COMP 250 Assignment 3

Christopher Hundt (110220945)

November 4, 2002

Question 1

The worst-case running time of all three of these algorithms is $\Theta(n)$, because the size of the set dealt with decreases by 1 with each recursive call, and all three algorithms end when the set dealt with becomes empty.

Insert

```
insert (a, S)
  if S is empty then return a.()
  else if a < first(S) then return a.S
  else if a > first(S) then return first(S).insert(a, rest(S))
  else return S
```

Delete

```
delete (a, S)
  if S is empty then return ()
  else if a < first(S) then return S
  else if a > first(S) then return first(S).delete(a, rest(S))
  else return rest(S)
```

Member

```
member (a, S)
  if S is empty then return false
  else if a < first(S) then return false
  else if a > first(S) then return member(a, rest(S))
  else return true
```

Question 2

Union

```
union(S1, S2)
  if S2 is empty then return S1
  else if S1 is empty then return S2
  else if first(S1) < first(S2) then return first(S1).union(rest(S1), S2)
  else if first(S1) > first(S2) then return first(S2).union(S1, rest(S2))
  else return first(S1).union(rest(S1), rest(S2))
```

In this algorithm, the size of at least one of the sets dealt with decreases by one with each recursive call, and recursive calls end when either set size becomes 0. Thus, the running time is no more than $n + m \leq 2n$, where n is the size of the larger set and m is the size of the smaller set. $2n \in \Theta(n)$, so the worst-case running time is $\Theta(n)$.

Intersection

```
intersection (S1, S2)
  if S2 is empty then return ()
  else if S1 is empty then return ()
  else if first(S1) < first(S2) then return intersection(rest(S1), S2)
  else if first(S1) > first(S2) then return intersection(S1, rest(S2))
  else return first(S1).intersection(rest(S1), rest(S2))
```

The running time of this algorithm is $\Theta(n)$, for the same reason that the running time of `union` is $\Theta(n)$.

Question 3

1. (a) $2^n \in O(3^n)$
   Show: $\exists c, n_0 \forall n \geq n_0, 2^n \leq c3^n$
   Pick: $c = 1$, $n_0 = 1$

$$\begin{array}{rcll} 2 & \leq & 3 & \\ \log_{10} 2 & \leq & \log_{10} 3 & \\ n \log_{10} 2 & \leq & n \log_{10} 3 & \text{(valid for } n \neq 0 < 1) \\ log_{10} 2^n & \leq & log_{10} 3^n & \\ 2^n & \leq & 3^n & \end{array}$$

   (b) $3^n \notin O(2^n)$
   Show: $\forall c, n_0 \exists n \geq n_0, 3^n > c2^n$
   If $c \geq 1$, pick: $n > \frac{\log_{10} c}{\log_{10} 3 - \log_{10} 2}$

$$\begin{array}{rcl} n & > & \dfrac{\log_{10} c}{\log_{10} 3 - \log_{10} 2} \\ n(\log_{10} 3 - \log_{10} 2) & > & \log_{10} c \\ n \log_{10} 3 - n \log_{10} 2 & > & \log_{10} c \\ \log_{10} 3^n & > & \log_{10} c + log_{10} 2^n \\ \log_{10} 3^n & > & \log_{10} c2^n \\ 3^n & > & c2^n \end{array}$$

   Otherwise, pick: $n = 1$

$$\begin{array}{rcll} \frac{3}{2} & > & c & \text{(Because } c < 1) \\ 3 & > & c2 & \\ 3^n & > & c2^n & \text{(Because } n = 1) \end{array}$$

2. $\log_a n \in \Theta(\log_b n)$
   First, we note that $\log_a n$ is a multiple of $\log_b n$ for any integers a and b (see Figure 1 for proof). Thus, we say $\log_a n = c \log_b n$. So, we must prove that $\exists c_1, c_2, n_0 \forall n \geq n_0, c_1 \log_b n \leq c \log_b n \leq c_2 \log_b n$. So we choose $c_1 = c_2 = c$ and the proof is trivial.

