Informatique - MPI

Question 1.

☐ **1.1.** Si n est un entier naturel non nul, notons u_n la valeur du dernier chiffre de 2^n dans sa représentation décimale. On peut écrire :

$$u_n = 2^n \bmod 10$$

On montre (le faire) alors que :

$$u_{n+1} = (2u_n) \bmod 10$$

Ce qui permet de calculer le dernier chiffre de 2^ns par l'algorithme *récursif non terminal* suivant.

```
let rec dernier_chiffre n =
  if n = 0 then 1
  else 2 * (dernier_chiffre (n-1)) mod 10
```

☐ **1.2.**

- Terminaison : la valeur de n décroît et le cas $n = 0$ met un terme aux appels.
- Correction : par induction.
- Complexités temporelle et spatiale : linéaire.

Question 2.

☐ **2.1.** Fonction *récursive terminale* `dernier_chiffre : int -> int`.

```
let dernier_chiffre n =
  let rec aux acc i =
    if i = 0 then acc
    else 2 * (aux acc (i-1)) mod 10
  in aux 1 n
```

☐ **2.2.**

- Terminaison : la valeur de i décroît et le cas $i = 0$ met un terme aux appels récursifs de la fonction `aux`.
- Correction : par induction.
- Complexité temporelle linéaire ; complexité spatiale constante.

Question 3.

☐ **3.1.** Observons les résultats des calculs suivants.

n	1	2	3	4	5	6	7	8	9
$n \bmod 4$	1	2	4	0	1	2	4	0	1
2^n	2	4	8	16	32	64	128	256	512
$2^n \bmod 10$	2	4	8	6	2	4	8	6	2

Il semble que pour $n \geqslant 1$, les derniers chiffres de 2^n suivent un cycle d'ordre 4.

- Si $n \bmod 4 = 0$ alors $2^n \bmod 10 = 6$.
- Si $n \bmod 4 = 1$ alors $2^n \bmod 10 = 2$.
- Si $n \bmod 4 = 2$ alors $2^n \bmod 10 = 4$.
- Si $n \bmod 4 = 3$ alors $2^n \bmod 10 = 8$.

Justifions ces résultats en écrivant n sous la forme $i + 4j$ avec $i \in \{1, 2, 3, 4\}$ et $j \in \mathbb{N}$. Pour $j = 0$, le résultat est évident. Pour $j \geqslant 1$, on a $2^{4j} = 16^j = 6^j \bmod 10$ puis $6^j \bmod 10 = 6 \bmod 10$. En effet, pour $j = 1$, $6^j = 6$. Si le résultat est vrai pour un entier $j \geqslant 1$ fixé, alors $6^{j+1} \bmod 10 = 36 \times 6^{j-1} \bmod 10 = (30 + 6) \times 6^{j-1} \bmod 10 = 6 \times 6^{j-1} \bmod 10 = 6^j \bmod 10$, qui vaut $6 \bmod 10$ par hypothèse de récurrence. Ce qui établit le résultat. Finalement :

$$2^{i+4j} \bmod 10 = 6 \times 2^i \bmod 10 = \begin{cases} 2 & \text{si } i = 1 \\ 4 & \text{si } i = 2 \\ 8 & \text{si } i = 3 \\ 6 & \text{si } i = 4 \end{cases}$$

☐ **3.2.** Le résultat de la question précédente permet l'écriture du code suivant.

```
let dernier_chiffre n =
  if n = 0 then 1
  else match n mod 4 with
    | 0 -> 6
    | 1 -> 2
    | 2 -> 4
    | 3 -> 8
```

- Terminaison : toujours (pas d'appels récursifs).
- Correction : par récurrence.
- Complexités temporelle et spatiale : constantes.