

## TD1

jeudi 8 septembre 2022 11:45

let vaut5 = fun x → x = 5

let vaut5\_str = fun x → if x = 5 then "oui" else "non"

r

## Exercice 19 : Nombres complexes

- Définir le type complexe
- Définir l'élément neutre pour l'addition des nombres complexes.
- Écrire une fonction qui additionne deux nombres complexes.
- Écrire une fonction qui donne le module d'un nombre complexe.
- Écrire une fonction qui donne l'opposé d'un nombre complexe.

⚠ Attention au majuscules dans identifiants

type cplx = float \* float

let c1 = (1.5, 2.3) ou let c1: cplx = (1.5, 2.3) (pas de diff entre types float)

let add\_cplx = fun c1 c2 → let (x1, y1) = c1 and (x2, y2) = c2 in (x1 + x2, y1 + y2)

let imag = fun c1 → let (r, e) = c1 in e ou en let c3 = (x1 + x2, y1 + y2)

let real = fun c1 → let (r, e) = c1 in r

let inverse = fun c1 → let (r, e) = c1 in (-r, -e)

let module = fun c1 → let (r, e) = c1 in sqrt(r<sup>2</sup> + e<sup>2</sup>)

let neutre = (0, 0)

type cplx2 = Cpx of float \* float

let c1 = Cpx(1.5, 2.3)

let add\_cplx2 = fun c1 c2 → match c1 with Cpx(x1, y1) → match c2 with Cpx(x2, y2) →

## Exercice 7 : Factorielle

Écrire une fonction qui calcule n!.

let rec fact = fun n → if n = 0 then 1 else (n \* fact(n-1))

let res = let f5 = fact 5 in let 2f5 = 2 \* f5 in (2022 - 2f5) \* f5 + 2f5 + f5 -

## Exercice 2 : Aires

Écrire des fonctions qui calculent :

- L'aire d'un carré de côté  $a$ .
- L'aire d'un rectangle de côtés  $a$  et  $b$ .
- L'aire d'un cercle de rayon  $r$ .
- L'aire d'un triangle rectangle de côté  $a$  et d'hypoténuse  $h$  (en utilisant le théorème de Pythagore).

 $a^2$  $h^2 =$  $b^2 = h^2$ 

let aire\_carre = fun c → sq c

let aire\_rect = fun a b → a \* b

let aire\_cer = fun r → pi \* (sq r)

let aire\_tr = fun h a → let b = sqrt ((sqr h) + (sqr a)) in (b \* a) /. 2

type Figure =

| Carre of Float  
| Rect of Float \* Float  
| Cerc of Float  
| Tr of Float \* Float

let aire = fun Figure → match Figure with

(= modif de f

| Carre(c) → aire-ca c  
| Rect(a,b) → aire-rect a b  
| Cerc(r) → aire-cer r  
| Tr(h,a) → aire-tr h a