

Quiz 2

Due Mar 12 at 23:59

Points 10

Questions 10

Available Mar 5 at 0:00 - Mar 12 at 23:59

Time Limit None

Allowed Attempts Unlimited

Instructions

You can take this quiz as **many** times as you like before its due date.

Your mark will be based on your **last attempt**. All previous attempts will be ignored.

If you're an overseas student connecting via VPN (or just want the some extra assurance), download [Q2-orderedtree-fig.jpg \(https://canvas.sydney.edu.au/courses/48384/files/29695446/download?wrap=1\)](https://canvas.sydney.edu.au/courses/48384/files/29695446/download?wrap=1) [↓ \(https://canvas.sydney.edu.au/courses/48384/files/29695446/download?download_frd=1\)](https://canvas.sydney.edu.au/courses/48384/files/29695446/download?download_frd=1) before starting, in case it doesn't load properly in the quiz itself.

[Take the Quiz Again](#)

Attempt History

	Attempt	Time	Score
KEPT	Attempt 7	3 minutes	10 out of 10
LATEST	Attempt 7	3 minutes	10 out of 10
	Attempt 6	3 minutes	8 out of 10
	Attempt 5	11 minutes	9 out of 10
	Attempt 4	11 minutes	9 out of 10
	Attempt 3	5 minutes	7 out of 10
	Attempt 2	3 minutes	6 out of 10
	Attempt 1	5 minutes	5 out of 10

! Correct answers are hidden.

Score for this attempt: **10** out of 10

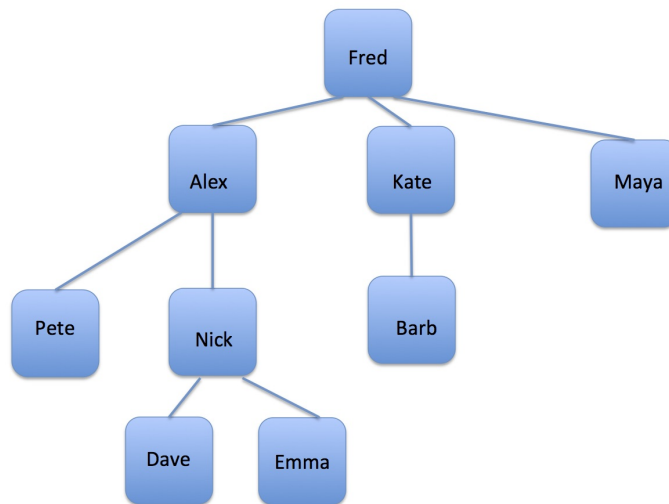
Submitted Mar 7 at 12:17

This attempt took 3 minutes.

Question 1

1 / 1 pts

Consider the ordered tree shown below. What is the depth of the node labeled Barb?



☐ 1

☐ 0

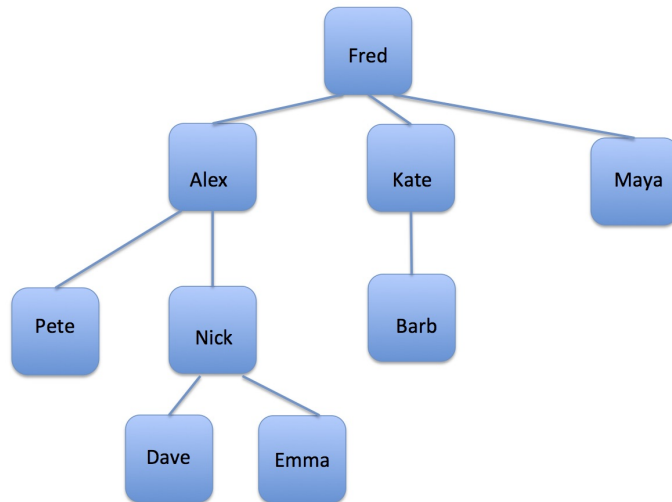
☒ 2

☐ 3

Question 2

1 / 1 pts

Consider the ordered tree shown below. What is the height of the tree?



