Name: Aman Agrawal Entry: 2015CS10210 Gp: 1

Indian Institute of Technology Delhi
Department of Computer Science and Engineering

COL226	Programming Languages	Quiz 0
January 10, 2017	10 minutes	Maximum Marks: 10

Instructions. Write your name, entry number and group number in the space provided at the top of the page. Write in black or blue pens. Manage your time effectively. Consider the two programs on OCaml lists:

```
(* length: 'a list -> int *)
let rec length l = match l with
  [ ] -> 0
| x::xs -> 1 + (length xs);;

 (* append: 'a list -> 'a list -> 'a list *)
let rec append l1 l2 = match l1 with
  [ ] ->  l2
| x::xs -> x :: (append xs l2);;
```

10

Q1 (10 marks) **Induction on Lists.**

Prove from the definitions of the functions that for all lists $l_1, l_2 : \alpha$ list,

$$\texttt{length}(\texttt{append}\ l_1\ l_2) = \texttt{length}(l_1) + \texttt{length}(l_2)$$

Proof. By induction on length of l_1 ✓

Base Cases:

$l_1 = [\,] \Rightarrow$ length of $l_1 = 0$ ✓ (2)

Now append $l_1\ l_2 = l_2$

∴ length of (append $l_1\ l_2$) = length of l_2 = length of l_1 + length of l_2

= 0 + length of l_2

Induction Hypothesis: Suppose we have shown that

For a list l_1 of length n (2)

We have

length (append $l_1\ l_2$) = length (l_1) + length (l_2)

= n + length (l_2)

We have to show that it is true for list l_1 of length n+1 ✓ (6)

Induction Cases:

For a list l_1 of length n+1 we have l_1 = x :: xs where xs is a list of length n

append ($l_1\ l_2$) = x :: append (xs l_2)

∴ length (append $l_1\ l_2$) = length (x :: append (xs, l_2)) ✓

= 1 + length (append (xs, l_2)) ✓

By induction hypothesis we have = 1 + length (xs) + length (l_2) = 1 + n + length l_2

∴ For l_1 of length n+1

∴ length (append $l_1\ l_2$) = 1+n + length (l_2) = length (l_1) + length (l_2) ✓