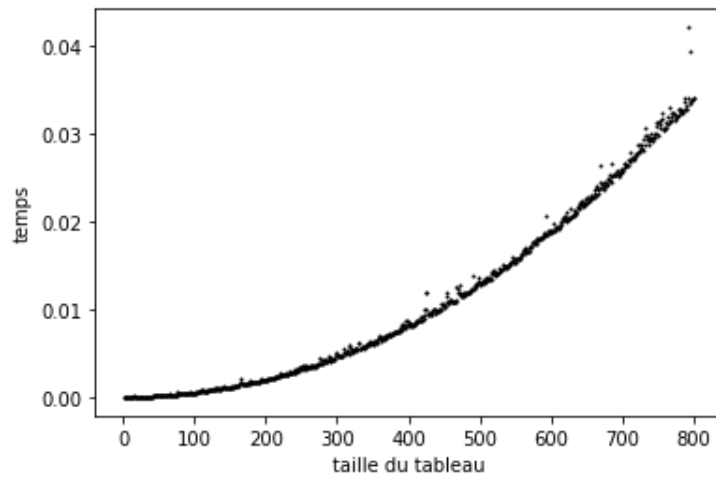


Algorithmes de tri (mini projet):

Exercice 1: tri par insertion:

1. A l'aide du pseudocode vu en cours, écrire un programme permettant de trier un tableau t par insertion.
2. Transformer ce programme afin d'écrire une fonction `tri_insertion(t)` permettant de trier un tableau t par insertion.
3. (bonus) Tracer un graphique représentant le temps de calcul en fonction de la taille du tableau. Ce graphique est il cohérent avec une complexité $O(n^2)$? On pourra utiliser les bibliothèques `time` et `matplotlib`. On devrait obtenir un graphique du type:



Exercice 2: tri par selection:

1. A l'aide du pseudocode vu en cours, écrire un programme permettant de trier un tableau t par sélection.
2. Transformer ce programme afin d'écrire une fonction `tri_selection(t)` permettant de trier un tableau t par selection.
3. (bonus) Tracer un graphique représentant le temps de calcul en fonction de la taille du tableau. Ce graphique est il cohérent avec une complexité $O(n^2)$? On pourra utiliser les bibliothèques `time` et `matplotlib`. On devrait obtenir un graphique du type:

