Homework 3

3.1

a) $\lim\limits_{n\to\infty} \frac{9n}{5n^3} = 0$ (higher rank rule) $\Rightarrow f\in o(g), f\in O(g)$

$\lim\limits_{n\to\infty} \frac{5n^3}{9n} = \infty \Rightarrow g\in \omega(f), g\in\Omega(f)$

b) $\lim\limits_{n\to\infty} \frac{9\cdot n^{0,8}+2n^{0,3}+14\log n}{\sqrt{n}} = \infty \; (0,8 > 0,5 (\frac{1}{2})) \Rightarrow f\in\omega(g), f\in\Omega(g$

$\lim\limits_{n\to\infty} \frac{\sqrt{n}}{9\cdot n^{0,8}+2n^{0,3}+14\log n} = 0 \Rightarrow g\in o(f), g\in O(f)$

$\cancel{c)\ \lim\limits_{n\to\infty} \frac{n^2}{\log n} \frac{\log^1 n}{\log n\cdot n} = \infty \Rightarrow f\in\omega(g), f\in\Omega(g)}$

$\cancel{\lim\limits_{n\to\infty}}$

c) $\lim\limits_{n\to\infty} \frac{\frac{n^2}{\log n}}{n\log n} = \lim\limits_{n\to\infty} \frac{n^3 \cancel{\log n}}{\cancel{\log n}} = \infty$

$\Rightarrow f\in\omega(g), f\in\Omega(g)$

$\lim\limits_{n\to\infty} \frac{n\log n}{\frac{n^2}{\log n}} = 0 \Rightarrow g\in O(f), g\in o(f)$

d) $\lim_{n \to \infty} \frac{(\log(3n))^3}{9 \log n} = \lim_{n \to \infty} \frac{\ln^3(3n)}{\ln n^9} = \infty$

$\Rightarrow f \in \Omega(g), f \in \omega(g)$

$\lim_{n \to \infty} \frac{\ln(n^9)}{\ln^3(3n)} = 0 \Rightarrow g \in o(f), g \in O(f)$

3.2 b) In order for the program to be correct, the loop invariant must hold true before, during and after ~~the loop.~~ each iteration. The loop invariant for selection sort is that all the elements until the current index i will contain the first i smallest elements of the array

Step I: prior to iteration: during this step, ~~there~~ the index is 0. Therefore, any 0 elements are sorted and the loop invariant holds

Step II: during the iteration: The loop invariant states that for an index i, all elements $A[0, \dots i-1]$ are the ~~first~~ i smallest numbers. During the iteration, i increases to $i+1$. Ergo, the ~~first~~ elements $A[0 \dots i]$ are the ~~first~~ $i+1$ smallest elements. This is proven by induction. Therefore, the loop invariant holds during this step.

Step III: after iteration: At the end of the loop, selection sort returns a sorted array. A sorted array can be described as the smallest n elements in order, ~~of~~ a the array having n elements. Therefore, the loop invariant holds.

Because the loop invariant holds through all these steps, the program can be deemed correct.

e) If we take a look at the structure of the algorithm, we can see it takes $n-i$ steps, $n-1$ times. This can be seen as:

$$\sum_{i=1}^{n-1} n-i = n-1+n-2+n-3+\dots n-(n-1)$$

$$= \frac{\cancel{1}(n^2-n)}{2}$$

$$= \frac{n^2-n}{2} \Rightarrow \boxed{\text{The complexity is } \Theta(n^2)}$$