

AQA Computer Science A-Level
4.3.2 Tree-traversal
Intermediate Notes



Specification:

4.3.2.1 Simple tree-traversal algorithms

Be able to trace the tree-traversal algorithms:

- pre-order
- post-order
- in-order.

Be able to describe uses of tree-traversal algorithms. Pre-Order: copying a tree. In-Order: binary search tree, outputting the contents of a binary search tree in ascending order. Post-Order: Infix to RPN (Reverse Polish Notation) conversions, producing a postfix expression from an expression tree, emptying a tree.



Tree-Traversal

Synoptic Link

A tree is a **connected acyclic graph**.

Trees are covered in **Trees** under **Fundamentals of Data Structures**.

Algorithm

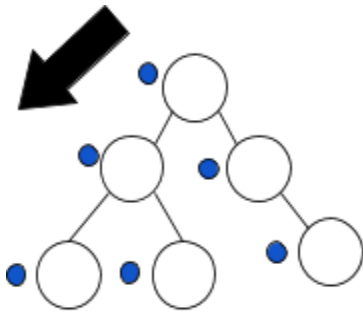
An algorithm is a set of instructions which **completes a task in a finite time and always terminates**.

Tree-traversal is the **process of visiting/updating/outputting each node** in a **tree** - it is a form of **algorithm**. Unlike a **graph-traversal**, tree-traversals are **unique to trees** and must **start at the root**. From the root, they travel left, down the tree. There are three types of tree-traversals; **pre-order, in-order and post-order**. **Pre-order and post-order tree-traversal** can be performed on **any tree including binary trees** but an **in-order traversal** is only well defined for **binary trees**.

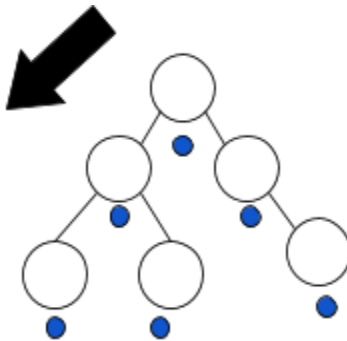
Synoptic Link

Graph-traversal is the process of **visiting each vertex in a graph**

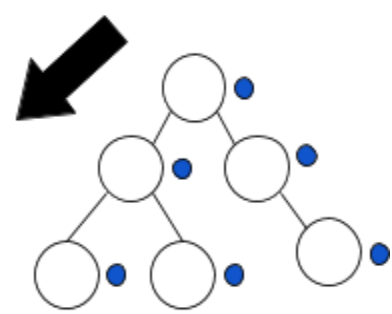
Graph-traversal is covered in **Graph-traversal** under **Fundamentals of Algorithms**.



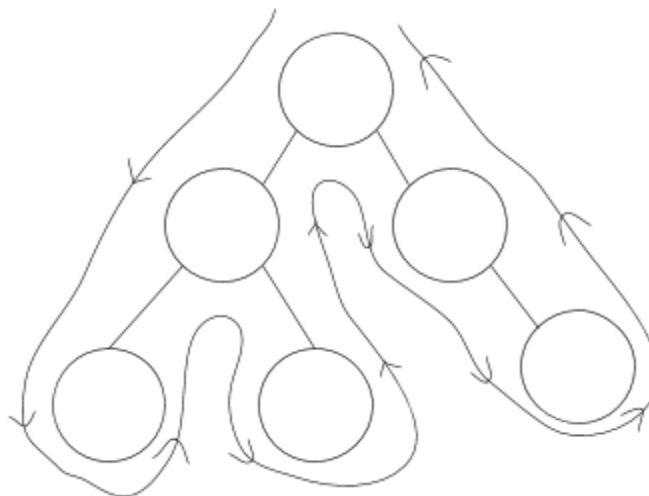
Pre-Order



In-Order



Pre-Order



The journey around a tree always occurs like this.



Pre-Order Traversal

Pre-order traversal is used for **copying a tree**. It can be performed on **any tree**.

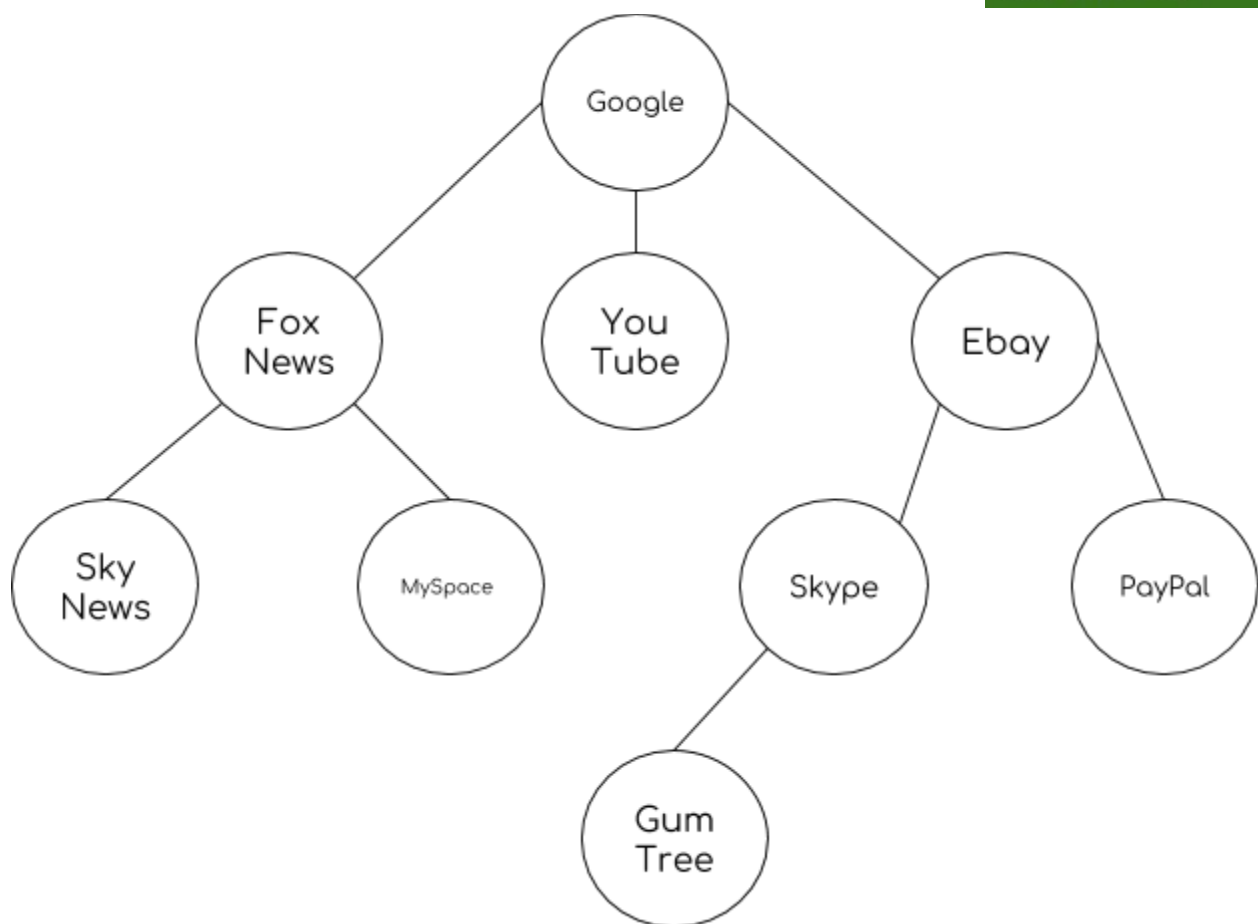
Example:

Here is a tree which shows the hierarchical relationships of companies owned by Google. This is **not** a **binary tree** because Google has more than two children. In this example, Google is the **root**.

Synoptic Link

A **binary tree** is a **rooted tree** where each node has at most **two** children. The **root node** has no parent.

Binary trees are covered in **Trees** under **Fundamentals of Data Structures**.

