

ABD - TP 07

Auteurs

- Anne-Sophie Saint-Omer
- Thomas Bernard

Description

Écrire un programme en pseudo-code qui détermine les anagrammes des mots lus dans un dictionnaire. Comme fichier dictionnaire on utilisera le fichier des mots de la langue anglaise dico.txt.

Objectifs

- On veut tous les mots en résultats, même ceux qui n'ont pas d'anagramme.
- On ne conserve que les mots du dictionnaire qui ont un autre mot comme anagramme.

Algorithme

Le principe de l'algorithme consiste à associer à un mot une **signature**. Cette signature se compose des lettres du mot, triées dans l'ordre alphabétique. Ainsi, **deux anagrammes auront la même signature**. En utilisant cette signature comme clé d'une **map**, on lui associe comme valeur la liste des mots à l'origine de cette signature.

Algorithme en pseudo-code :

```
Map<String, List<String>> map

Pour chaque mot du dictionnaire:
    signature = lettres du mot triées dans l'ordre alphabétique
    map[signature].append(mot)
```

Recherche d'anagrammes :

Une fois la map remplie avec les mots du dictionnaire, il est possible de rechercher les anagrammes d'un mot donné.

`map[mot.sort()]` retourne en effet la liste des anagrammes du `mot` passé en paramètre.

Exemple

Exemples de signatures :

Mot	Signature
LOOP	LOOP
POOL	LOOP
POLO	LOOP
STOP	OPST
POST	OPST

Map après traitement de l’algorithme :

Signature	Mots
LOOP	[LOOP, POOL, POLO]
OPST	[STOP, POST]