

Projet de programmation : Recherche d'informations

1 Description du projet

Nous allons, dans le cadre de ce projet, développer une application permettant de charger une base de donnée minimaliste et de chercher des informations dans celle-ci.

La base de donnée contiendra des médicaments ainsi que leurs effets indésirables. Elle sera présentée sous la forme :

nom du médicament₁ : effets indésirables notoires : effet₁, effet₂, effet₃ et effet₄.
 nom du médicament₂ : effets indésirables notoires : effet₁, et effet₂.

On note que tous les médicaments n'ont pas le même nombre d'effets secondaires.

2 Cahier des charges

Voici les fonctionnalités minimales que nous attendons de vos programmes.

1. Chargement des données : les données, contenues dans un fichier format `txt`, devront être chargées dans le programme. Vous pourrez utiliser une `hashmap` pour stocker les différents médicaments ainsi que la liste de leurs effets secondaires. Lors de l'étape de *parsing* de chaque entrée, il faudra retirer le "bruit" de la liste des effets secondaires, en reconnaissant les caractères de séparation ainsi que les mots sans importance.
2. Création de l'histogramme : un histogramme des effets secondaires existant devra être créé.
3. Visualisation des données : la visualisation de l'histogramme ainsi que la liste des médicaments présents devront pouvoir être visualisés.
4. Recherche d'informations :
 - liste des médicaments ayant un effet secondaire donné : en entrant un effet secondaire, voir en choisissant parmi la liste des effets secondaires existants, l'utilisateur doit voir apparaître tous les médicaments correspondant.
 - liste des médicaments partageant avec un autre entré par l'utilisateur le plus d'effets secondaires possibles : en entrant un nouveau médicament au même format que ceux présents dans la base de données, l'utilisateur aura accès à tous les médicaments partageant un ou plusieurs effets secondaires avec (facultatif : l'utilisateur peut choisir le nombre d'effets secondaires partagés).

5. Calcul du rappel et de la précision : le rappel et la précision permettront de calculer l'efficacité de votre algorithme de recherche. Le rappel permet de connaître la marge d'erreur de l'algorithme et la précision la présence de bruit dans la réponse donnée.
6. (facultatif) Donner la possibilité à l'utilisateur de rajouter des médicaments dans la base de donnée sans recharger le programme, en saisissant une nouvelle entrée directement pendant le fonctionnement de l'application.
7. (facultatif) Création d'une interface graphique : vous pourrez créer une interface graphique à votre programme, en utilisant par exemple **Qtcreator**.

Vous travaillerez en binôme de **même groupe** sur ce projet.

Le code devra respecter quelques normes :

- une séparation des classes en headers et code C++,
- la fonction main dans un fichier n'implémentant pas déjà une classe,
- des noms explicites, aussi bien pour les classes que les variables ou les fonctions,
- des commentaires pertinents.

Le rapport se composera :

- d'une page de garde,
- d'un sommaire,
- d'une brève description du projet (évitez le copier/coller),
- du cahier des charges,
- d'une partie d'explication du code (il n'est pas question de voir de la paraphrase de votre code : n'expliquez que ce qui a besoin de l'être),
- d'une partie test de l'application où vous décrierez l'utilisation normale de l'application ainsi que les potentiels cas limites qui font planter celle-ci,
- d'une conclusion.

3 Dates importantes

Le projet sera à rendre le **13 avril**. Vous enverrez une unique archive à votre chargé de TD. Le mail aura pour nom [POO PROJET MIAGE] et l'archive **nom_prenom.zip**. Elle devra contenir :

- le rapport au format PDF,
- les sources de votre programme,
- un fichier **Makefile**, permettant à votre enseignant compiler vos sources,
- un **README** expliquant l'utilisation de votre programme,
- potentiellement, tout fichier de test nécessaire.
- **Ne mettez pas d'exécutable dans vos archives.**