☐ 1

☐ 4

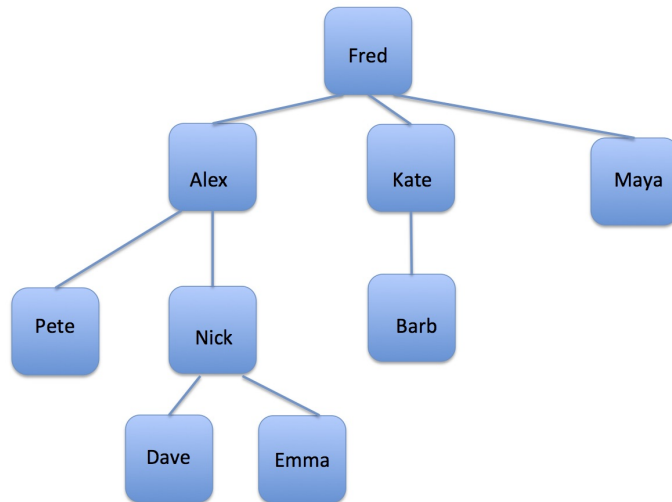
☐ 2

☒ 3

Question 3

1 / 1 pts

Consider the ordered tree shown below. What is the label on the parent of the node labelled Nick?



☒ Alex

☐ Dave

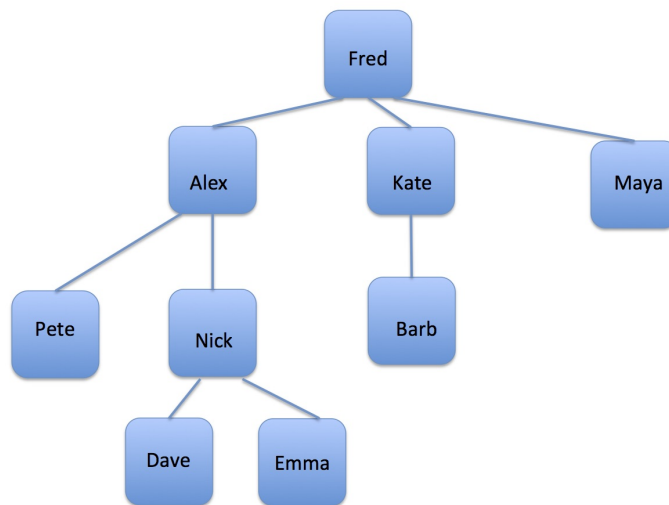
☐ Fred

☐ Pete

Question 4

1 / 1 pts

Consider the ordered tree shown below. What is the label of the first node in a post-order traversal of the tree?



☐ Maya

☐ Fred

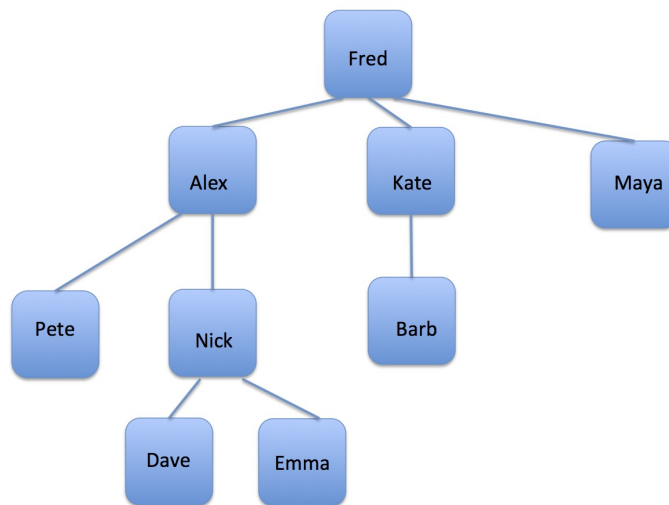
☐ Alex

☒ Pete

Question 5

1 / 1 pts

Consider the ordered tree shown below. What is the label of the second node in a post-order traversal of the tree?