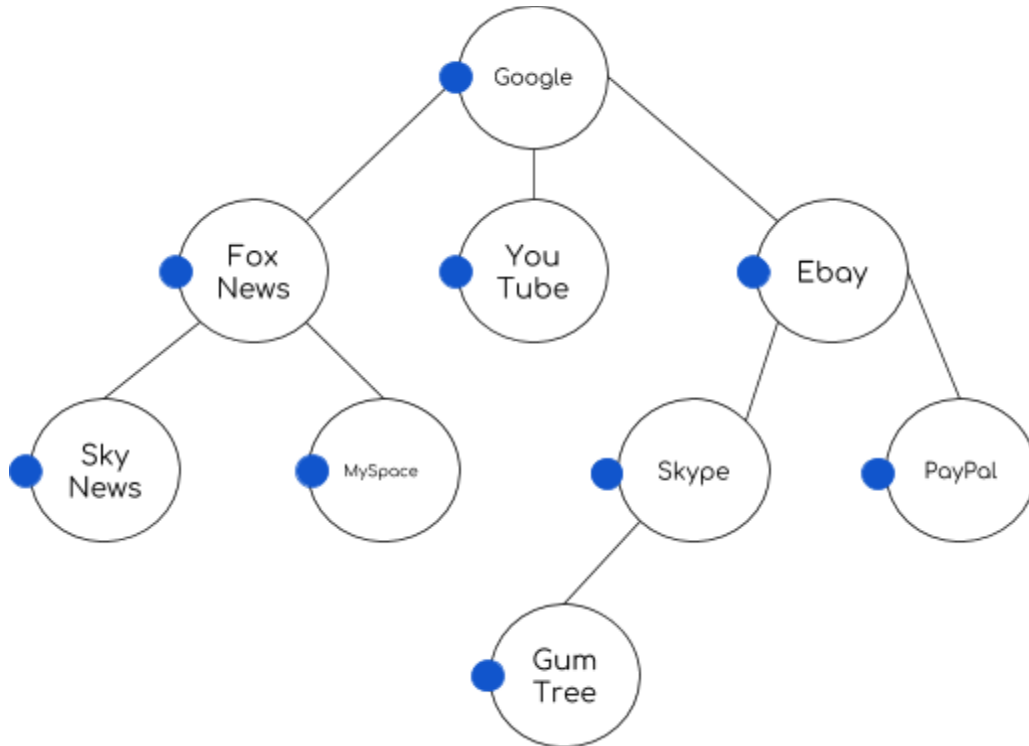


Leaf

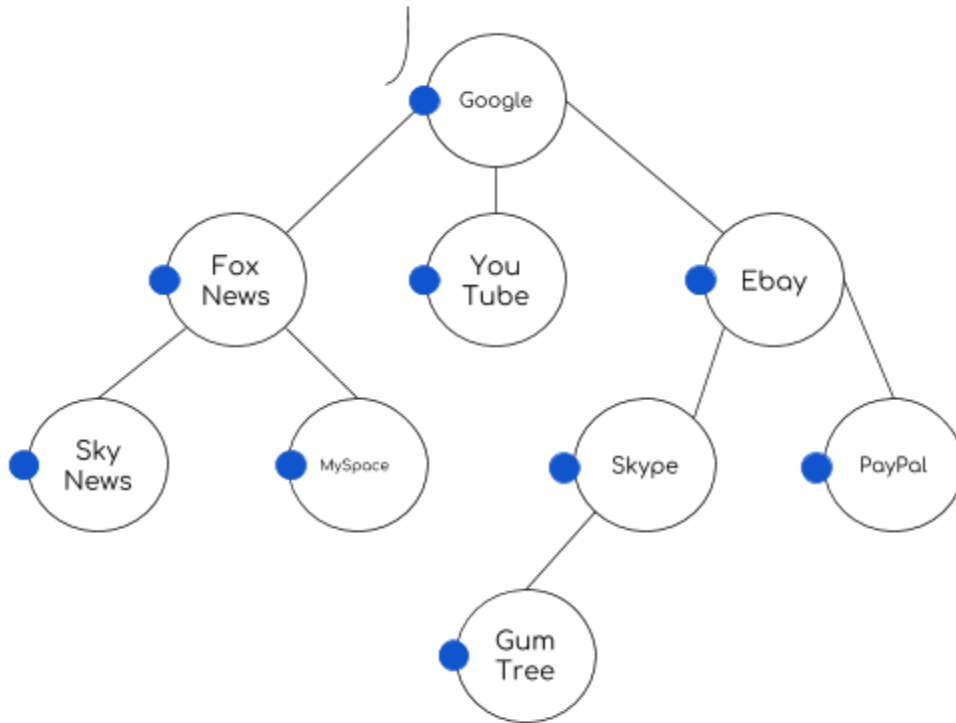
A vertex/node can also be known as a leaf when it is on a tree.

When performing a pre-order traversal the first step is to mark the **left** hand side of each **leaf**.





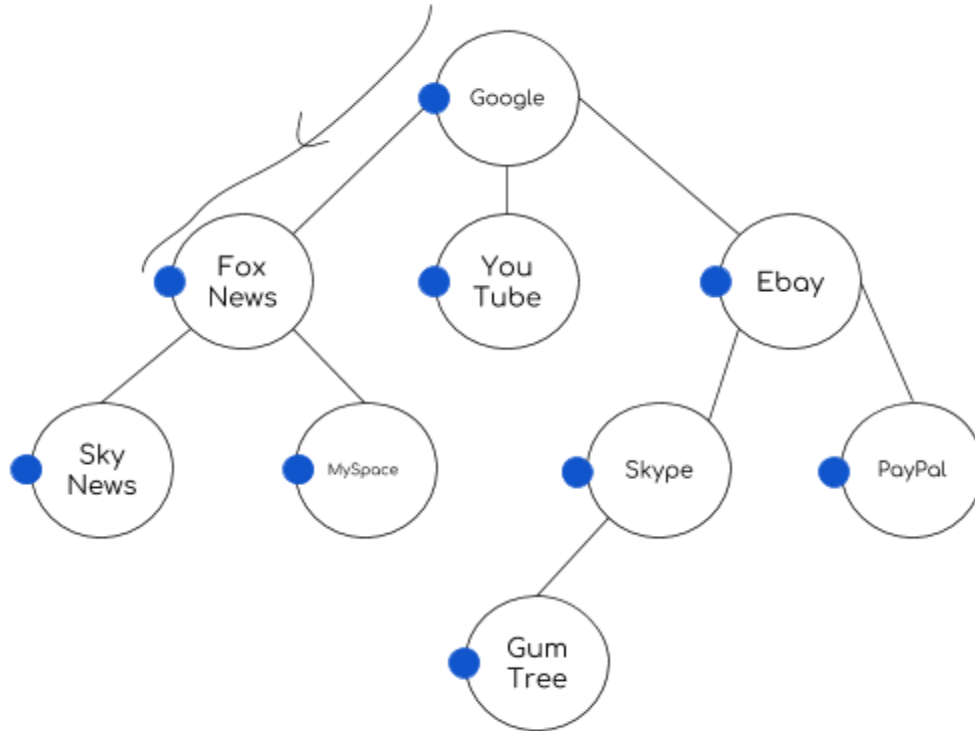
The traversal starts from the left and works down the tree. Whenever a blue spot is passed, the information on the node is outputted.



Output: Google

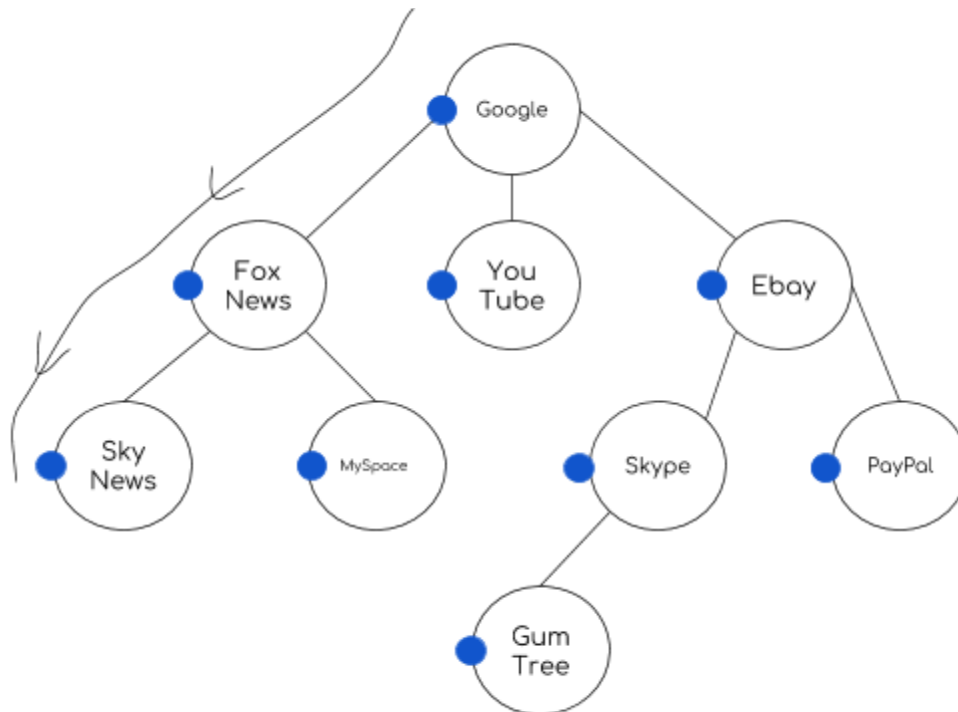
The blue spot on Google has been passed, so Google is outputted.





Output: Google, Fox News

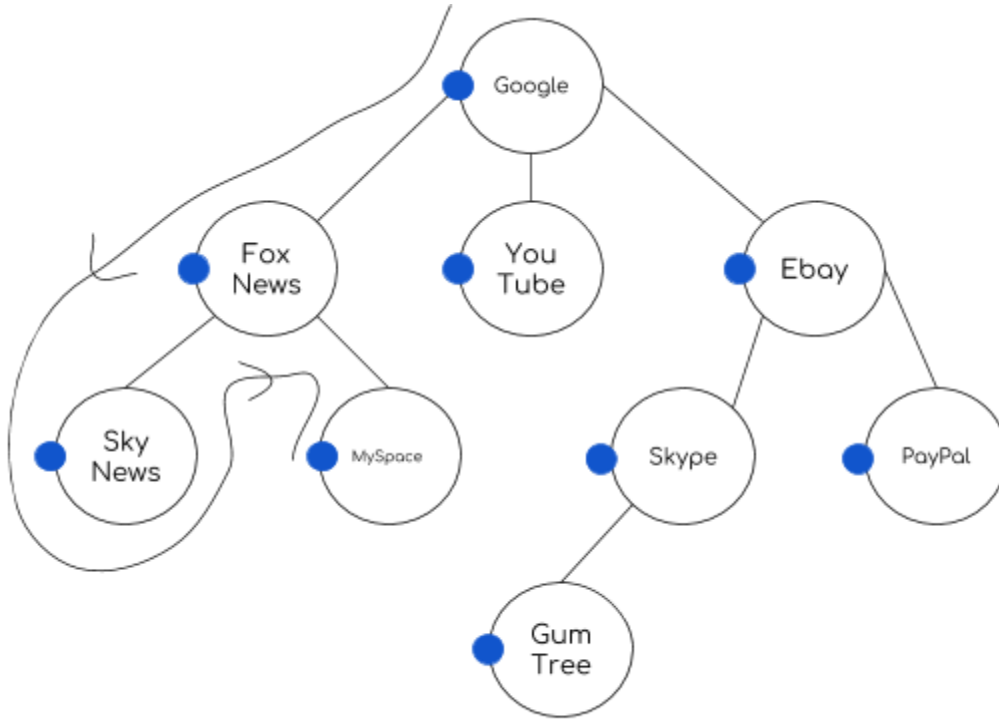
Fox News has been passed, and outputted.



