Our proofs of these theorems are given at the end of this document.! There are other ways to prove these theorems; the constants c and N that we come up with are not the only possibilities. Wee prefer out proof format for two reasons: (1) with each step, it explains why the step is logical. (2) Each proof has a natural progression, changing f(n) into c g(n).

**Definition**. Function f(n) ∈ O(g(n)) iff there exist positive constants c and N such that for all n, n ≥ N, f(n) ≤ c \* g(n).

**Theorem 1**. 20n ∈ O(n).

**Theorem 2**. n ∈ O(20n).

**Theorem 3**. n + 50 ∈ O(n).

**Theorem 4**. log n ∈ O(n).

**Theorem 5**. If f(n) ∈ O(n) and h(n) ∈ O(n) then f(n) + g(n) ∈ O(n).  
That is, if two functions are in O(n), so is their sum.

**Theorem 6**. n + log n ∈ O(n).

**Theorem 7**. If f(n) ∈ O(g(n)) and h(n) ∈ O(g(n)) then f(n) + g(n) ∈ O(g(n)).

**Theorem 8**. 5n2 + 50n ∈ O(n2).

**Theorem 9**. n (n–1)/2 ∈ O(n2).

**Theorem 10**. 80 2n + 60n3 ∈ O(2n).

**Theorem 1**. 20n ∈ O(n). We change 20n into c n, finding N and c as we go.

20n

= <Choose N = 1 and c = 20>

c n for n ≥ N

**Theorem 2**. n ∈ O(20n). We change n into c(20n), finding N and c as we go.

n

< <Choose N = 1. For all n ≥ 1, n < 2n>

20 n for n ≥ N

< <Choose c = 1>

c (20n) for n ≥ N

**Theorem 3**. n + 50 ∈ O(n). We change n + 50 into c(20n), finding N and c as we go.

n + 50

≤ <Choose N = 50. For all n ≥ 50, 50 ≤ n>

n + n for n ≥ N

< <Arithmetic, and choose c = 2>

c n for n ≥ N

**Theorem 4**. log n ∈ O(n). We change log n into c n, finding N and c as we go.

log n

≤ <Choose N = 1. If n = 2y, log n = y. From this, we know that for n ≥ 1, log n ≤ n>

n for n ≥ N

< <Choose c = 1>

c n for n ≥ N

**Theorem 5**. If f(n) ∈ O(n) and h(n) ∈ O(n) then f(n) + h(n) ∈ O(n).

**Proof**. Since f(n) ∈ O(n), there exist positive N1 and c1 such that f(n) ≤ c1 n for all n ≥ N1.   
Since h(n) ∈ O(n), there exist positive N2 and c2 such that h(n) ≤ c2 n for all n ≥ N2. We calculate:

f(n) + h(n)

≤ <Above-mentioned fact about f(n) ∈ O(n)>

c1 n + h(n) for n ≥ N1

≤ <Above-mentioned fact about h(n) ∈ O(n)>

c1 n + c2 n for n ≥ N1 and n ≥ N2

≤ <Choose N = max(N1, N2) >

c1 n + c2 n for n ≥ N

≤ <Arithmetic, and choose c = c1 + c2 >

c n for n ≥ N

**Theorem 6**. n + log n ∈ O(n).  
**Proof**. You can prove this as we did most of the previous ones. However, you can also apply theorem 5, since by theorems 1 and 2, n ∈ O(n), and by theorem 4, log n ∈ O(n).

**Theorem 7**. If f(n) ∈ O(g(n)) and h(n) ∈ O(g(n)) then f(n) + g(n) ∈ O(g(n)).  
**Proof**. This proof will be exactly the same as the proof of Theorem 5, with all occurrences of O(n) replaced by O(g(n)).

**Theorem 8**. 5n2 + 50n ∈ O(n2). We change 5n2 + 50n into c n2, finding N and c as we go.

5n2 + 50n

≤ <Choose N = 1, because 50 n <= 50n2 for n ≥ 1>

5n2 + 50n2  for n ≥ N

≤ <Arithmetic>

55n2 for n ≥ N

≤ <Choose c = 55>

cn2 for n ≥ N

**Theorem 9**. n(n–1)/2 ∈ O(n2). We change 5n2 + 50n into c n2, finding N and c as we go.

n(n­–1)/2

≤ <Arithmetic>

n2/2 – n/2

≤ <Arithmetic>

n2

≤ <Choose N = 1 and N = 1>

cn2 for n ≥ N

**Theorem 10**. 80 2n + 60n3 ∈ O(2n). We change 80 2n + 60n3 into c 2n, finding N and c as we go. .

80 2n + 60n3

≤ <For n = 10, n3 = 1000 < 1024 = 2n. For n > 10, 2n > n3. Choose N = 10.>

80 2n + 60 2n for n ≥ N

≤ <Arithmetic>

140 2n for n ≥ N

≤ <Choose c = 140>

c 2n for n ≥ N

.