

# Point minimal

## Point minimal

On considère dans cet exercice une liste contenant les coordonnées de points de l'espace. Chaque point est représenté par un triplet des trois coordonnées  $(x, y, z)$ .

Par exemple :

```
#      (x  y  z)  (x  y  z)  ...  (x  y  z)
points = [(9, 2, -8), (-3, 9, -8), (1, 9, 7), (-3, 6, 7), (-7, 1, 3), (-4, 0, 5)]
```

On souhaite récupérer les coordonnées du point ayant la plus petite coordonnée selon  $x$ ,  $y$  ou  $z$ .

## Objectif

On demande donc d'écrire la fonction `minimum` qui prend en paramètre la liste `points` des coordonnées des points ainsi qu'une chaîne de caractères `composante` valant soit `"0x"`, `"0y"` ou `"0z"` et renvoyant les coordonnées du point ayant la plus petite coordonnée pour la composante indiquée.

On garantit que la liste est non-vide et que le paramètre `composante` est toujours égal à `"0x"`, `"0y"` ou `"0z"`.

Si deux points ont la même valeur minimale pour la composante indiquée, la fonction renverra le point ayant le plus petit indice dans la liste `points`.

## Exemples

```
>>> points = [(9, 2, -8), (-3, 9, -8), (1, 9, 7), (-3, 6, 7), (-7, 1, 3), (-4, 0, 5)]
>>> minimum(points, "0x")
(-7, 1, 3)
>>> minimum(points, "0y")
(-4, 0, 5)
>>> minimum(points, "0z")
(9, 2, -8)
```