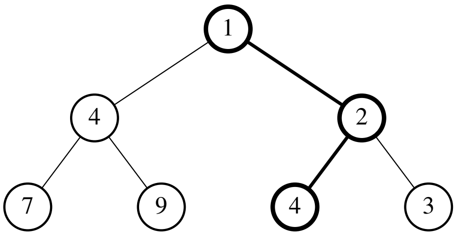
6 Foundations of Computer Science (LCP)

This question has been translated from Standard ML to OCaml

Consider a variant type of binary trees where both leaves and branches carry labels:

```
type 'a tree = Twig of 'a
             | Br of 'a * 'a tree * 'a tree
```

A *path* in a binary tree is a series of labels proceeding from the root to a leaf, as shown in the diagram:



Consider the problem of finding a path in a binary tree such that the integer sum of the labels satisfies a given property. (In the example above, the highlighted path sums to a prime number.)

(*a*) Write an OCaml function `find_path` such that `find_path p t` returns some path in `t` whose sum satisfies the boolean-valued function `p`. If no such path exists, the function should raise an exception. [5 marks]

(*b*) Write an OCaml function `all_paths` such that `all_paths p t` returns the list of all paths in `t` whose sums satisfy the boolean-valued function `p`. [6 marks]

(*c*) Write an OCaml function `all_pathq` that is analogous to `all_paths` but returns a lazy list of paths. For full credit, your function should find paths upon demand rather than all at once. [Hint: try adding solutions to an accumulating argument.] [9 marks]