1 Foundations of Computer Science (LCP)

This question has been translated from Standard ML to OCaml

(a) Write brief notes on functions as values and results in OCaml, illustrated with the help of the functionals map and exists. What functions can we obtain from these via currying? [6 marks]

(b) Consider the function zarg defined below:

\[
\text{let rec zarg f l e = match l with}
\]
\[
| [] \rightarrow e
\]
\[
| x::xs \rightarrow f x (zarg f xs e)
\]

Show that with the help of this function, it is possible to write an expression for the sum of a given list of integers. Then describe what zarg does in general. [4 marks]

(c) A polymorphic type of branching trees can be declared as follows. Note that the children of a branch node are given as a list of trees, and that only the leaf nodes carry labels.

\[
\text{type 'a vtree = Lf of 'a}
\]
\[
| Br of ('a vtree) list
\]

(i) Write a function flat t that converts a given tree t of this type to a list of the labels (without eliminating duplicates). Your function should run in linear time in the size of the tree. [4 marks]

(ii) Write a function count x t that counts the number of times that x occurs as a label in t, but without first converting t to a list.

Note: Minimal credit will be given for solutions that use flat. [5 marks]

(iii) What is the type of count? [1 mark]

All OCaml code must be explained clearly and should be free of needless complexity.