

## TP N°2 - Algorithmes de Tri (II)

### Exercice 1.

Triez les tableaux suivants en utilisant les algorithmes counting sort et radix sort.

- i. [1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 1, 1, 1] (counting seulement)
- ii. [132, 31, 45, 52, 134, 132, 52]
- iii. [a, b, b, d, e, f, c, g, a, s, b, t, h] (counting seulement)
- iv. [abac, aab, cada, pepe, toto, perro, abra, zorro] (radix seulement)
- v. [(1, f), (4, a), (1, a), (4, b), (3, t), (1, z), (4, s), (2, w)] (trier par la 1<sup>ère</sup> composante)

### Exercice 2.

Implémentez et testez tous les algorithmes de tri vus en cours.

### Exercice 3.

Calculez la complexité temporelle (en détail) de tous vos algorithmes. Dans quels cas les algorithmes sont meilleurs que les algorithmes classiques ? Justifiez.