

Zadanie D: SML - Przeszukiwanie wszerek

Drzewo binarne jest zdefiniowane w następujący sposób:

```
datatype 'a Tree = Leaf | Branch of 'a Tree * 'a * 'a Tree
```

Napisz funkcję `bfnum : 'a Tree -> ('a * int) Tree`. Dla drzewa T funkcja ma stworzyć drzewo T' o identycznej strukturze, w którym wartości w wierzchołkach zostały zastąpione parami składającymi się z początkowej wartości wierzchołka oraz jego numeru w porządku BFS. Numerujemy od zera, więc korzeń zawsze dostanie numer 0, dzieci korzenia 1 i 2, itd. Bfs ma rozpatrywać krawędzie drzewa w kolejności od lewej do prawej (najpierw lewy branch, potem prawy).

Uwaga: Nie jest dozwolone korzystanie ze zmiennych komórek pamięci (ref oraz Array). Programy, które wykorzystują tę funkcjonalność, dostaną błąd CME. Wymagany czas działania programu: $O(n)$ gdzie n to liczba wierzchołków drzewa.

W pliku `.sml` powinna znaleźć się definicja:

```
fun bfnum T = ...
```

Przykład

```
- Control.Print.printDepth := 10;
- bfnum (Branch(Branch(Branch(Leaf,10,Leaf),20,Branch(Leaf,30,Leaf)),40,Branch(Branch(
  Leaf,50,Leaf),60,Branch(Leaf,70,Leaf)))));
val it =
  Branch
    (Branch (Branch (Leaf,(10,3),Leaf),(20,1),Branch (Leaf,(30,4),Leaf)),
      (40,0),
      Branch (Branch (Leaf,(50,5),Leaf),(60,2),Branch (Leaf,(70,6),Leaf)))
  : (int * int) Tree
```