

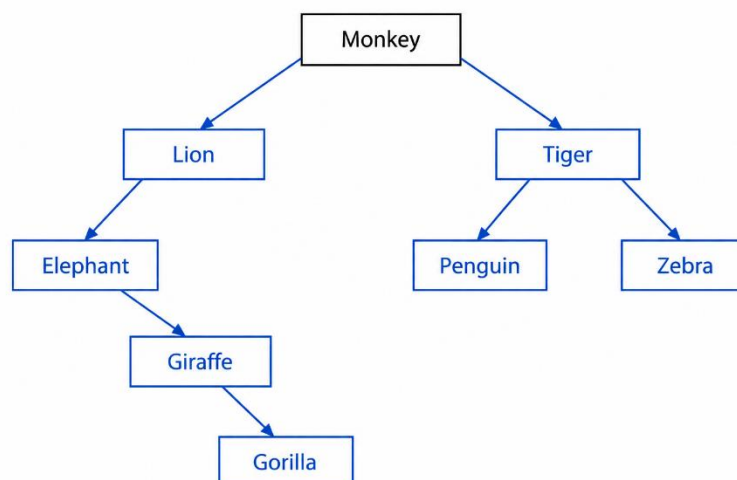


1.4.2 Graphs & Trees

Q1

A binary search tree is used to store the names of animals at the zoo.

Monkey is set as the root node, with Lion, Elephant, Giraffe, Tiger, Gorilla, Zebra and Penguin added one by one.



- a) Complete the tree diagram above to show where the animals would be added to this binary search tree.

[4 marks]

- b) Describe the process of using the binary search tree to search for the animal “Snake”.

[3 marks]

- Compare with Monkey – Snake is larger so go right
- Compare with Tiger – Snake is smaller so go left
- Compare with Penguin – Snake is larger but there is no right element so stop / output not found

The binary search tree is held in a 2-dimensional array called *animals* with 8 rows and 3 columns.

- c) Write a line of program code or pseudocode to declare the array *animals*.

[3 marks]

- `string[,] animals = new string[8, 3]`
Accept any suitable attempt at typing as an array





- d) In the 2-dimensional array *animals*:
- the first column contains a pointer to the left side
 - the second column contains the data
 - the third column contains a pointer to the right side.

Leaf nodes have the pointer *null*.

Complete the table showing the contents of the *animals* array to store the binary search tree.

	Left	Data	Right
0	1	Monkey	4
1	2	Lion	null
2	null	Elephant	3
3	null	Giraffe	6
4	7	Tiger	5
5	null	Zebra	null
6	null	Gorilla	null
7	null	Penguin	null

[4 marks]

- e) Compare the efficiency of a binary search tree to a hash table when searching for data.

[2 marks]

- Binary tree is less efficient than a hash table
- Hash table can find data immediately without checking other values

- f) Compare the efficiency of a binary search tree to a linked list when searching for data.

[2 marks]

- Binary tree is more efficient than a linked list
- Do not have to check every value / tree removes half of the values each time

