

UNIVERSITÉ JEAN MONNET

FACULTÉ DES SCIENCES ET TECHNIQUES

PROJET DE PROGRAMMATION

LICENCE 1 MATH-INFO
2016-2017

BINÔME :

Hugo Mohamed et Thomas Grant.

Table des matières

1 Fonctions de tri	2
1.1 Tri par sélection du minimum	2
1.2 Tri par partition-fusion	2
1.3 Tri par sélection du maximum	2
1.4 Tri par insertion	2
1.5 Tri par pivot	2
1.6 Tri à bulle	2
2 Choix d'une fonction de tri	3
2.1 Création d'un module Test efficacite	3
2.2 Test	3
3 Module List sup définitif	3
3.1 Les Fonctions retenues	3

1 Fonctions de tri

1.1 Tri par sélection du minimum

La fonction `tri_selection_min` utilise les fonctions `selectionne` et `supprime` pour sélectionner le plus petit élément de la liste et de l'ajouter au début de la liste sans cet élément qui aura été supprimé de sa position initiale.

1.2 Tri par partition-fusion

On peut créer la fonction `tri_partition_fusion` en utilisant les fonctions `partitionne` et `fusionne` où la liste reçue est séparée en deux listes pour être triées séparément, puis elles sont fusionnées ensemble pour former une liste triée suivant le comparateur.

1.3 Tri par sélection du maximum

Pour cette fonction de tri, nous avons fait une fonction bis pour pouvoir ajouter le plus grand élément de la liste que l'on veut traiter dans une nouvelle liste vide.

1.4 Tri par insertion

Pour cette fonction, on utilise une liste vide que l'on remplit en ajoutant les éléments de la liste donnée au fur et à mesure dans le bon ordre.

1.5 Tri par pivot

Pour `tri_pivot`, nous divisons la liste en 3 listes à partir du premier élément, puis nous recommençons jusqu'à obtenir plusieurs listes à un élément que l'on concatène.

1.6 Tri à bulle

Cette fonction trie la liste en comparant les éléments deux à deux de manière récursive et en les intervertissant si nécessaire. Au final, on compare notre liste initiale avec la liste modifiée et on recommence tant que les deux listes ne sont pas égales.

2 Choix d'une fonction de tri

2.1 Création d'un module Test efficacite

Nous avons créé un module `efficacite` dans lequel nous avons défini `temps_exe_min` à partir de l'aide fourni dans le sujet pour connaître la vitesse d'un programme.

Nous pouvons donc ensuite comparer toutes nos fonctions de tri en fonction de la liste donnée.

2.2 Test

Nous avons décidé de définir la fonction la plus rapide en se basant sur une moyenne des tests effectués avec des petites listes, puis des grandes listes, en utilisant la fonction `Random_list`.

En testant toutes les fonctions deux à deux, on se rend donc compte que la plus rapide est `tri_pivot`.

3 Module List sup définitif

3.1 Les Fonctions retenues

La fonction `tri` :

```
#let tri = tri_pivot
```

On définit tout simplement la fonction la plus rapide comme étant la fonction de tri finale.

La fonction `min_list` qui permet d'avoir le minimum de la liste.

Enfin `suppr_doublons` qui renvoie notre liste sans doublons.