MPSI : TP Organisation des activations sous forme d'arbre en cas d'appels multiples.

En mathématiques, la suite de Fibonacci est une suite d'entiers dans laquelle chaque terme est la somme des deux termes qui le précèdent.

Notée (F_n), elle est définie par $F_0 = 0, \quad F_1 = 1$, et $F_n = F_{n-1} + F_{n-2}$ pour $n \geqslant 2$.

Q.1 Calculer F_5, F_6

Q.2 Ecrire en très peu de lignes OCaml la fonction récursive `fib : int -> int`. On ne cherchera pas à mémoriser les résultats intermédiaires.

```
 # fib 0, fib 1, fib 2;;
- : int * int * int = (0, 1, 1)
```

Q.3 En représentant les appels internes à `fib` dans `fib 4` sous forme d'arbre, établir que certains appels sont effectués plusieurs fois.

Q.4 Donner une relation de récurrence vérifiée par la complexité temporelle de cette fonction.

Q.5 En déduire que la complexité est au moins exponentielle.

Q.6 Cette utilisation multiple des résultats intermédiaires suggère l'utilisation d'un tableau pour les mémoriser (on parle de *mémoïzation*.

Modifier la fonction `fib` pour *mémoïzer* les résultats intermédiaires.

```
# fib 15;;
- : int = 610
```

Q.7 Montrer que pour tous $(p, q) \in \mathbb{N} \times \mathbb{N}^*$, $F_{p+q} = F_{p+1}F_q + F_pF_{q-1}$.

Montrer d'abord que la relation est vraie pour p fixé et $q = 1$ puis $q = 2$ puis raisonner par récurrence.

Q.8 Montrer que pour tout $n \in \mathbb{N}$ on a

$$\begin{aligned} (F_{2n}, F_{2n+1}) &= (2F_nF_{n+1} - F_n^2, F_{n+1}^2 + F_n^2) \\ (F_{2n+1}, F_{2n+2}) &= (F_{n+1}^2 + F_n^2, 2F_n, F_{n+1}^2 + 2F_nF_{n+1}) \end{aligned}$$

Q.9 En déduire une fonction `fib2` efficace.

Q.10 Donner la complexité de `fib2` si on compte pour $O(1)$ les opérations arithmétiques (ce qui est faux à partir d'une certaine taille).