

# TP 5 : Tableaux, Structures, Pointeurs en C

22 novembre 2023

## Exercice 1

- 1) Définir un type `complexe` prenant deux champs flottants correspondant à la partie réelle et la partie imaginaire.
- 2) Définir une fonction `addition` qui prend en entrée deux complexes et renvoie leur somme.
- 3) Définir une fonction `multi` qui prend en entrée deux complexes et renvoie leur produit.
- 4) Définir une fonction `conjugue` qui prend en entrée un complexe et renvoie son conjugué.
- 5) Définir une fonction `division` qui prend en entrée deux complexes  $z_1$  et  $z_2$  avec  $z_2 \neq 0$  et renvoie  $z_1/z_2$ .

## Exercice 2

- 1) Définir un type `point` prenant deux champs flottants correspondant à l'abscisse et l'ordonnée d'un point de  $\mathbb{R}^2$ .
- 2) Définir une fonction `project` qui prend en entrée trois points  $A$ ,  $B$  et  $C$  et renvoie la projection orthogonale de  $A$  sur  $(BC)$ .
- 3) Définir une fonction `aire` qui prend en entrée trois points  $A$ ,  $B$  et  $C$  et renvoie l'aire du triangle  $ABC$ .

## Exercice 3

Ecrire une fonction `void incr(int *x)` qui augmente d'une unité la valeur de `*x`.

## Exercice 4

Ecrire une fonction `void syracuse(int *x)` qui change la valeur de `*x` en son successeur dans la suite de Syracuse.

## Exercice 5

Ecrire une fonction `void swap(int *x, int *y)` qui échange les valeurs de `*x` et `*y`.

## Exercice 6

Ecrire une fonction `void minmax(int *x, int *y)` qui place dans `*x` la plus grande valeur entre `*x` et `*y` et place dans `*y` la plus petite valeur.

## Exercice 7

Lorsqu'une fonction prend en entrée un tableau, et si des calculs doivent être faits à partir de la longueur du tableau, alors il faut la rentrer en argument de la fonction.

Ecrire une fonction `bool est_triee(int a[], int n)` qui vérifie qu'un tableau `a` est trié.

**Exercice 8**

Ecrire une fonction `void swaptab(int a[], int i, int j)` qui intervertit les éléments aux positions  $i$  et  $j$  du tableau  $a$  (on suppose qu'il s'agit bien d'indices du tableau).

**Exercice 9**

Ecrire une fonction `int recherche_dicho(int v, int a[], int n)` qui renvoie l'indice d'apparition de  $v$  dans  $a$  si  $v$  appartient au tableau  $a$ , et -1 sinon.

**Exercice 10**

Ecrire une fonction `init_tableau` qui prend en entrée un entier  $n$  et demande à un utilisateur de remplir une à une les valeurs du tableau.

**Exercice 11**

Ecrire une fonction `tri_bulles` qui effectue le tri à bulles en place sur un tableau.