Figure 1: $\log_a n$ is a multiple of $\log_b n$

Let x and n be numbers such that $\log_x n$ is valid.

$$\begin{aligned}\log_x n &= \log_x n \\ \log_x a^{\log_a n} &= \log_x b^{\log_b n} \\ \log_a n \log_x a &= \log_b n \log_x b \\ \log_a n &= \frac{\log_x b}{\log_x a} \log_b n\end{aligned}$$

3. $n! \in O(n^n)$
   Show: $\exists c, n_0 \forall n \geq n_0, n! \leq n^n$
   Choose: $c = 1$, $n_0 = 1$

$$\begin{aligned}n! &\overset{?}{\leq} n^n \\ \frac{n!}{n^n} &\overset{?}{\leq} 1 \\ \frac{1 \cdot 2 \cdots (n-1) \cdot n}{n \cdot n \cdots n \cdot n} &\overset{?}{\leq} 1 \\ \frac{1}{n} \cdot \frac{2}{n} \cdots \frac{n-1}{n} \cdot \frac{n}{n} &\overset{?}{\leq} 1\end{aligned}$$

We notice that $\frac{n}{n} = 1$ and, in all other terms from the last line, the numerator is less than n, so the term is less than 1. Thus $\frac{n!}{n^n}$ is 1 multiplied by a series of fractions less than 1, so it must be less than one, so the inequality must hold, so it must be true that

$$n! \in O(n^n).$$

4. $n^n \in O(n!)^2$
   Show: $\exists c, n_0 \forall n \geq n_0, n^n \leq (n!)^2$
   Choose: $c = 1$, $n_0 = 1$

$$\begin{aligned}n^n &\overset{?}{\leq} (n!)^2 \\ \frac{n^n}{(n!)^2} &\overset{?}{\leq} 1 \\ \frac{n^n}{(n \cdot (n-1) \cdots 2 \cdot 1) \cdot (1 \cdot 2 \cdots (n-1) \cdot n)} &\overset{?}{\leq} 1 \\ \frac{n \cdot n \cdot n \cdots n \cdot n}{(1 \cdot n) \cdot (2 \cdot (n-1)) \cdot (3 \cdot (n-2,)) \cdots ((n-1)\, 2) \cdot (n \cdot 1)} &\overset{?}{\leq} 1 \\ \frac{n}{n} \cdot \frac{n}{2 \cdot (n-1)} \cdot \frac{n}{3 \cdot (n-2)} \cdots \frac{n}{(n-1) \cdot 2} \cdot \frac{n}{n} &\overset{?}{\leq} 1\end{aligned}$$

We see that, in general, each term of the product represented by the last line above is $\frac{n}{i(n-i+1)}$, where i is some integer between 0 and n. We observe that, as in the previous problem, if each term in the product is less than or equal to 1, then so will be the product itself.

$$\frac{n}{i\,(n-i+1)} \overset{?}{\leq} 1$$

$$\begin{aligned} n &\overset{?}{\leq} i\,(n-i+1) \\ n &\overset{?}{\leq} in - i^2 + i \\ i^2 - i &\overset{?}{\leq} in - n \\ i\,(i-1) &\overset{?}{\leq} n\,(i-1) \end{aligned}$$

If $i = 1$, then the last line above becomes $0 \leq 0$, which is true. Otherwise:

$$\begin{aligned} \frac{i\,(i-1)}{i-1} &\overset{?}{\leq} \frac{n\,(i-1)}{i-1} \\ i &\overset{?}{\leq} n \end{aligned}$$

This is obviously true for every term of the aforementioned product, because $0 < i < n$. Thus, all the above inequalities are true, and

$$n^n \in O\,(n!)^2\,.$$

Question 4

If there is a problem with the diskette, see http://www.cs.mcgill.ca/~chundt/comp250.