Types & Programming Languages
Exercises 2 (Partial Solutions)

These exercises are based on the material in Lectures 1, 2, 3 and 4.

1. To prove by induction that for all $n \in N$, $0 + 1 + 4 + \cdots + n^2 = \frac{n(n+1)(2n+1)}{6}$:

   We are trying to prove: for all $n \in N$, $P(n)$ where $P(n)$ means $0+1+4+\cdots+n^2 = \frac{n(n+1)(2n+1)}{6}$.

   For the base case, we prove $P(0)$, which means

   $$0 = \frac{0 \times (0+1) \times (2 \times 0 + 1)}{6}$$

   and this is clearly true.

   For the induction step, we assume $P(k)$ (for a general k) and prove $P(k+1)$.

   This means that we assume

   $$0 + 1 + 4 + \cdots + k^2 = \frac{k(k+1)(2k+1)}{6}$$

   (the induction hypothesis) and prove

   $$0 + 1 + 4 + \cdots + k^2 + (k+1)^2 = \frac{(k+1)((k+1)+1)(2(k+1)+1)}{6}$$

   Using the induction hypothesis, the left hand side of what we are trying to prove is

   $$\frac{k(k+1)(2k+1)}{6} + (k+1)^2$$

   and then algebra completes the proof.

2. Fill in the details in the proof of Type Preservation (Exercise, Lecture 3 Slide 17).

3. *Call by name* means that the only reduction rule for function applications is the rule on Lecture 4 Slide 22. What are the advantages and disadvantages of call by name compared with call by value, from the point of view of practical programming? Hint: think about the function definitions

   ```
   f(x:int):int is 3
   ```

   ```
   g(x:int):int is x+x
   ```

   and the applications

   ```
   f(2+2)
   g(2+2)
   ```

   What would you want to do in a language implementation?

4. The operational semantics we have defined is sometimes called *small step* operational semantics. The alternative is *big step* operational semantics, which directly associates a final value with each expression by means of the following inductive rules. $e \Downarrow v$ means that the result of evaluating the expression e is the value v.

- If v is a value then $v \Downarrow v$.
- $$\frac{e \Downarrow u \quad f \Downarrow v \quad w \text{ is the sum of } u \text{ and } v}{e + f \Downarrow w}$$
- Similar rules for $==$ and $\&$.
- $$\frac{c \Downarrow \mathsf{true} \quad e \Downarrow v}{\mathsf{if}\ c\ \mathsf{then}\ e\ \mathsf{else}\ e' \Downarrow v} \text{ and } \frac{c \Downarrow \mathsf{false} \quad e' \Downarrow v}{\mathsf{if}\ c\ \mathsf{then}\ e\ \mathsf{else}\ e' \Downarrow v}$$

(a) Prove by induction on e that if $e \to e'$ and $e' \Downarrow v$ then $e \Downarrow v$.

(b) Prove that if $e \to^* v$ then $e \Downarrow v$. Use induction on the length of the reduction sequence $e \to^* v$, and the previous result.

(c) Prove by induction on e that if $e \Downarrow v$ then $e \to^* v$.