Output: Google, Fox News, Sky News

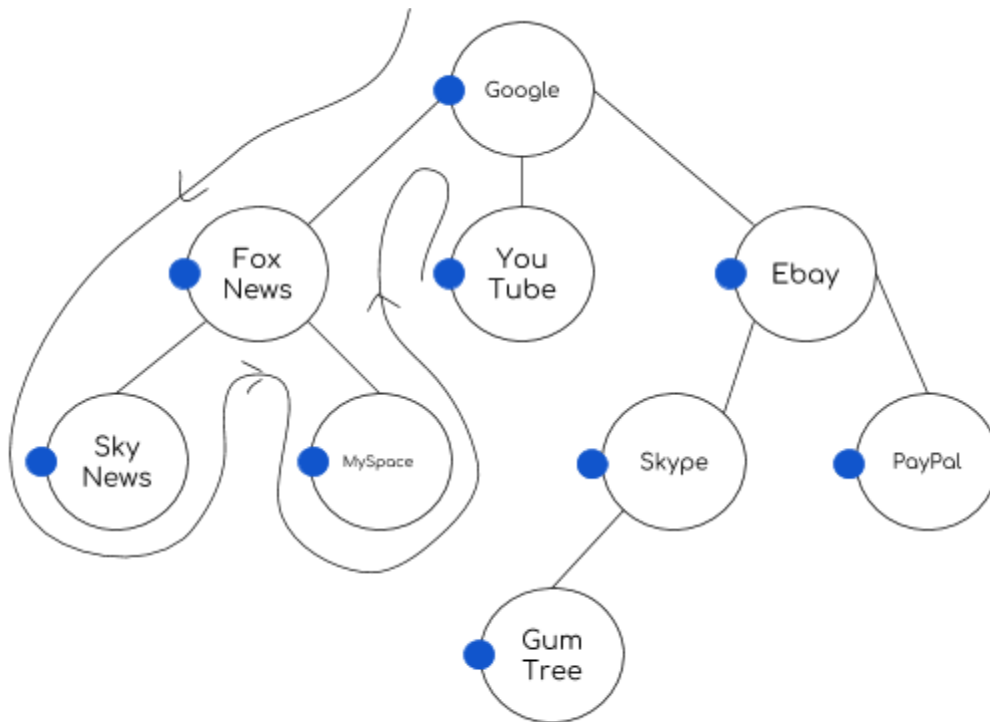
Sky News has been passed and outputted.





Output: Google, Fox News, Sky News, MySpace

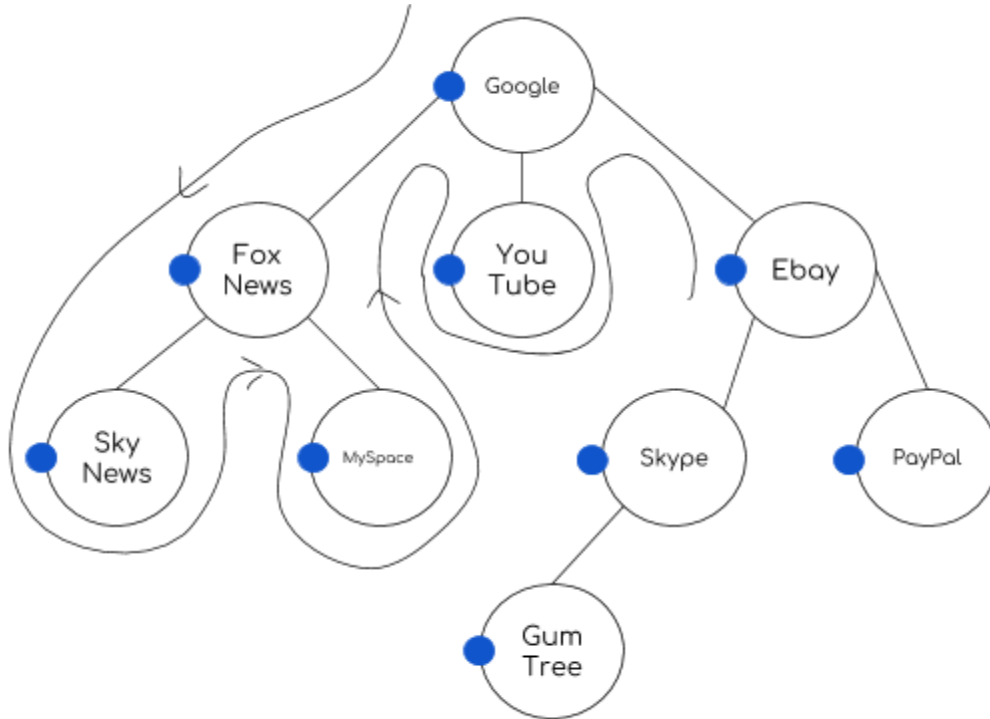
MySpace has been passed and outputted.



Output: Google, Fox News, Sky News, MySpace, YouTube

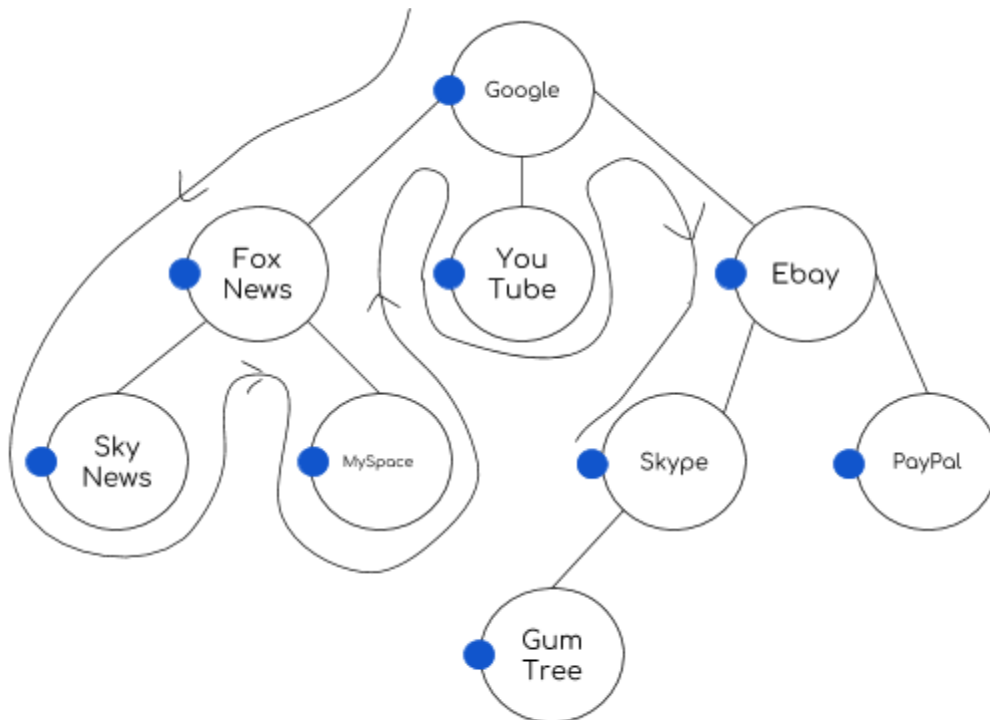
Youtube has been passed and outputted.





Output: Google, Fox News, Sky News, MySpace, YouTube, Ebay

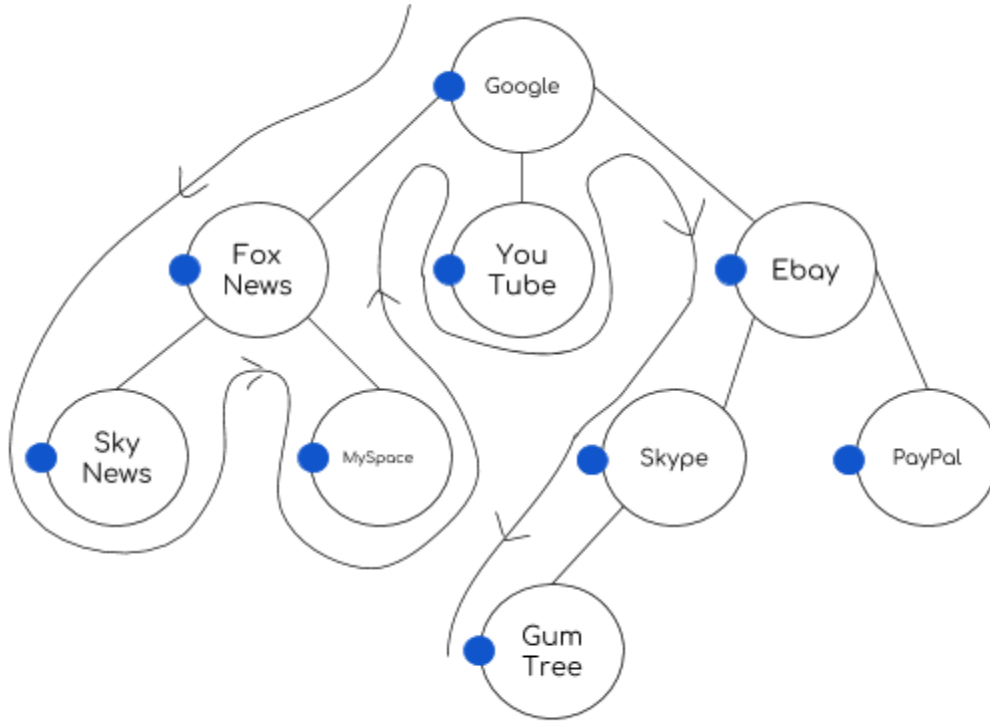
Ebay has been passed and outputted.