☐ Alex

☐ Fred

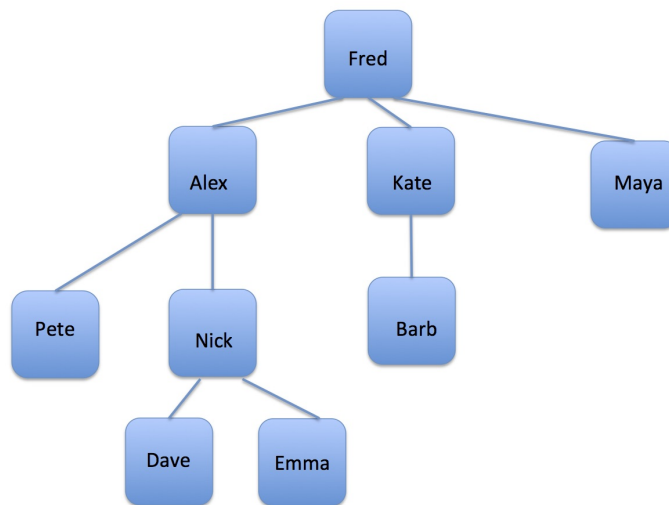
☒ Dave

☐ Kate

Question 6

1 / 1 pts

Consider the ordered tree shown below. What is the label of the last node in a post-order traversal of the tree?



☐ Maya

☐ Dave

☐ Emma

☒ Fred

Question 7

1 / 1 pts

Consider an initially empty tree T . Now perform n consecutive insert operations into T . What is the worst-case upper bound on the height of T after the insertions? Select the tightest bound that holds.

☐ $O(n^2)$

☐ $O(1)$

☐ $O(\log n)$

☒ $O(n)$

Question 8**1 / 1 pts**

Suppose we have a recursive function defined on a binary tree. The function takes as input a node in the tree, calls another function `bar` that performs some work on the node and then makes recursive calls to itself on the left and right subtrees provided they are not empty.

```
def foo(u):  
    bar(u)  
    if u.left is not empty then  
        foo(u.left)  
    if u.right is not empty then  
        foo(u.right)
```

Suppose `bar(u)` takes $O(|\text{subtree}(u)|)$ time to execute, where $|\text{subtree}(u)|$ is the size of the subtree rooted at `u`. What is the time complexity of running `foo` from the root a tree with n nodes. Select the tightest bound that holds.

☐ $O(1)$ ☒ $O(n^2)$ ☐ $O(n)$ ☐ $O(n^3)$ **Question 9****1 / 1 pts**

Suppose we have a recursive function defined on a binary tree. The function takes as input a node in the tree, calls another function `bar` on the node that performs some work and then makes recursive calls either to the left or the right subtree provided they are not empty.


```
def foo(u):  
    bar(u)  
    flip coin  
    if coin is heads and u.left is not empty then  
        foo(u.left)  
    if coin is tails and u.right is not empty then  
        foo(u.right)
```

Suppose $\text{bar}(u)$ takes $O(1)$ time to execute. What is the time complexity of running foo from the root a tree with n nodes.

- ☐ $O(n^2)$
- ☐ $O(\log n)$
- ☒ $O(n)$
- ☐ $O(1)$

Question 10

1 / 1 pts

Suppose we have a recursive function defined on a binary tree. The function takes as input a node in the tree, calls another function bar that performs some work on the node and then makes recursive calls to itself on the left and right subtrees provided they are not empty.

```
def foo(u):  
    bar(u)  
    if u.left is not empty then  
        foo(u.left)  
    if u.right is not empty then  
        foo(u.right)
```

Suppose $\text{bar}(u)$ takes $O(1)$ time to execute. What is the time complexity of running foo from the root of a tree with n nodes. Select the tightest bound that holds.

☒ $O(n)$

☐ $O(n^3)$

☐ $O(1)$

☐ $O(n^2)$

Quiz Score: **10** out of 10