

HOW TO

Compilation et exécution sur les données données en annexe

- Ces deux commandes compilent et exécutent le projet sur les données du fichier "./Donnees_test" et met les analyses dans le fichier "./analyses" et les diagrammes Gnuplot dans "./Diagramme gnuplot"

```
cd ./code  
make plot
```

- Exécution sans générer les diagrammes Gnuplot

```
cd ./code  
make exec_all
```

Compilation

- Ces commandes permettent de compiler tous le projet (modules et la main et génèrent un exécutable "./main.exe" dans le répertoire "./code"

```
cd ./code  
make main
```

Execution

- Ces commandes permettent d'exécuter le "test.exe" avec 3 paramètres qui sont, le premier fichier de données, le deuxième fichier de données et un fichier où on met l'analyse des temps

```
cd ./code  
./main.exe ${fichier_donnee1} ${fichier_donnee2} ${fichier_analyse}
```

Nettoyage

Nettoyage des executables

- Suppression des .cmi, .cmo, .exe

```
cd ./code
make clean
```

Nettoyage de toutes les données

- Suppression des .cmi, .cmo, .exe, fichiers d'analyses, diagrammes Gnuplot

```
cd ./code
make clean_all
```

Tests

- Test sur les 3 fonctions sousChainesCommunes, sousChainesCommunesComprimee et sousChainesCommunesDynamique, il suffit d'executer l'un des 3 scripts

1 - sousChainesCommunes :

```
cd ./code
make arbre_suffixe_test
./arbre_suffixe_test.exe
```

2 - sousChainesCommunesComprimee :

```
cd ./code
make arbre_suffixe_comprimee_test
./arbre_suffixe_comprimee_test.exe
```

3 - sousChainesCommunesDynamique :

```
cd ./code
make programmation_dynamique_test
./programmation_dynamique_test.exe
```

Fichiers DOT

- Pour dessiner un arbre des suffixes, tapez les commandes suivantes. Le fichier .dot genere est dans le repertoire "./dot"

```
cd ./code
make dot_test
./dot_test.exe
```

