



IEPSCF-Uccle.
Rue Gatti de Gamond 95
1180 Uccle Belgique

Fédération Wallonie-Bruxelles

Base de la programmation

Bachelier en informatique et systèmes

Introduction à la complexité

(Version 201803090900)

<Prénom1> <Nom1>
<Prénom2> <Nom2>

Chargé de cours : M. E.Derasse

Année académique 2016-2017

1. Objectif et spécifications

Objectif du projet

L'objectif de ce TP est d'établir la courbe de complexité (cas moyen et pire des cas) du tri rapide.

Objectif pédagogique

Les objectifs pédagogiques sont :

- Etablir et interpréter une courbe de complexité.
- Programmer un langage de haut niveau
- Implémenter/Utiliser une architecture en couche
- Utiliser un systèmes de gestuon de code en ligne

Spécifications techniques

L'architecture utilisée est l'architecture en couche.

- L'architecture du programme respecte la base de l'interface de programmation donnée.

Spécification pédagogique

Le travail se réalise par groupe de deux. Le dossier est déposé sur la plate-forme habituelle Moodle. Le code est déposé sur le git-up du groupe.

2. Stratégie de résolution

Expliquez sur la séquence de chiffre ci-dessous le principe de résolution de votre programme.

20 10 15 10 12 5 8 12 9 4 1

3. Découpage fonctionnel (Interface de programmation)

Ce tableau regroupe l'ensemble des fonctions que vous utilisez pour résoudre votre problème. Pour chaque fonction vous indiquez les paramètres (entrée/sortie) ainsi que les typages. Que font les différentes fonctions ?

Compléter le tableau d'interface de programmation

Couche présentation
Menu () # Cette fonction permet d'afficher les différentes fonctions offertes
SelectMenuInit () # Cette fonction permet de donner un nom au fichier qui contiendra le tableau ainsi que le nombre d'élément contenu dans ce dernier.
SelectMenuCharger (maTable : Tableau de [1 a n] type entier) # Cette fonction permet de charger un fichier existant contenant des valeurs générées précédemment et d'en extraire un nombre d'éléments pour les mettre dans le tableau
SelectMenuSauvegarde(maTable: Tableau de [1 a n] type entier) # Cette fonction prend un tableau trié en entrée et le met dans un fichier.
xxx
Couche logique
TriRapide(maTable):
Couche donnée
initTab(fichier:chaîne de caractères,nb:entier) # Génération des nombres entiers aléatoires et sauvegarde dans un fichier
loadTab(maTable: Tableau de [1 a n] type entier ,fichier,nb:entier) # Chargement du fichier dans un tableau en mémoire
saveTab(maTable: Tableau de [1 a n] type entier,fichier) # Sauvegarde du tableau en mémoire dans un fichier

4. Diagramme fonctionnel

Réaliser le diagramme fonctionnel du programme.

4. Courbe de complexité

- Réaliser les courbes de complexité pour le cas moyen et le pire des cas.
- Comparez ensuite vos courbe avec les courbes théoriques. Sont-elles cohérentes ?
- Si votre courbe ne correspond pas à la courbe théorique, comme l'expliquez-vous ?
- Estimez théoriquement la complexité du tri.

5. Repositories gitHub

Indiquez le repositories gitHub de votre groupe utilisé pour ce projet.

5. Bibliographie