

## CAML vs ALGO

Que fait le "programme" ci-dessous ?

```
1
2 let test n =
3   (n >= 100) && (n < 1000) ;;
4
5
6
7 let sum_digits n =
8   let (a, b, c) =
9     (n / 100, (n/10) mod 10, n mod 10)
10  in
11    a + b + c ;;
12
13
14
15 let product_digits n =
16   let (a, b, c) =
17     (n / 100, (n/10) mod 10, n mod 10)
18   in
19     a * b * c ;;
20
21
22
23 let abs n =
24   if n > 0 then
25     n
26   else
27     -n ;;
28
29
30
31 let rec loop n =
32   let n = abs n in
33   if sum_digits n = product_digits n
34   then
35     n
36   else
37     loop (n+1) ;;
```



```
fonction test (entier n) : booléen
debut
  retourne ( (n >= 100) et (n < 1000) )
fin
```

```
fonction sum_digits (entier n) : entier
variables
  entier a, b, c
debut
  a ← n div 100
  b ← (n div 10) mod 10
  c ← n mod 10
  retourne (a + b + c)
fin
```

```
fonction product_digits (entier n) : entier
variables
  entier a, b, c
debut
  a ← n div 100
  b ← (n div 10) mod 10
  c ← n mod 10
  retourne (a * b * c)
fin
```

```
fonction abs (entier n) : entier
debut
  si n > 0 alors
    retourne n
  sinon
    retourne -n
  fin si
fin
```

```
fonction loop (entier n) : entier
debut
  n ← abs(n)
  si sum_digits(n) = product_digits(n) alors
    retourne n
  sinon
    retourne loop(n+1)
  fin si
fin
```