Output: Google, Fox News, Sky News, MySpace, YouTube, Ebay, Skype

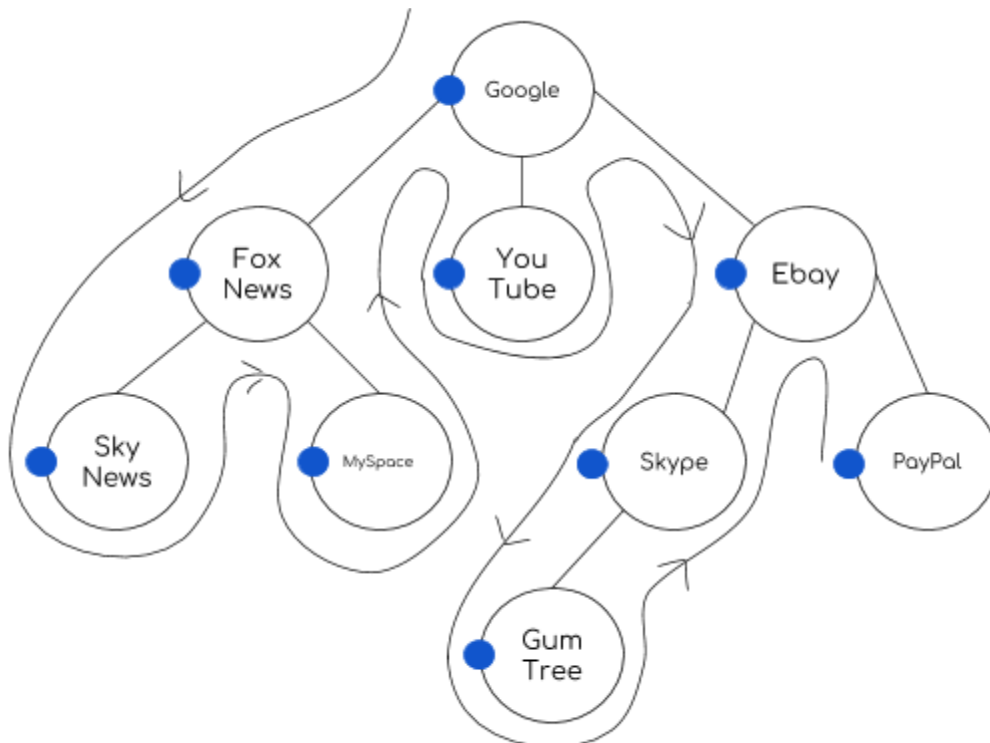
Skype has been passed and outputted.





Output: Google, Fox News, Sky News, MySpace, YouTube, Ebay, Skype, GumTree

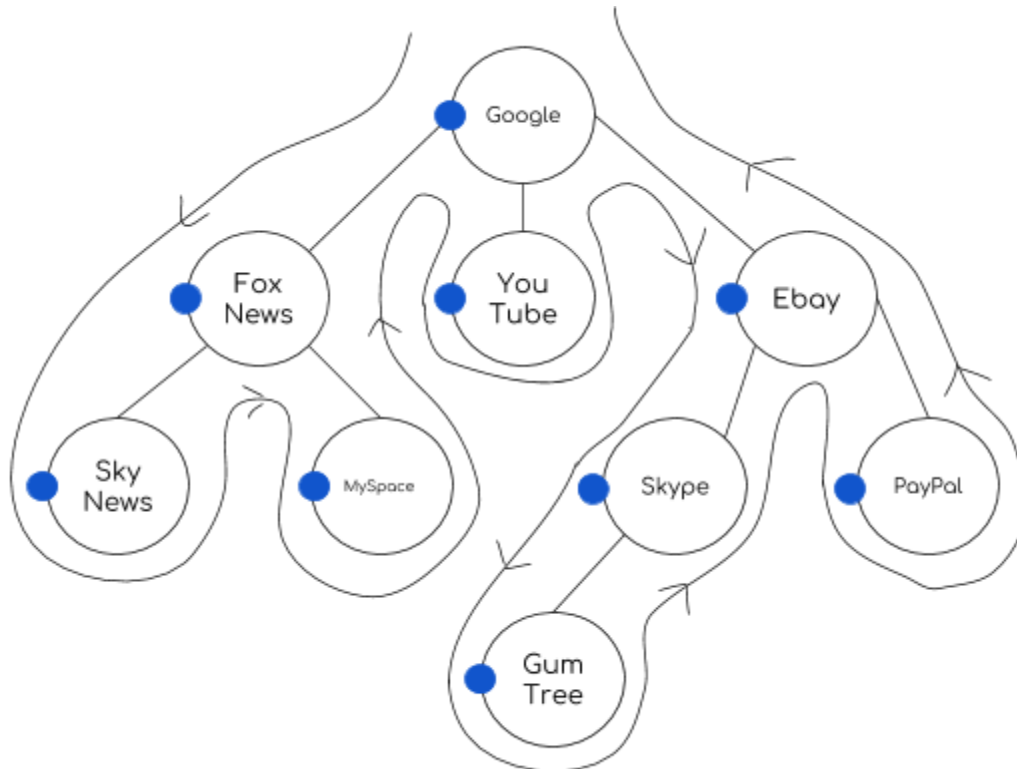
GumTree has been passed and outputted.



Output: Google, Fox News, Sky News, MySpace, YouTube, Ebay, Skype, GumTree, Paypal

PayPal has been passed and outputted.





Output: Google, Fox News, Sky News, MySpace, YouTube, Ebay, Skype, GumTree, Paypal

Note

Traversals are algorithms and algorithms always terminate.

The traversal has **completed**.

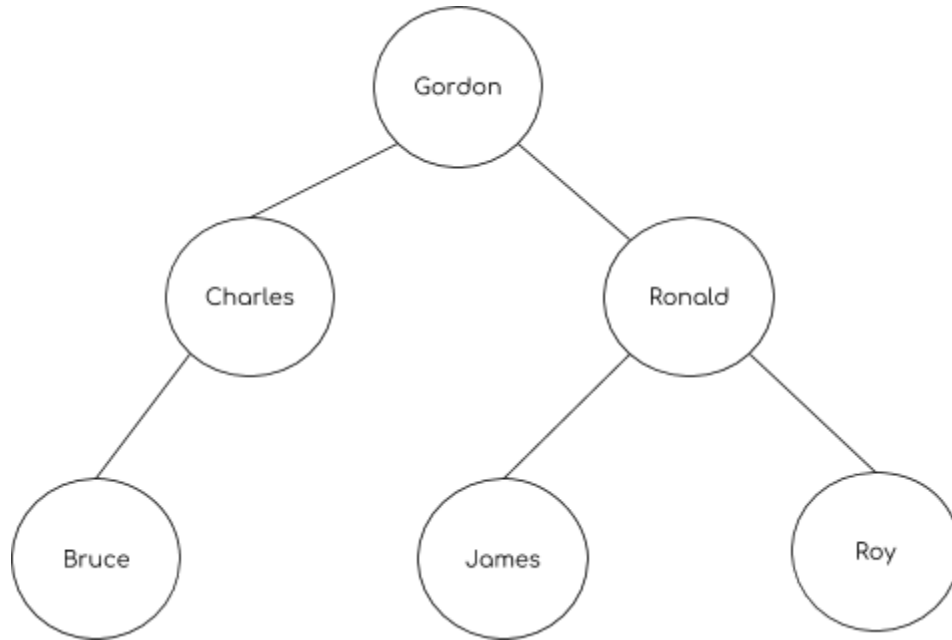
In-Order Traversal

In-order traversal is useful for a **binary search tree** and because it will **output the contents of a binary search tree in ascending order**. It can only be performed on **binary trees**.

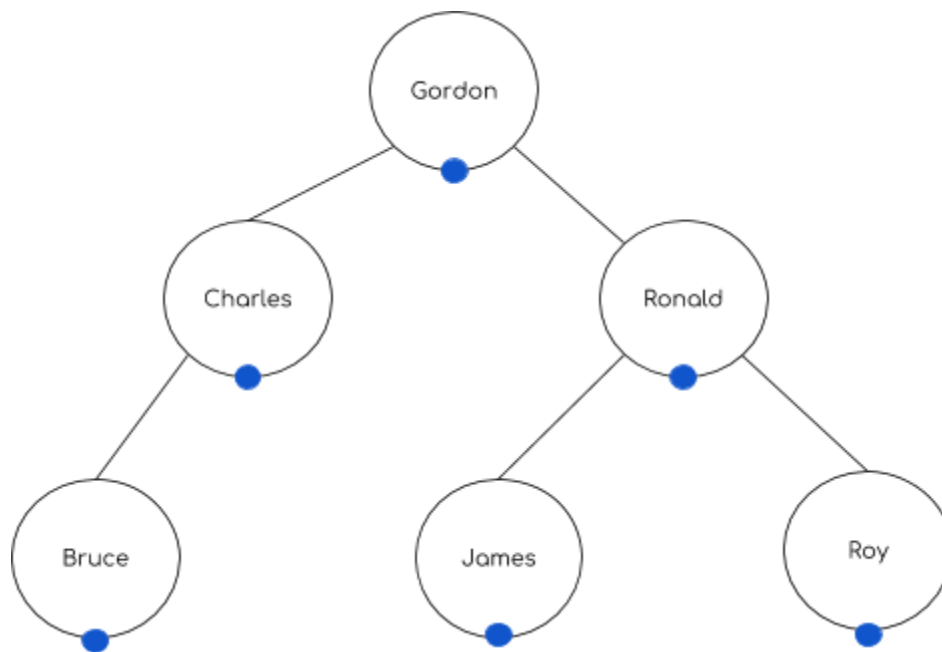
Example:

Here is a binary tree.



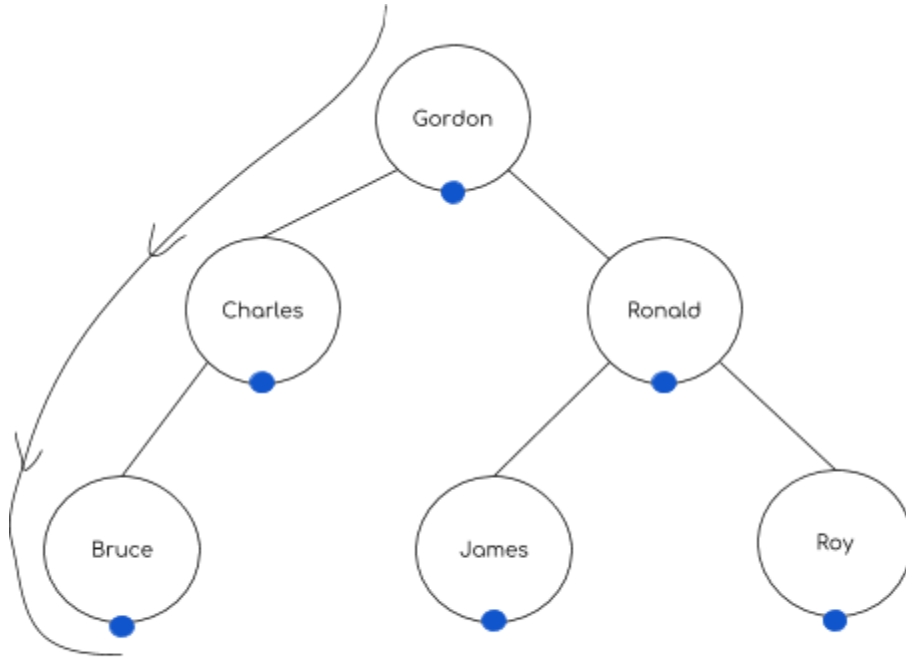


The first step in an in-order traversal is to mark the bottom of the children and parents.



