Correction TD 6

2021-05-12

1

Pour un arbre quasi-complet de hauteur h

$$\begin{aligned} & 2^{h-1} \leq n \leq 2^{h+1} - 1 \\ \implies & h - 1 \leq \lg n < h \\ \implies & h = \lfloor \lg n \rfloor \end{aligned}$$

2

```
let est_quasi_complet a =
  let rec aux_list a la = match a with
    | []                -> []
    | Noeud(x, fg, fd) -> aux_list t @ [fg; fd]
    | Vide::t           -> aux_list t
  in
  let rec aux_bool bool l = match l with
    | []                -> true
    | h::t when h=Vide -> aux_bool true t
    | h::t when bool    -> false
    | h::t              -> aux_bool bool t
  in
  aux_bool false (aux_list [a])
```

3

Un tas est un arbre quasi-complet avec étiquette nœud $\geq$ étiquettes fils

```
let est_tas a =
  let rec walk a = match a with
  | Vide -> true
  | Noeud(x, Vide, Vide) -> true (* Pauvre Barbi... *)
  | Noeud(x, Noeud(y, fgg, fdg), Vide) -> (x >= y) && walk Noeud(y, fgg, fgd)
  | Noeud(x, Vide, Noeud(y, fdg, fdd)) -> (x >= y) && walk Noeud(y, fdg, fdd)
  | Noeud(x, Noeud(y, fgg, fgd), Noeud(z, fdg, fdd)) ->
    (x >=(max y z)) && (walk Noeud(y, fgg, fgd)) && (walk Noeud(y, fgd, fdd))
  in
  walk a && est_quasi_complet a
```

4

```
let fg k = 2*k+1
let fd k = 2*k+2
let pere k = (k-1) / 2
```

5

```
let permuter i j t =
  let temp = t.(i) in
  t.(i) <- t.(j);
  t.(j) <- temp

let insérer x k t =
  let rec remonter k t =
    let p = pere k in
    if t.(p) < t.(k) then begin
      permuter p k t;
      remonter p t
    end
    t.(k) <- x
  in
  remonter k t
```

Complexité: $O(\lg n)$