Homework 2

1a $5n^3 + 2n^2 + 3n = O(n^3)$

$f(n) \le c \cdot g(n)$

$5n^3 + 2n^2 + 3n \le c \cdot n^3$

$5n^3 + 2n^2 + 3n \le (5+2+3)n^3 = cn^3$, for $c = 10$ when $n \ge n_0 = 1$

1b. $\sqrt{7n^2+2n-8} = \Theta(n)$

$\sqrt{7n^2+2n-8} = O(n)$

$\sqrt{7n^2+2n-8} \le c \cdot n$

$(7n^2+2n-8)^{1/2} \le c \cdot n$

$\sqrt{7}n + \sqrt{2}n^{1/2} - \sqrt{8} \le c \cdot n$ $\quad$ $\sqrt{7}n + \sqrt{2}n^{1/2} - \sqrt{8} \le (\sqrt{7}+\sqrt{2})n = cn$ $\quad c = \sqrt{7}+\sqrt{2}$ when $n \ge n_0 = 1$

$\sqrt{7n^2+2n-8} = \Omega(n)$

$\sqrt{7n^2+2n-8} \ge c \cdot n$

$\sqrt{2}n^{1/2} - \sqrt{8} \ge 0$

$n^{1/2} \ge 2$ $\quad$ $n \ge 4$

$\sqrt{7n^2+2n-8} \ge c \cdot n$ for $c = \sqrt{7}$

$\sqrt{7}\,n \le \sqrt{7n^2+2n-8} \le (\sqrt{7}+\sqrt{2})n$ for $n \ge 4$

1c $d(n) = O(f(n))$ $\quad$ $e(n) = O(g(n))$ $\quad$ $d(n)e(n) = O(f(n)g(n))$

$d(n) \le c \cdot f(n)$ $n \ge n_0$ $\quad$ $e(n) \le c \cdot g(n)$ $n \ge m_0$ $\quad$ $d(n)e(n) \le f(n)g(n) \cdot c$

$e(n) \cdot d(n)$

$\hookrightarrow c_1 \cdot c_2 \cdot f(n) \cdot (g(n))$ $\quad$ $n \ge m_0$ or $n \ge n_0$ $\quad$ n will always be greater or equal to the intial values therefore

$(d(n))(e(n)) \le f(n)g(n) \cdot c$

Question 2:

```
def example1(lst):                    Θ(n²)
Θ(1) [ n = len(lst)                   ] Θ(1)
     [ total = 0                      ] Θ(1)
Θ(n²)[ for j in range(n):
     Θ(j)[ for k in range(1+j):
     [         total += lst[k]        ] Θ(1)
Θ(1) [ return total                   ] Θ(1)
```

```
def example2(lst):                    Θ(n)
Θ(1) [ n = len(lst)                   ] Θ(1)
     [ prefix = 0                     ] Θ(1)
     [ total = 0                      ] Θ(1)
Θ(n) [ for j in range(n)
     [     prefix += lst[j]           ] Θ(1)
     [     total += prefix            ] Θ(1)
Θ(1) [ return total                   ] Θ(1)
```

```
def example3(n):                      Θ(log(n))
Θ(1) [ i = 1                          ] Θ(1)
     [ sum = 0                        ] Θ(1)
Θ(log(n²)) [ while (i < n*n):
     [     i *= 2                     ] Θ(1)
     [     sum += i                   ] Θ(1)
Θ(1) [ return sum                     ] Θ(1)
```

```
def example4(n):                      Θ(n log(n))
Θ(1) [ i = n                          ] Θ(1)
     [ sum = 0                        ] Θ(1)
Θ(n log n) [ while (i > 1):
     Θ(n)[ for j in range(i):
     [         sum += i*j             ] Θ(1)
     [     i //= 2
Θ(1) [ return sum
```