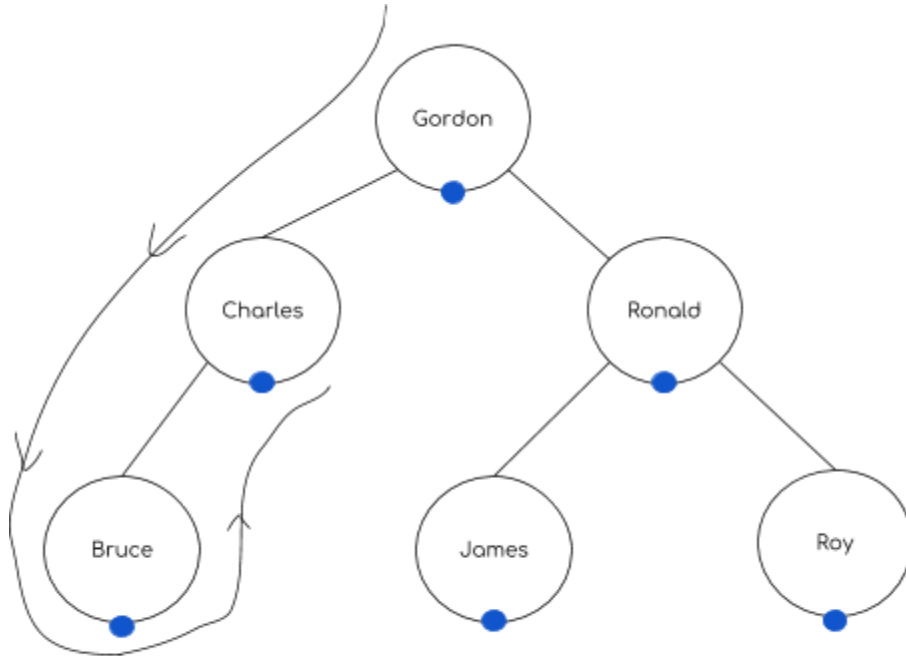
The traversal starts from the left and works around the tree. When a blue spot is passed, the node is outputted.





Output: Bruce

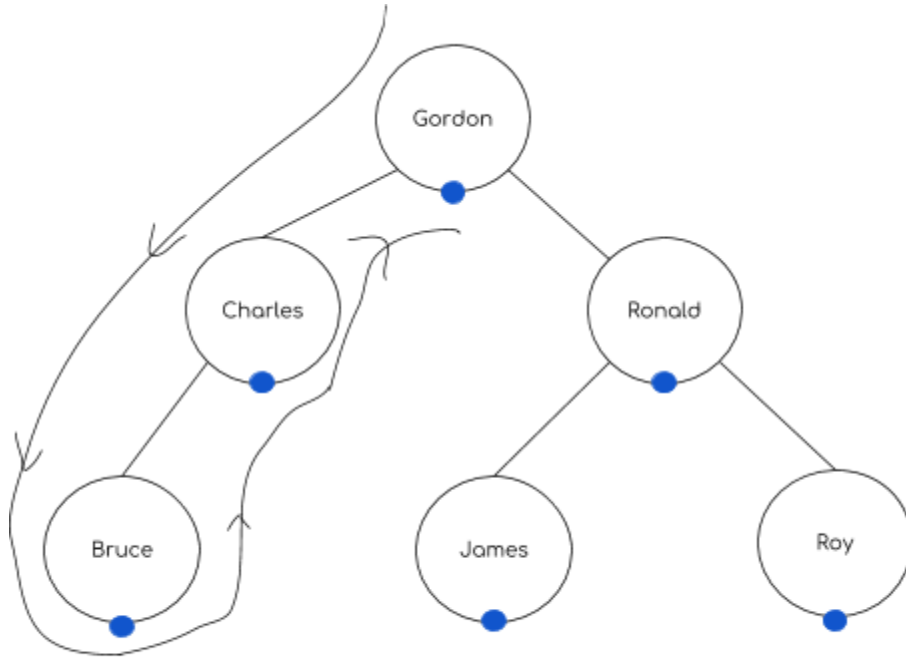
Bruce has been passed and outputted.



Output: Bruce, Charles

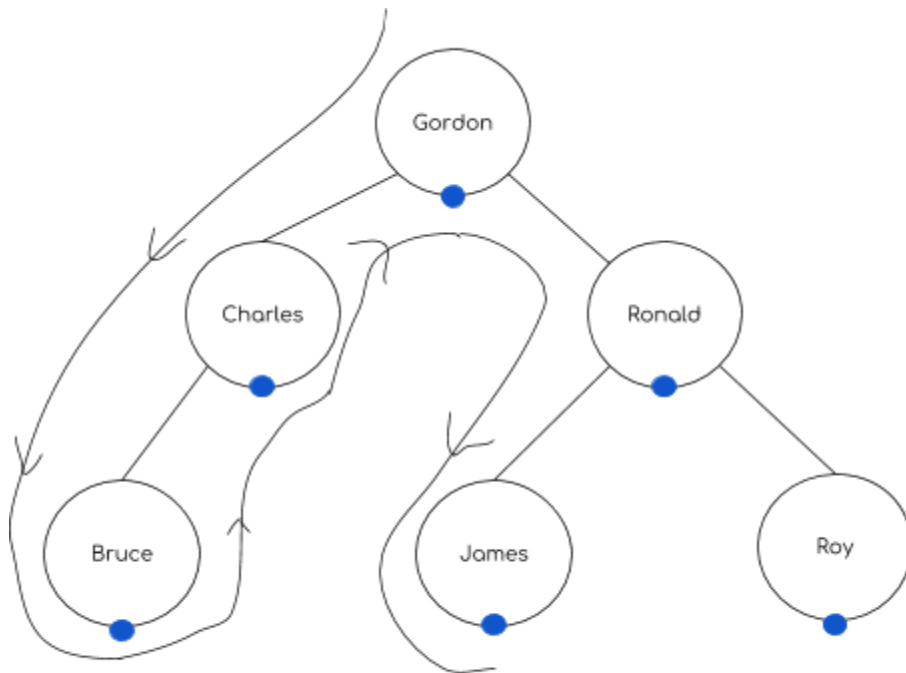
Charles has been passed and outputted.





Output: Bruce, Charles, Gordon

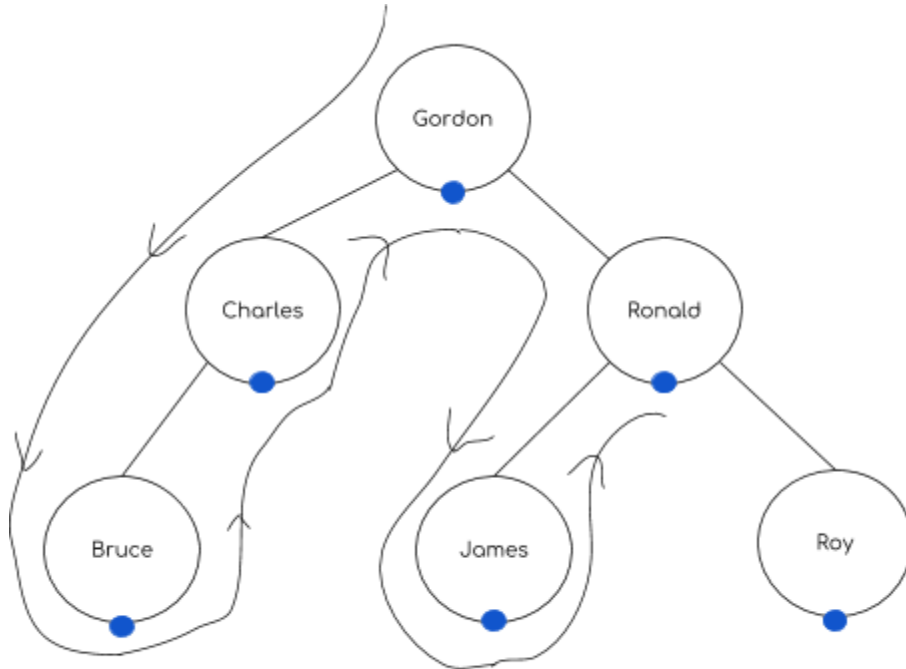
Gordon has been passed and outputted.



