HASKELL

TD 2 exercice 1

```
compteUno :: [Int] -> Int
compteUno [] = 0
compteUno x:l = compte'

compte' :: [int] -> int -> int
compte' [] i = i
compte' (x:xs) i = if x == 1 then compte xs i+1
                   else compte xs i

compte2 :: [int] -> int
compte2 [] = 0
compte2 (x:xs) = if x == 2
                 then 1 + compte2 xs
                 else comtpte2 xs

compteX :: [Int] -> Int -> Int
compteX (y:xs) x = if y == x
                   then 1 + compteX (xs) xe
                   else compteX (xs) x

comptepositifs :: [Int] -> Int
comptepositifs (x:xs) = if x > 0
                        then 1 + (comptepositifs xs)
                        else (comptepositifs xs)
```

```
moyenne :: [Int] -> Float
moyenne [] = 0
moyenne l =
```