Functional Programming : Assignment - Part I

The objective is to introduce you to functional programming using OCaml. Follow the functional programming paradigm. **Do not** use any of the imperative features of OCaml. Submit the *.ml* files as a single *.zip* file using the submission link in the course page. The evaluation schedule will be announced later.

1. Represent set of integers using OCaml list. Implement the operations $Union$ and $Intersection$ on sets. Each operation should return the resultant set (as a list).

2. Write a routine *findpos(x, l)*, that returns the position of element x in list l, if x is in the list, and returns 0 otherwise. Write a routine *delete(x, l)* that invokes *findpos(x, l)* to first get the position of x in the list, and then removes the element at this position.

3. Encode a list of names l with possible duplicates as a list of pairs of the form *(x, n)*, where x is an element in l and n is the number of occurrences of x in l.