Output: Bruce, Charles, Gordon, James

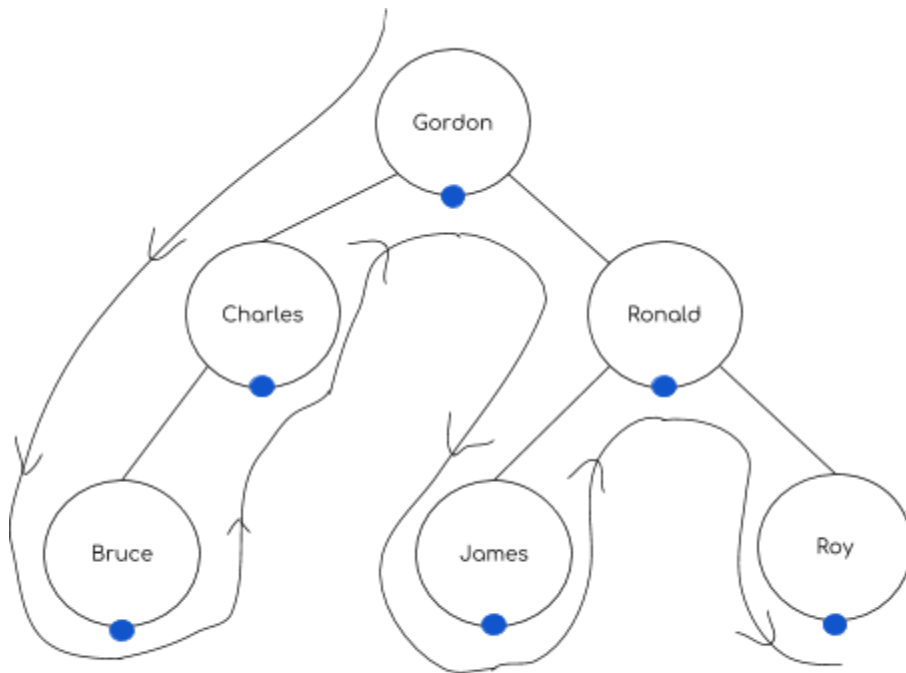
James has been passed and outputted.





Output: Bruce, Charles, Gordon, James, Ronald

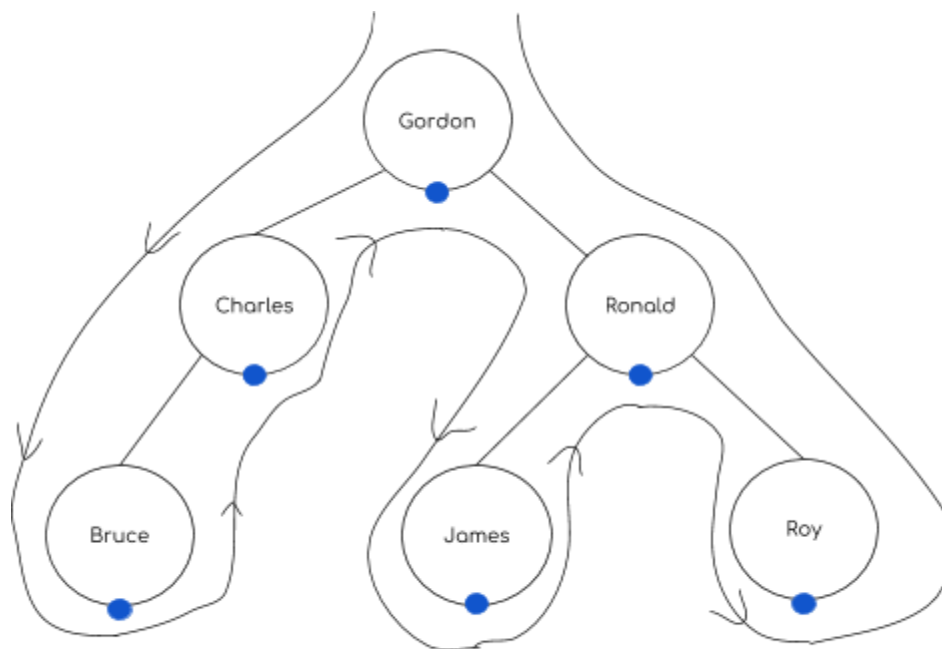
Ronald has been passed and outputted.



Output: Bruce, Charles, Gordon, James, Ronald, Roy

Roy has been passed and outputted.





Output: Bruce, Charles, Gordon, James, Ronald, Roy

The traversal has completed.

Synoptic Link

Reverse Polish Notation is a postfix way of writing expressions. RPN eliminates confusion over the order of execution and the need for brackets.

Reverse Polish Notation is covered in **Reverse Polish** under **Fundamentals of Algorithms**.

Post-Order Traversal

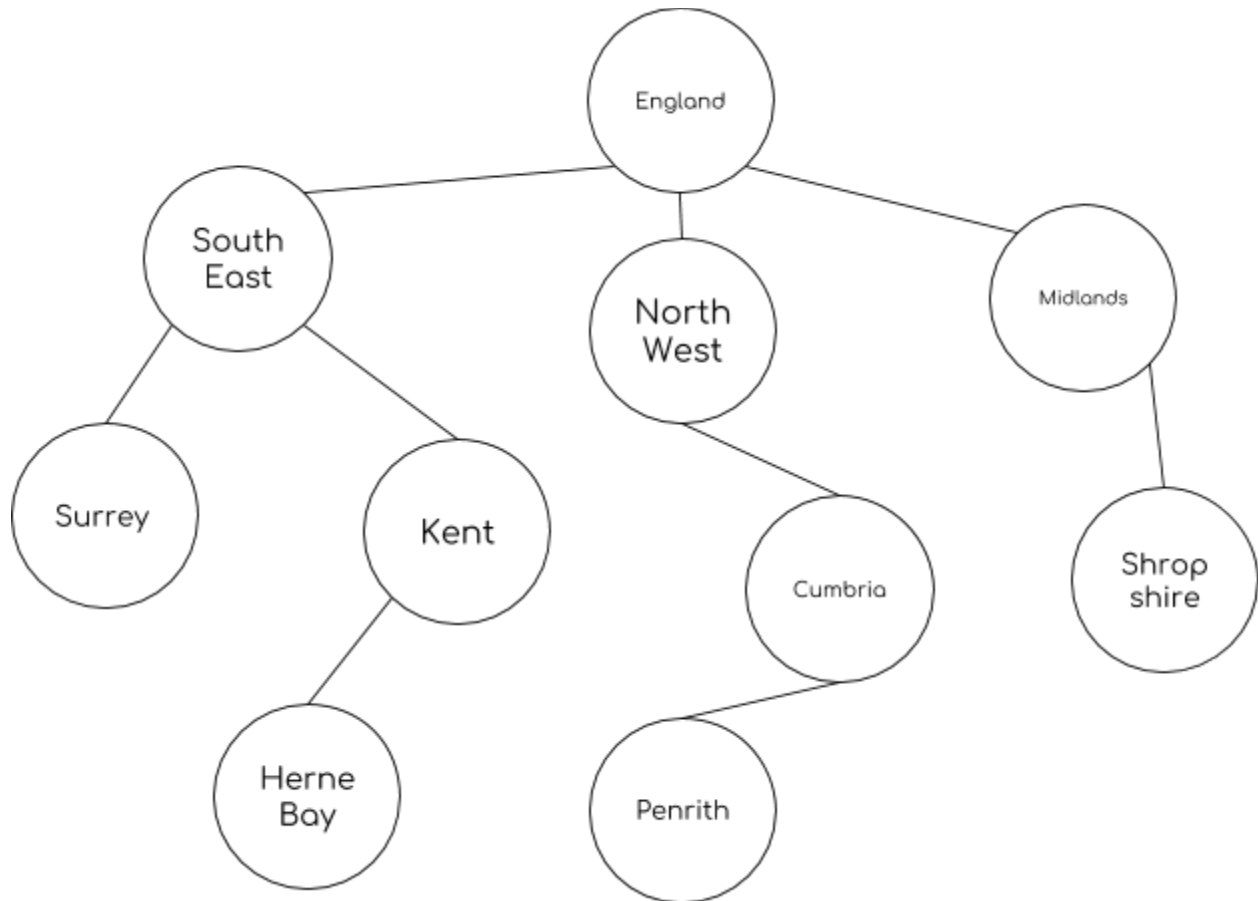
Post-order traversals can be performed on **any tree**. They are useful for **Infix to RPN (Reverse Polish Notation) conversions**, producing a postfix expression from an expression tree and emptying a tree.

Example:



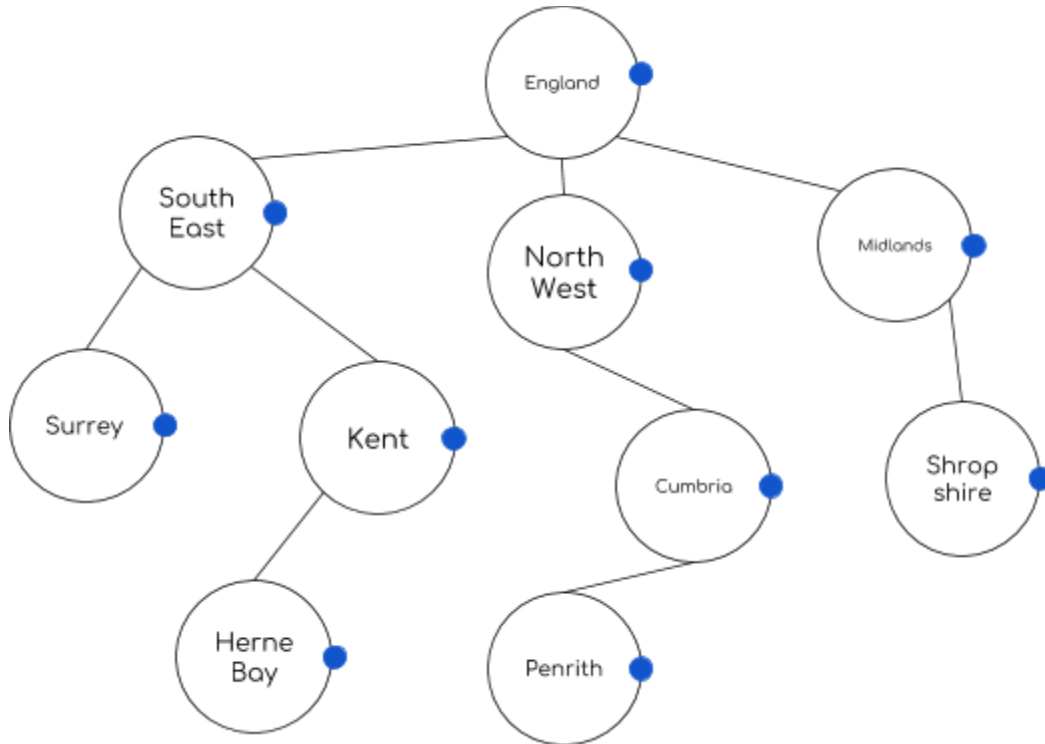


Here is a tree for some of the constituent parts of England.

