order traversal?



What is post-order traversal?

Traversal algorithm in which you traverse the left subtree, the right subtree followed by the root node.



What does the operation `isEmpty()` do?
(*Lists*)



What does the operation isEmpty() do?

Checks if the list is empty



What does the operation `append(value)`
do? (*Lists*)



What does the operation `append(value)` do?

It adds the given value to the end of the list.



What does the operation `remove(value)`
do? (*Lists*)



What does the operation remove value do?

It finds the first instance of the given value and removes it



What does the operation `search(value)`
do? (*Lists*)



What does the operation `search(value)` do?

Searches for the given value in a list



What does the operation `length()` do?
(*Lists*)



What does the operation `length()` do?

Returns the length of the list



What does the operation `index(value)`
do? (*Lists*)



What does the operation `index(value)` do?

Returns the index of a value in the list.



What does the operation insert(position, value) do? (*Lists*)



What does the operation `insert(position, value)` do?

Adds the value to the position in the list.



What does the operation `pop()` do (*Lists*)



What does the operation `pop()` do?

Returns and removes the last value in the list



What does `pop(position)` do? (*Lists*)



What does pop(position) do?

It returns and removes the value at the given position in the list.



What does the operation `isEmpty()` do?
(*Stacks*)



What does the operation isEmpty() do?

Checks to see if the stack is empty



What does the operation `push(value)`
do? (*Stacks*)



What does the operation `push(value)` do?

Adds the given value to the top of the stack



What does the operation `peek()` do?
(*Stacks*)



What does the operation peek() do?

Returns the top value of the stack



What does the operation `pop()` do?
(*Stacks*)



What does the operation `pop()` do?

Returns and removes the value at the top of the stack



What does the operation `size()` do?
(*Stacks*)



What does the operation `size()` do?

It returns the size of the stack



What does the operation `isFull()` do?
(*Stacks*)



What does the operation `isFull()` do?

Checks to see if the stack is full.



What does the operation
`enQueue(value)` do? (*Queue*)



What does the operation enQueue(value) do?

Adds the given value to the end of the queue



What does the operation `deQueue()` do?
(*Queue*)



What does the operation deQueue() do?

Removes the item from the end of the queue



What does the operation `isEmpty()` do?
(*Queue*)



What does the operation isEmpty() do?

It checks to see if the queue is empty



What does the operation `isFull()` do?
(*Queue*)



What does the operation `isFull()` do?

Checks to see if the queue is full

