X

swayam **(https://swayam.gov.in)**   **(https://swayam.gov.in/nc_details/NPTEL)**
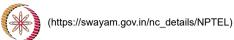
reviewer8@nptel.iitm.ac.in ⌄

NPTEL (https://swayam.gov.in/explorer?ncCode=NPTEL) » Programming, Data Structures And Algorithms Using Python (course)

☰

Course outline

How does an NPTEL online course work? ()

Week 1 : Introduction ()

Week 1 Quiz ()

Week 2: Basics of Python ()

Week 2 Quiz ()

Week 2 Programming Assignment ()

Week 3: Lists, inductive function

Week 7 Quiz

The due date for submitting this assignment has passed.

Due on 2022-03-16, 23:59 IST.

As per our records you have not submitted this assignment.

All questions carry equal weightage. All Python code is assumed to be executed using Python3. You may submit as many times as you like within the deadline. Your final submission will be graded.

Note:

- If the question asks about a value of type `string`, remember to enclose your answer in single or double quotes.
- If the question asks about a value of type `list`, remember to enclose your answer in square brackets and use commas to separate list items.

1) Given the following permutation of a,b,c,d,e,f,g,h,i,j, what is the **previous** permutation in lexicographic (dictionary) order? Write your answer without any blank spaces between letters.

    fjadchbegi

No, the answer is incorrect.
Score: 0

Feedback:
Invert the algorithm given in the video. Look for the longest suffix in increasing order, here `begi`*. The letter before is* `h`*. Replace by the next smallest letter in the suffix,* `g` *and arrange the remaining letters in descending order, to get* `giheb`*. So the final answer is* `fjadcgiheb`*.*

Accepted Answers:
(Type: Regex Match) [ ]*fjadcgiheb[ ]*
(Type: Regex Match) [ ]*\'fjadcgiheb\'[ ]*
(Type: Regex Match) [ ]*\"fjadcgiheb\"[ ]*

2.5 points

2) Assume we have defined a class Node that implements user defined lists of   *2.5 points*
numbers. Each object node of type Node has two attributes node.value and node.next with the
usual interpretation. We want to add a function sum() to the class Node which will compute the
sum of values in the list. An incomplete implementation of sum() given below. What should be
put in place of XXX and YYY?

```
def sum(self):
  if self.value == None:
    return(0)
  elif self.next == None:
    return(XXX)
  else:
    return(YYY)
```

    ○ Replace XXX by 1 and YYY by 1 + self.next.sum()

    ○ Replace XXX by 1 and YYY by self.value + self.next.sum()

    ○ Replace XXX by self.value and YYY by 1 + self.next.sum()

    ○ Replace XXX by self.value and YYY by self.value + self.next.sum()

No, the answer is incorrect.
Score: 0

Accepted Answers:
Replace XXX by self.value and YYY by self.value + self.next.sum()

3) Suppose we add this function `foo()` to the class `Tree` that implements search   *2.5 points*
trees. For a name mytree with a value of type `Tree`, what would `mytree.foo()` compute?

```
def foo(self):
        if self.isempty():
                return(0)
        elif self.isleaf():
                return(self.value)
        else:
                return(self.value + max(self.left.foo(),
                                        self.right.foo()))
```

    ○ The sum of the elements in the tree

    ○ The maximum sum across all root to leaf paths in the tree

    ○ The length of the longest root to leaf path in the tree

    ○ The number of root to leaf paths in the tree.

No, the answer is incorrect.
Score: 0

Feedback:
This computes the maximum sum along the paths from the root to the leaves.

Accepted Answers:
The maximum sum across all root to leaf paths in the tree

4) The preorder traversal of a binary search tree with integer values produces the   *2.5 points*
following sequence: 35, 23, 26, 46, 40, 39, 41, 52. What is the value of the right child of the root
of the tree?

○ 39

○ 40

○ 41

○ 46

No, the answer is incorrect.
Score: 0

Feedback:
The inorder traversal of a search tree is always the sorted sequence. In this case: 23, 26, 35, 39, 40, 41, 46, 52. From the preorder traversal, we know that 35 is the root of the tree, so the segment 23, 26 corresponds to the left subtree and the segment 39, 40, 41, 46, 52 corresponds to the right subtree. The preorder traversal of the right subtree starts with 46, so this is the right child of the root node.

Accepted Answers:
46