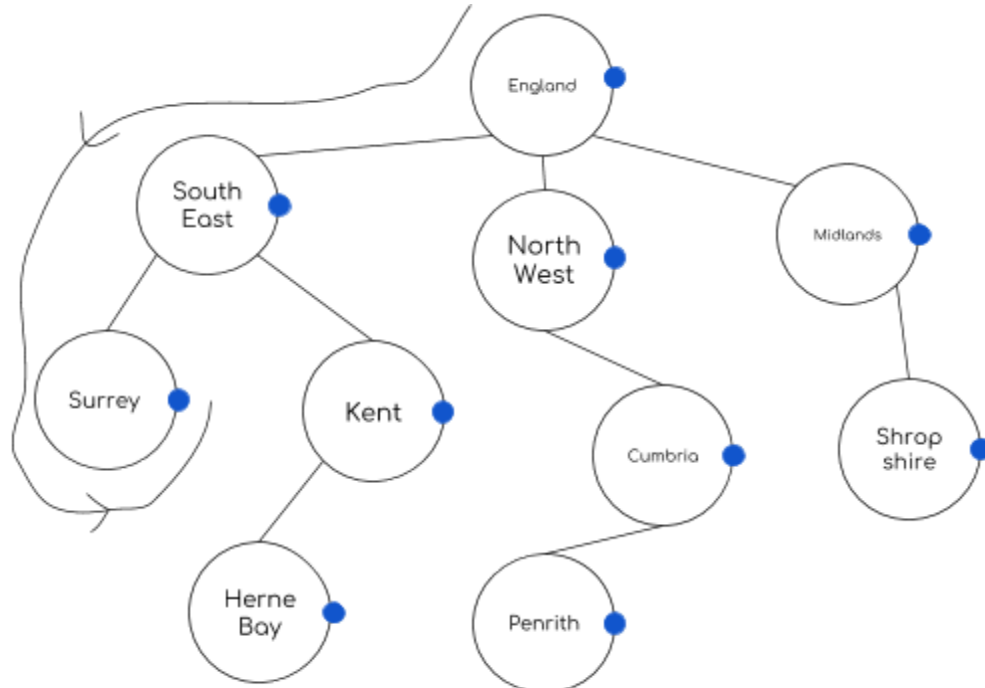


The first step in post-order traversal is to mark the right hand side of the nodes.





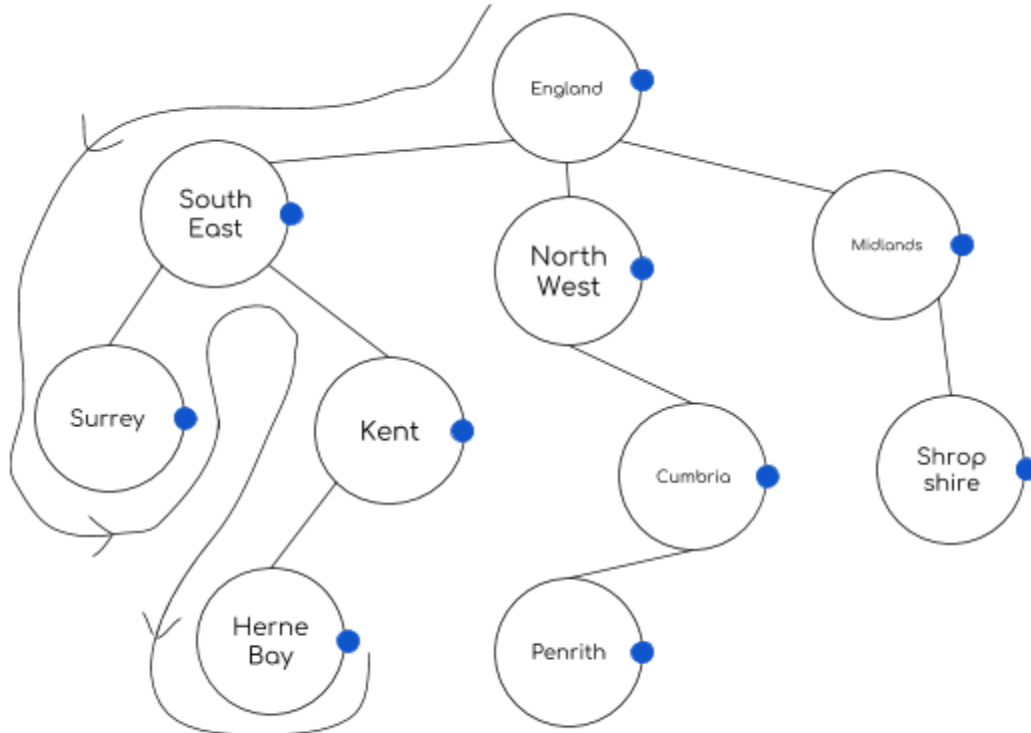
The traversal starts from the left and works its way around the nodes. As the traversal passes the blue dots the node is outputted.



Output: Surrey

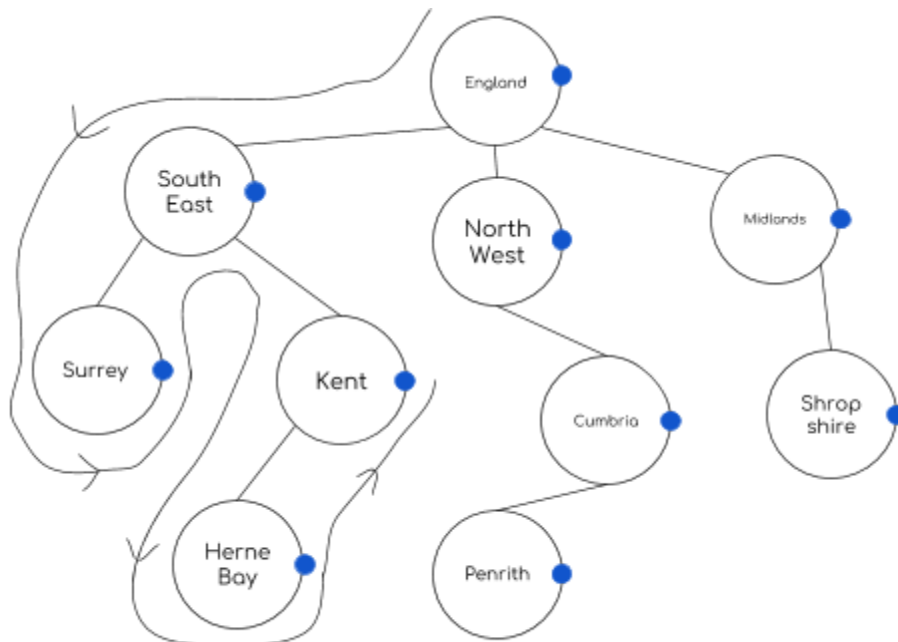
Surrey has been passed and outputted.





Output: Surrey, Herne Bay

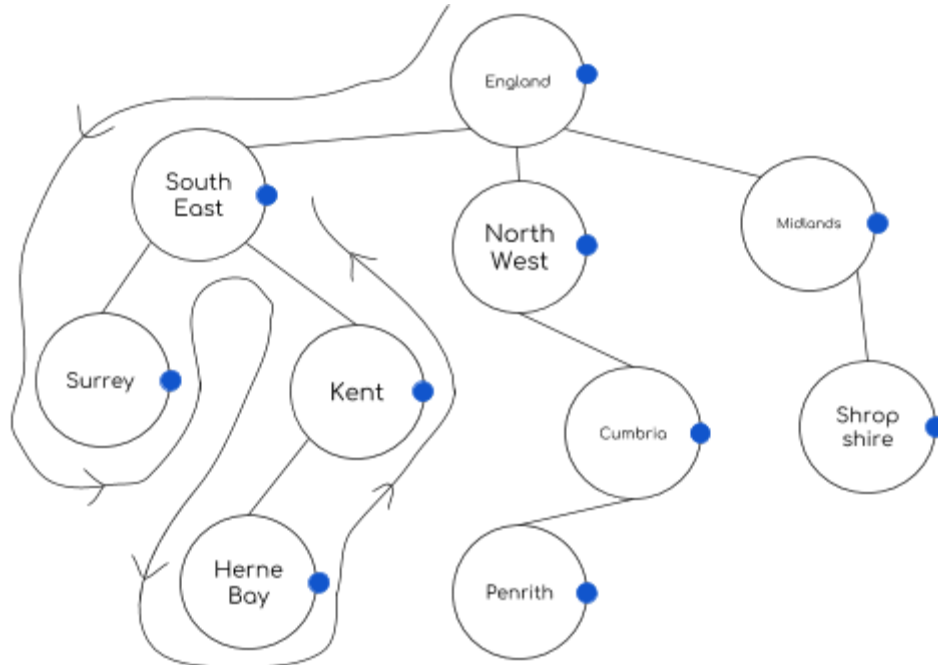
Herne Bay has been passed and outputted.



Output: Surrey, Herne Bay, Kent

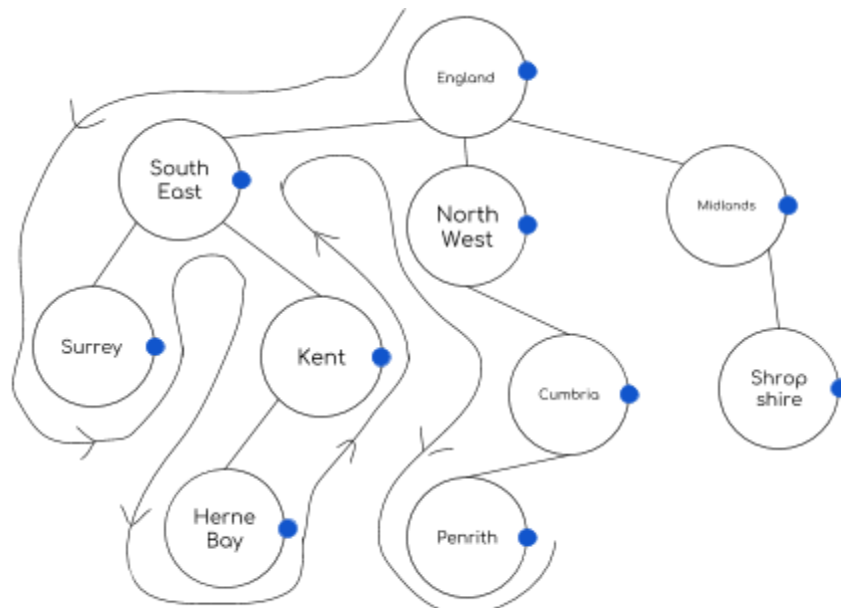
Kent has been passed and outputted.





