

Correction du devoir à faire pour le Lundi 2 Mars

Exercice 1 (Mots bien parenthésés ou mots de Dyck)

1. Voici le programme demandé :

```
let parentheses m =
  let i = ref 0 and n = Array.length m in
  for k = 0 to n-1 do
    if m.(k) = '(' then
      i := !i+1
    else
      i := !i-1
  done;
  !i = 0
;;
```

Cette condition nécessaire n'est pas suffisante. Par exemple, le mot `) (` contient autant de parenthèses ouvrantes que de parenthèses fermantes mais il n'est pas bien parenthésé.

2. (a) Le profil du mot `((()()))((` est donné par le tableau

`[| 0; 1; 2; 1; 2; 1; 0; -1; 0; 1; 0 |]`.

Ce mot n'est pas bien parenthésé : la 7e parenthèse qui est fermante n'est pas associée à une parenthèse ouvrante par exemple.

- (b) Voici le programme demandé :

```
let profil m =
  let n = Array.length m in
  let p = Array.make (n+1) 0 in
  for k = 0 to n-1 do
    if m.(k) = '(' then
      p.(k+1) <- p.(k) + 1
    else
      p.(k+1) <- p.(k) - 1
  done;
  p
;;
```

- (c) — L'ensemble $\{i \in \llbracket 1, l(m) \rrbracket / p_m(i) = 0\}$ est une partie finie (elle contient au plus $l(m)$ éléments) et non vide (car $l(m)$ appartient à cette partie par hypothèse). Elle contient donc un minimum noté k .

— Supposons que $k = l(m)$.

Comme p_m est à valeur dans $\mathbb{N}$ et que $p_m(0) = 0$, on a nécessairement $p_m(1) = 1$ et donc m commence par une parenthèse `(`.

De la même façon, comme p_m est à valeur dans $\mathbb{N}$ et que $p_m(l(m)) = 0$, on a nécessairement $p_m(l(m) - 1) = 1$ et donc m termine par une parenthèse `)`.

Il existe donc un mot m' de longueur $l(m) - 2$ tel que $m = (m')$. Son profil $p_{m'}$ est une application définie sur $\llbracket 0, l(m) - 2 \rrbracket$ à valeurs dans $\mathbb{Z}$ tel que :

$$\forall i \in \llbracket 0, l(m) - 2 \rrbracket, \quad p_{m'}(i) = p_m(i + 1) - 1.$$

Par minimalité de $k = l(m)$, $p_m(i) \neq 0$ pour tout $i \in \llbracket 1, l(m) - 1 \rrbracket$.

Comme p_m est à valeur dans $\mathbb{N}$, on en déduit que $p_m(i) \geq 1$ pour tout $i \in \llbracket 1, l(m) - 1 \rrbracket$. Alors :

$$\forall i \in \llbracket 0, l(m) - 2 \rrbracket, \quad p_{m'}(i) = p_m(i + 1) - 1 \geq 1 - 1 = 0.$$

Et $p_{m'}(l(m')) = p_{m'}(l(m) - 2) = p_m(l(m') - 1) - 1 = 1 - 1 = 0$.

— Supposons que $k < l(m)$.

On pose m_1 le mot constitué des k premières lettres de m et m_2 le mot constitué des $l(m) - k$ dernières lettres de m , de sorte que $m = m_1 m_2$ (m_1 et m_2 sont bien de longueur strictement inférieure à $l(m)$ car $1 \leq k < l(m)$).

Alors :

— Pour tout $i \in \llbracket 0, l(m_1) \rrbracket$, $p_{m_1}(i) = p_m(i)$ qui est à valeurs dans $\mathbb{N}$ par hypothèse.

Et $p_{m_1}(l(m_1)) = p_m(l(m_1)) = p_m(k) = 0$.

— Pour tout $i \in \llbracket 0, l(m_2) \rrbracket$, $p_{m_2}(i) = p_m(k + i)$ qui est à valeurs dans $\mathbb{N}$ par hypothèse.

