Problem Set 6

Anthony Tam, CSC236 11/04/2016

Question 1: Analyze the given binary search.

ANSWER: Part a)

$$T(n) = \begin{cases} \mathbf{a} & \mathbf{n = 0} \\ \mathbf{a} & \mathbf{n = 1} \\ \mathbf{T(\frac{n}{2}) + c} & \textit{Otherwise} \end{cases}$$

$T(n) = T(\frac{n}{2}) + c$ $a = 1$

$log_2 1 = 0$ $b = 2$

$k = log_2 1$ $k = 0$

$\therefore \mathcal{O}(n^0\ log(n))$ or $\mathcal{O}(log(n))$

ANSWER: Part b)

Precondition: A list of objects with equals, greater then, and less then defined. The list can be empty

Postcondition: An integer value within the range of the list, or -1

Proof: Path One (Lines 6&7

And empty list is given, and -1 is returned.

$\therefore$ Path one passes.

Path Two (Lines 9&10

The midpoint of the list is what we are looking for, this index is returned.

$\therefore$ Path two passes.

Path Three (Lines 11&12)

The midpoint of the list is larger then what we are looking for. We can divide the list in half and check the left half of the list for the value we are looking for. Eventually this will cause a return from one of the previous paths which we already proved to be true.

$\therefore$ Path three passes.

Path Four (Lines 13-18)

The value we are looking for must be on the right of of the list. We will split it here and check for the result. If the value is not found on this half of the list, return -1. If the value is found, will will take that index (The index of half the list) and add it to the size of the other half. We also must account for both lists starting at index 0, so we must add 1 more. This will be an integer within the bounds of the array.

$\therefore$ Path four passes.

Given all four paths pass the pre and post conditions, the program is correct.

Question Two: Analyze the given mystery function

ANSWER: Part a)

$$T(n) = \begin{cases} \text{a} & \text{n} \leq 2 \\ 3\text{T}(\frac{2n}{3}) + \text{d} & \text{n} \geq 3 \end{cases}$$

$T(n) = 3T(\frac{2n}{3}) + d$	$a = 3$
$log_2 3 \approx 1.585$	$b = 3/2$
$k < log_2 3$	$k = 0$

This means case 2 of the master theorem applies.

$\therefore \mathcal{O}(n^{log_2 3})$

Part b)

Precondition: A list of objects which have equals, less then, and greater then defined. The list can be zero.

Postcondition: The list of objects closer to a low to high order.

Part c)

Path One (Lines 2&3)

The list is of one of less elements, so it is already sorted.

$\therefore$ Path one passes.

Path Two (Lines 4&5)

The first element in the list is larger then the last element in the list. We can swap these entries and have a list which is closer to being sorted.

Path two passes.

Path Three (Lines 6-10)

The length of the list is split into 3, the list is then sorted 3 times. First the program sorts the first 2 thirds of the list, then the last third. It then goes back to sort the first 2 thirds to check for changes. After these sorts, the list is then sorted.

$\therefore$ Path three passes.

Given all three paths pass the pre and post conditions, the program is correct.