```
// week 7 file sysytem
type FileSys = Element list

and Element =
    | File of string * string
    | Dir of string * FileSys

let d1 = Dir("d1",[File("a1","java");
                   Dir("d2", [File("a2","fsx");
                              Dir("d3", [File("a3","fs")])]);
                   File("a4","fsx");
                   Dir("d3", [File("a5","pdf")])])


let rec nameFileSys =
    function
    | [] -> []
    | e :: tails -> (nameElement e) @ (nameFileSys tails)

and nameElement =
    function
    | File (s, ext) -> [ s + "." + ext ]
    | Dir (s, fsys) -> s :: (nameFileSys fsys)

//nameElement d1

let rec searchFileSys ext =
    function
    | [] -> Set []
    | e :: tails -> Set.union (searchElement ext e) (searchFileSys ext tails)

and searchElement ext =
    function
    | File (s, extention) when ext = extention -> Set [ s ]
    | Dir (_, fsys) -> (searchFileSys ext fsys)
    | _ -> Set []

//searchElement "fsx" d1

let rec longNamesFileSys =
    function
    | [] -> Set []
    | e :: tails -> Set.union (longNamesElement e) (longNamesFileSys tails)

and longNamesElement =
    function
    | File (s, ext) -> set [ s + "." + ext ]
    | Dir (s, fsys) -> Set.map (fun e -> s + "\\" + e) (longNamesFileSys fsys)

```

```fsharp
longNamesElement d1

```