CS2500 Homework 2

Evan Wilcox

Due Feburary 7, 2019

1. **2.2-2**

```
selectionSort(A)
for i = 1 to A.length-1
    min = i

    for j = i+1 to A.length
        if A[j] < A[min]
            min = j

    swap A[i] with A[min]

return A
```

The subarray $A[1...i-1]$ consists of the smallest elements in sorted order. After the first $n-1$ elements, the subarray A[1...n-1] contains the smallest $n-1$ elements so the nth element is the largest element. The running time of the algorithm is $\Theta(n^2)$ for all cases.

2.2-3
Average case would be $\Theta(n)$ because the average search time is $\frac{n}{2}$. Worst case would be $\Theta(n)$ because the worst case search time is n, or when n is not found.

2. **2.3-4**

$$T(n) = \begin{cases} \Theta(1) & \text{if } n \leq 1, \\ T(n-1) + D(n) + C(n) & \text{otherwise.} \end{cases}$$

Where $D(n)$ is the time taken to divide the problem and $C(n)$ is the time taken to combine the sub problems.

2.3-5

```
binarySearch(v, A):
    if A.length == 0:
        return -1
    l = 1
    m = A.length/2
    h = A.length
    while(l < m)
        if v < A[m]
            h = m
        else
            l = m
        m = (h+l)/2
    if A[m] != v
        return -1
    return m
```

The algorithm splits the range in half based on the comparison of the middle element to v. The recurrence for this is $T(n) = T(n/2) + \Theta(1)$, whose solution is $T(n) = \Theta(\lg n)$.

3. **2-2**

   d) Bubblesort's worst-case running time is $\Theta(n^2)$ which is the same as insertion sort.

4. **3-1**

   a) $0 \leq p(n) \leq n^k$ for all $n \geq n_0$

   b)

   c)

   d)

   e)

5. **3-2**

	A	B	O	o	Ω	ω	Θ
a.	$\lg^k n$	$n^{\in}$					
b.	n^k	c^n					
d.	2^n	$2^{n/2}$					
e.	$n^{\lg c}$	$c^{\lg n}$					
f.	$\lg(n!)$	$\lg(n^n)$					

6. a) $5n^2 - 6n = \Theta(n^2)$
There exist positive constants c_1, c_2, and n_0 such that

$$0 \leq c_1 n^2 \leq 5n^2 - 6n \leq c_2 n^2 \text{ for all } n \geq n_0$$

Simplified,

$$0 \leq c_1 \leq 5 - \frac{6}{n} \leq c_2$$

With constants $c_1 = 2, c_2 = 8, n = 12$,

$$0 \leq 2 \leq 5 - \frac{6}{12} \leq 8$$

$$0 \leq 2 \leq 4\frac{1}{2} \leq 8$$

b) $n^3 + 10^6 n^2 = \Theta(n^3)$
There exist positive constants c_1, c_2, and n_0 such that

$$0 \leq c_1 n^3 \leq n^3 + 10^6 n^2 \leq c_2 n^3 \text{ for all } n \geq n_0$$

Simplified,

$$0 \leq c_1 \leq 1 + \frac{10^6}{n} \leq c_2$$

With constants $c_1 = 2, c_2 = 2, n = 10^6$,

$$0 \leq 2 \leq 1 + \frac{10^6}{10^6} \leq 2$$

$$0 \leq 2 \leq 2 \leq 2$$

c) $6(2^n) + n^2 = O(2^n)$
There exist positive constants c and n_0 such that

$$0 \leq 6(2^n) + n^2 \leq c2^n \text{ for all } n \geq n_0$$

Simplified,

$$0 \leq 6 + \frac{n^2}{2^n} \leq c$$

With constants $c = 8, n = 4$,

$$0 \leq 6 + \frac{4^2}{2^4} \leq 8$$

$$0 \leq 7 \leq 8$$

7. a) $10n^2 + 9 \neq \Theta(n)$ There exist positive constants c_1, c_2, and n_0 such that

$$0 \leq c_1 n \leq 10n^2 + 9 \leq c_2 n \text{ for all } n \geq n_0$$

Simplified,

$$0 \leq c_1 \leq 10n + \frac{9}{n} \leq c_2$$

There is no value for constant c_2 large enough to always be greater than $10n + \frac{9}{n}$ for all $n \geq n_0$.

b) $n^2 lgn \neq \Theta(n^2)$
There exist positive constants c_1, c_2, and n_0 such that

$$0 \leq c_1 n^2 \leq n^2 lgn \leq c_2 n^2 \text{ for all } n \geq n_0$$

Simplified,

$$0 \leq c_1 \leq lgn \leq c_2$$

There is no value for constant c_2 large enough to always be greater than lgn for all $n \geq n_0$.

8. $\sum_{i=0}^{n} 2^i$ for $n = 31$ equals 2,147,483,648.

9. $\log_2 1024 = 10$

10. $a^{\log_b c} = b^{(\log_b a)(\log_b c)} = (b^{\log_b (c)})^{\log_b (a)} = c^{\log_b a}$