HW2:

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

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

$5n^3+2n^2+3n \le 7n^3$

$c = 7$

$g(n) = n^3$

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

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

$\sqrt{7}\,n \le \sqrt{7n^2+2n-8} \le 3n$

$c_1 = \sqrt{7}$ $c_2 = 3$

$g(n) = n$

c) $d(n) = O(f(n))$ $e(n) = O(g(n))$

$d(n) \le c_1 \cdot f(n)$ $e(n) \le c_2 \cdot g(n)$

$d(n)\,e(n) \le c_1 c_2 f(n) g(n)$

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

$d(n)\,e(n) = O(f(n) g(n))$

(2) a)

```
def example1(lst):
 """Return the sum of the prefix sums of sequence S."""
      n = len(lst)
      total = 0
      for j in range(n):
           for k in range(1+j):
                total += lst[k]
      return total
```

n n

$\theta(n^2)$

b)

```
def example2(lst):
"""Return the sum of the prefix sums of sequence S."""
      n = len(lst)
      prefix = 0
      total = 0
      for j in range(n):
            prefix += lst[j]
            total += prefix
      return total
```

] n

$\theta(n)$

c)

```
def example3(n):
      i = 1
      sum = 0
      while (i < n*n):
            i *= 2
            sum += i
      return sum
```

n^2

] $\log_2(n^2) = 2\log n$

$\theta(\log n)$

d)

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

] n

] $\log(n)$

$\theta(n \log n)$