

Interrogation écrite 2

INF 201 — IMA3&4 — 27/02/2026 — 25 minutes

Exercice 1. (/10) On souhaite avoir un type qui représente soit un scalaire appartenant à $\mathbb{R}$, soit un vecteur dans le plan $\mathbb{R}^2$. Mathématiquement on écrira:

$$\text{euclidien} = \{\text{Scalaire}(s), s \in \mathbb{R}\} \cup \{\text{Vec2}(v), v \in \mathbb{R} \times \mathbb{R}\}$$

Question 1. (/2) En déduire une implémentation en OCaml du type `euclidien`:

Question 2. (/5) On souhaite réaliser une fonction `produit_scalaire` qui effectue le produit scalaire de deux types `euclidien`. Le produit scalaire de deux réels sera juste leur produit tandis que le produit scalaire d'un réel λ et d'un vecteur v sera $(\lambda \ 0) \cdot v$. Donner le profil de la fonction:

```
val produit_scalaire :
```

compléter les exemples:

```
produit_scalaire (Scalaire(3.)) (Vec2(1., 1.)) =  
produit_scalaire (Vec2(1.,1.)) (Vec2(1., -1.)) =
```

et enfin l'implémenter:

Question 3. (/3) On souhaite en déduire une fonction `norme` qui donne la longueur d'un vecteur. Pour un `Scalaire` on renverra la valeur absolue à l'aide de la fonction `abs_float` tandis que pour un `Vec2` on utilisera la fonction `sqrt`: `float -> float` après un appel judicieux à `produit_scalaire`. Compléter les exemples.

```
norme (Scalaire (-2.)) =  
norme (Vec2 (1., 1.)) =
```

Puis donner une implémentation:

Exercice 2. (/11) De façon analogue à ce qui a été vu en CM/TD on définit les entiers de Peano de la façon suivante:

$$\text{peano} = \{Z\} \cup \{S(p), p \in \text{peano}\}$$

Le type `peano` peut-être implémenté en OCaml de la façon suivante:

```
type peano = Z | S of peano;;
```

Question 4. (/2) Donner une implémentation non récursive d'une fonction `est_nul : peano -> bool` qui dit si un entier de Peano est nul:

Question 5. (/4) On aimerait avoir une fonction `int_vers_peano : int -> peano` qui donne la représentation de Peano d'un entier positif.

Compléter les exemples:

```
int_vers_peano 0 =  
int_vers_peano 3 =
```

Donner une implémentation récursive de cette fonction:

Question 6. (/2) On suppose l'existence d'une fonction `peano_vers_int` qui transforme un entier dans la représentation de Peano en un entier

Compléter les exemples:

```
peano_vers_int Z =  
peano_vers_int (S (S Z)) =
```

Que peut-on dire de la fonction `peano_vers_int o int_vers_peano`:

Question 7. (/3) Finalement on veut écrire une fonction `lire: peano -> string` qui « lit » le nombre de `peano` dans le langage naturel:

```
lire Z = "Zero"  
lire (S (S (S Z))) = "Successeur_de_Successeur_de_Successeur_de_Zero"
```

Donner une implémentation récursive de cette fonction: