Cette onzième colle vous fera écrire une fonction simple en OCaml sur des listes associative, et un exercice en C sur une fonction entière récursive.

Ex.1 Autour de listes associatives - OCaml (25 minutes)

Une liste associative est une liste de valeurs `(x, y)` de type `'a * 'b`, et représente une association : à `x` est associée `y`. Cette structure de donnée naïve permet de représenter des tables associatives, et on supposera que pour chaque paire `(x, y)` dans une liste associative, la clé `x` n'est présente qu'une et une seule fois.

1. Écrire la fonction `assoc : 'a -> ('a * 'b) list -> 'b` qui à une valeur `x : 'a` et à une liste `[(x0,y0); (x1,y1); (x2,y2); ...; (xn,yn)]` de type `('a * 'b) list` associe la valeur correspondante à `x` (de type `'b`), si elle existe, ou une exception `failwith "Not found"` sinon.
2. Écrire la même fonction mais avec le typage `assoc_opt : 'a -> ('a * 'b) list -> 'b option` qui renvoie `None` si `x` n'est pas dans la liste (et pas d'exception) et `Some y` si `x` est présent dans la liste.
3. Tester sur *au moins* deux exemples.
4. Quelle est la complexité temporelle (dans le pire cas) de votre fonction `assoc`, exprimée avec n la longueur de la liste ?

Ex.1 Fonction 91 de McCarthy - C (30 minutes)

Écrire une fichier C `colle11.c` important `stdio.h`, `string.h`.

On rappelle qu'on compile ce fichier avec `COMPILATEUR = gcc` ou `clang` avec la ligne de commande suivante, puis on exécute le binaire produit avec la deuxième ligne (sans les dollars qui représentent le prompt de la ligne de commande du terminal) :

```
$ COMPILATEUR -O0 -Wall -Wextra -Wvla -Werror -fsanitize=address
  -fsanitize=undefined -pedantic -std=c11 -o colle11.exe colle11.c
$ ./colle11.exe
```

Si une ligne affiche un résultat qui vous semble bizarre, commenter le.

On définit sur $\mathbb{N}$ la fonction de McCarthy f_M par

$$f_M(n) = \begin{cases} n - 10 & \text{si } n > 100 \\ \\ f_M(f_M(n+11)) & \text{sinon.} \end{cases}$$

— Écrire cette fonction `f_M` en C.

— Dans votre `main`, afficher ses valeurs pour tout n entre 0 et 120 inclus. Qu'observez-vous ?

— Justifier bien proprement que cette fonction termine pour tout $n \in \mathbb{N}$.

— (plus difficile) Justifier votre observation, pour tout $0 \leqslant n \leqslant 101$.

— Quelle est la complexité temporelle asymptotique de $f_M(n)$? *Attention il y a un piège.*

