

Manipulation de "struct"

1. Afin de manipuler des points dans un espace à 2 dimensions, on définit le type `point` :

```
typedef struct {float x, y;} point;
```

- 1.1. Définir la fonction `distance` qui prend 2 points en paramètres et renvoie un réel qui est la distance entre ces 2 points. (on utilisera la fonction prédéfinie `sqrt(x)` pour calculer la racine carrée de `x`).
 - 1.2. Définir la fonction `perimetre_triangle` qui prend 3 points en paramètres et retourne la longueur du périmètre du triangle défini par ces 3 points. Tester cette fonction dans un programme qui saisit les coordonnées des 3 points et affiche le périmètre calculé.
2. Donner en C la structure de données (struct) `cercle`, permettant de définir un cercle par son centre et son rayon. Ecrire le programme qui saisit les informations relatives à un cercle ainsi que les coordonnées d'un point `p`, puis affiche le périmètre du cercle et indique si le point `p` se trouve sur le cercle, à l'intérieur ou à l'extérieur du cercle.

Algorithmes de tri

1. Ecrire la fonction `tri_bulles` qui effectue le tri à bulles (cf. cours) d'un tableau de `n` entiers, avec `n` défini en constante et le tableau en variable globale.
2. Ecrire la fonction `tri_select` qui effectue également le tri d'un tableau en utilisant l'algorithme de tri par sélection (cf. cours).
3. Ecrire un programme permettant de saisir les valeurs d'une tableau d'entiers, de lancer l'un ou l'autre des 2 algorithmes de tri et d'afficher le tableau trié.