

# NSI - Première

## TD - tri sélection

### Exercice 1

1. Faire tourner l'algorithme du tri par sélection présenté en cours sur le tableau [5, 7, 3, 1, 9]
2. Combien de comparaisons sont nécessaires pour trier ce tableau ?
3. Recommencer avec le tableau déjà trié [1, 3, 5, 7, 9].
4. Gagne-t-on quelque chose à partir d'un tableau déjà trié ?

### Exercice 2

1. Proposer une version avec des indices, en langage naturel du tri par sélection.
2. Faire tourner votre version sur le tableau [5, 7, 3, 1, 9]

Compléter le tableau suivant, en écrivant les éléments triés et non triés après chaque tour de la boucle principale.

Étape	Partie triée	Partie non triée
1	[]	[5, 7, 3, 1, 9]
2		

### Exercice 3 - Vérifier qu'un tableau est trié

On dispose d'un tableau de nombres, on souhaite vérifier s'il est trié par ordre croissant.

Par exemple : 1, 2, 3, 5, 4, 6, 7 n'est pas trié mais 1, 2, 3, 4, 5, 6, 7 est trié.

1. Proposer un algorithme permettant de vérifier qu'un tableau de nombres est trié. Quelle devrait en être sa signature ?
2. Combien de comparaisons sont nécessaire pour s'assurer qu'un tableau est trié ?
3. Traduire cette fonction en Python
4. Si ce n'est pas déjà fait, améliorer l'algorithme en quittant dès qu'on repère deux éléments non triés dans le tableau.