

CS2302 - Data Structures - Fall 2020

Exercise - Analyzing Recursive Functions

1. For each of the following functions, write a recurrence equation describing its running time with respect to n , the length of the input list a . Then find the O running time of the function by solving the recurrence.

```
# recursion_running_times.py
import numpy as np
```

```
def r0(a):
    if len(a)>0:
        print(a[0])
        r0(a[1:])
```

```
def r1(a):
    if len(a)>0:
        print(a)
        r1(a[1:])
```

```
def r2(a):
    if len(a)>0:
        print(a)
        mid = len(a)//2
        r2(a[:mid])
```

```
def r3(a):
    if len(a)>0:
        mid = len(a)//2
        print(a[mid])
        r3(a[:mid])
        r3(a[mid+1:])
```

```
def r4(a):
    if len(a)>0:
        mid = len(a)//2
        print(a)
        r4(a[:mid])
        r4(a[mid+1:])
```

```
def r5(a):
    if len(a)>0:
        mid = len(a)//2
        print(a[mid])
        r5(a[:mid])
```

```
def r6(a):
    if len(a)>0:
        print(a[-1])
        r6(a[:-1])
        r6(a[:-1])
```

```
def r7(a):
    if len(a)>0:
        mid = len(a)//2
        print(a[mid])
        for i in range(4):
            r7(a[:mid])
```