Output: Surrey, Herne Bay, Kent, South East

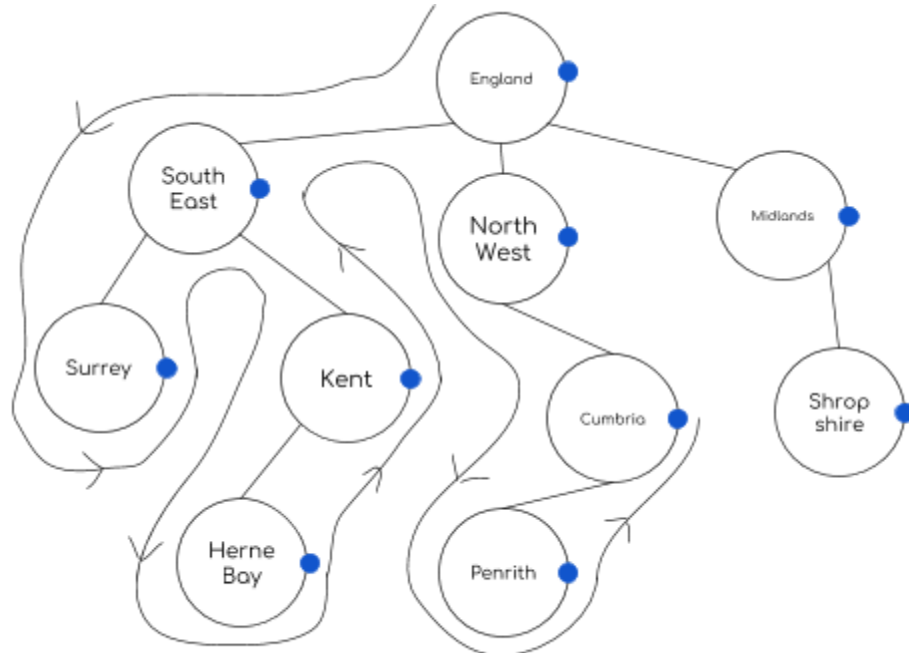
South East has been passed and outputted.



Output: Surrey, Herne Bay, Kent, South East, Penrith

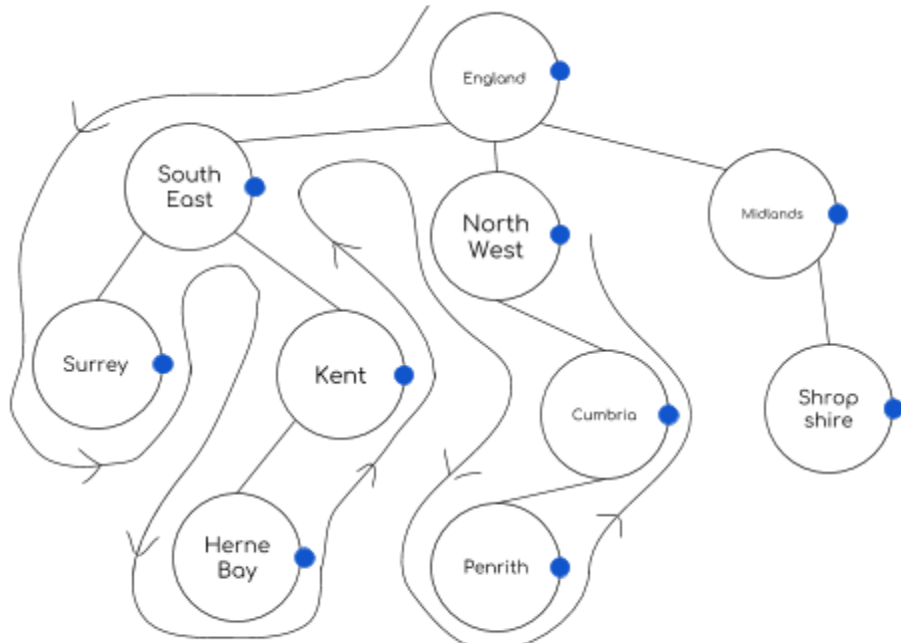
Penrith has been passed and outputted.





Output: Surrey, Herne Bay, Kent, South East, Penrith, Cumbria

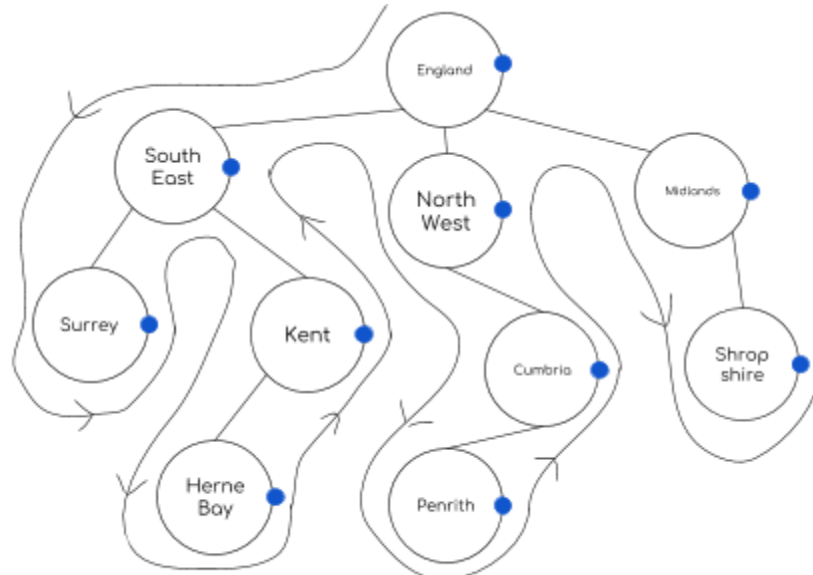
Cumbria has been passed and outputted.



Output: Surrey, Herne Bay, Kent, South East, Penrith, Cumbria, North West

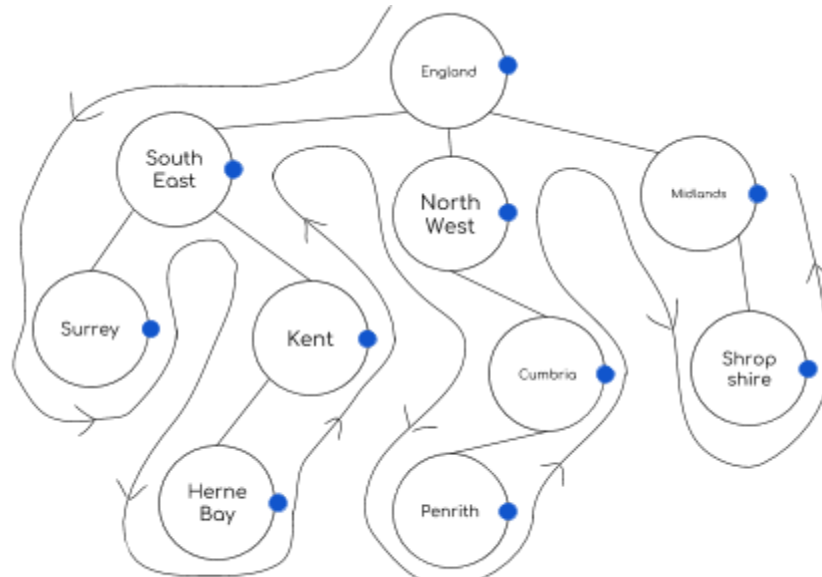
North West has been passed and outputted.





Output: Surrey, Herne Bay, Kent, South East, Penrith, Cumbria, North West, Shropshire

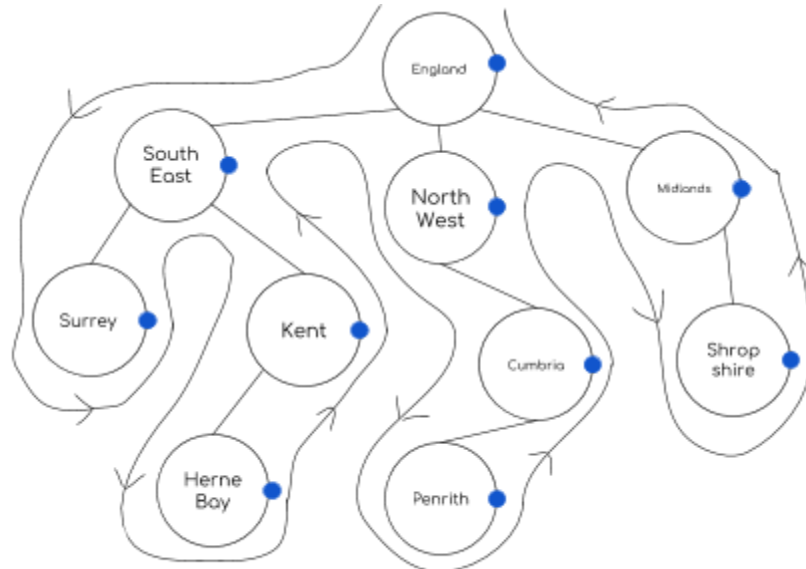
Shropshire has been passed and outputted.



Output: Surrey, Herne Bay, Kent, South East, Penrith, Cumbria, North West, Shropshire, Midlands

Midlands has been passed and outputted.





Output: Surrey, Herne Bay, Kent, South East, Penrith, Cumbria, North West, Shropshire, Midlands. England

England has been passed and outputted. The traversal has finished.