Et $p_{m_2}(l(m_2)) = p_m(k + l(m_2)) = p_m(l(m_1) + l(m_2)) = p_m(l(m)) = 0$.

— Remarquons d'abord que, par définition de p_m , $p_m(i)$ est de la même parité que i . Donc, avec la condition $p_m(l(m)) = 0$, $l(m)$ est nécessairement paire.

Raisonnons maintenant par récurrence sur la longueur de m .

— Initialisation (pour $l(m) = 2$) : Par hypothèse, $p_m(0) = p_m(2) = 0$ et p_m est à valeurs dans $\mathbb{N}$. Donc $p_m(1) = 1$ et donc $m = ()$ qui est bien parenthésé.

— Hérité : Soit m un mot de longueur $l(m) \geq 4$ et supposons la propriété établie pour tous les mots de longueur inférieure strictement à $l(m)$.

On pose k comme dans l'énoncé.

Si $k = l(m)$, alors $m = (m')$. Par hypothèse de récurrence, m' est bien parenthésé et donc m est bien parenthésé.

Si $k < l(m)$, alors $m = m_1 m_2$. Par hypothèse de récurrence, m_1 et m_2 sont bien parenthésés et donc m est bien parenthésé.

(d) Par induction structurale.

— Etape de base :

$m = \varepsilon$ et le profil de m vérifie les conditions de l'énoncé.

— Etape d'induction :

— Si $m = (m')$ avec m' bien parenthésé, alors par hypothèse d'induction

$$\begin{cases} p_{m'}(l(m')) = 0 \\ p_{m'} \text{ est à valeurs dans } \mathbb{N} \end{cases}$$

Par définition $p_m(0) = 0$. Comme $m = (m')$,

$$\forall i \in \llbracket 1, l(m) - 1 \rrbracket, \quad p_m(i) = p_{m'}(i - 1) + 1 \geq 1$$

et $p_m(l(m)) = p_m(l(m) - 1) - 1 = p_{m'}(l(m) - 2) + 1 - 1 = p_{m'}(l(m')) = 0$.

Le profil de m vérifie donc les hypothèses de l'énoncé.

- Si $m = m_1 m_2$ avec m_1 et m_2 bien parenthésés, alors par hypothèse d'induction

$$\begin{cases} p_{m_1}(l(m_1)) = 0 \\ p_{m_1} \text{ est à valeurs dans } \mathbb{N} \end{cases} \quad \text{et} \quad \begin{cases} p_{m_2}(l(m_2)) = 0 \\ p_{m_2} \text{ est à valeurs dans } \mathbb{N} \end{cases}$$

Comme $m = m_1 m_2$,

$$\forall i \in \llbracket 0, l(m_1) \rrbracket, \quad p_m(i) = p_{m_1}(i)$$

et, comme $p_{m_1}(l(m_1)) = 0$,

$$\begin{aligned} \forall i \in \llbracket l(m_1) + 1, l(m) \rrbracket, \quad p_m(i) &= p_{m_2}(i - l(m_1)) + p_{m_1}(l(m_1)) \\ &= p_{m_2}(i - l(m_1)). \end{aligned}$$

Comme p_{m_1} et p_{m_2} sont à valeurs dans $\mathbb{N}$, p_m également. De plus,

$$p_m(l(m)) = p_{m_2}(l(m) - l(m_1)) = p_{m_2}(l(m_2)) = 0.$$

Le profil de m vérifie donc les hypothèses de l'énoncé.

- (e) Un mot m est bien parenthésé si et seulement si $\begin{cases} p_m(l(m)) = 0 \\ p_m \text{ est à valeurs dans } \mathbb{N} \end{cases}$.

On a montré le sens direct à la question précédente et la réciproque à la 2.(c).

- (f) Voici la fonction demandée :

```
let bien_parenthese m =
  let p = profil m and n = Array.length m and k = ref 0 in
    while !k <= n and p.( !k) >= 0 :
      k := !k + 1
    done;
  !k = n+1
;;
```