Polytechnic Tutoring Center

Exam 1 Review - CS 1134, Fall 2021

Disclaimer: This mock exam is only for practice. It was made by tutors in the Polytechnic Tutoring Center and is not representative of the actual exam given by the CS Department.

1 Write a recursive function that takes a list, first and last index of the list in as parameters and prints it in reverse (in place).

Sample Output:

```
lst = [1,2,3,4]
revPrint(lst, 0, 3)
>> 4 3 2 1
```

Code:

2 Circle the faster runtime:

O($nlog(n)$) or O($\sqrt{n}$)

O($\sqrt{n}$) or O($log(n)$)

O($nlog(n)$) or O($n^{1.25}$)

3 Let f(x,n) = $\sum_{i=1}^{n} \frac{x}{i} = \frac{x}{1} + \frac{x}{2} + \frac{x}{3}$ …, so f(4,3) = $\frac{4}{1} + \frac{4}{2} + \frac{4}{3} = 7.333$ Write a function for f. State your runtime.

Code:

4 What is the runtime of the following code snippets?

a
```
def function3(lst,low,high):
    if (low >= high):
        return 3
    for elem in lst:
        elem += 2
    return function3(lst,low+1,high-1)
```

b
```
def function2(lst):
    if (len(lst) == 1):
        lst[0] = 0
        return 2
```

```
        return function2(lst[:len(lst)//2])
```

c

```
def function1(lst,lst2):
    for elem in lst:
        if (elem in lst2):
            print('iteration')
```

5 If **A = [0,0,0,0,0,0]**, **B = [3,1,6,2]**, what does A and B look like after **function2(B)** and **function3(A,2,len(A)-1)** ? (Refer to Question 4)

6 Write a generator function that provides the values for a harmonic series of n elements. Hint: Harmonic series is 1, 1/2, 1/3…

Sample Output:

```
iters = 4
display_list = list(harmonic(iters))
display_list
>> [1.0,0.5,0.33,0.25]
```

Code:

7. Given a non-empty list with integers, write a function separate_num() to separate a list of even numbers and odd numbers and returns a list that contains all the odd numbers in the front and all even numbers in the back.

Example: an input list [3,15,44,2,51,89,20] to separate_num() will return [3,15,51,89,44,2,20]

Requirement: O (n) runtime and in place

(1) Do the implementation of separate_num( ) **iteratively**

(2) Do the implementation of separate_num( ) **recursively with helper function**