

Les molécules

TP 4

Objectifs

- ★ Manipulez les conteneurs
- ★ Manipulez les itérateurs

Préparation du TP

- Cliquez sur le lien écrit dans moodle, votre dépôt git sera à l'adresse <https://github.com/Polytech-Sorbonne-Cpp-2020/tp2-LOGIN> où **LOGIN** est votre login sur GitHub
- Cloner votre répertoire sur votre compte

```
git clone https://git
```

 l'adresse qui vous a été attribué.
- Pendant le TP n'oubliez pas de comiter régulièrement
- N'oubliez pas de pusher l'ensemble
- Un ensemble de tests unitaires est fourni dans le répertoire `tests`, vous pouvez les compiler avec `make testcase`. Les tests utilise la bibliothèque `catch2`, pour en savoir plus <https://github.com/catchorg/Catch2>.

Concept

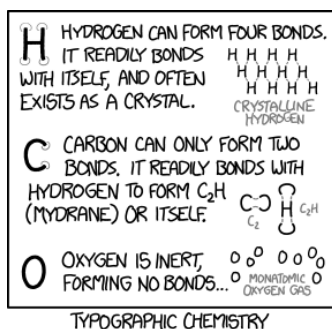


Figure 1: These are all sans-serif compounds. Serif compounds are dramatically different and usually much more reactive.xkcd

1 Normalisation de la forme d'une molécule

Une molécule chimique est un assemblage d'éléments individuels (atome). Chaque molécule peut être écrite sous une forme éclatée : H-O-H, O-Nb-O-Nb-O

ou bien sous forme normale : H₂O, Nb₂O₃

La forme éclatée est donc une suite d'atomes séparés par des tirets. Sous la forme normale, les atomes sont réunis selon leur nom, et classés par ordre alphabétique. Le but de l'exercice est d'écrire une fonction permettant de transformer une forme éclatée en une forme normale.

1. Écrire une classe `Molecule` qui possède un attribut contenant l'ensemble de ses atomes sous forme éclatée.
2. Écrire le constructeur créant, à partir d'une chaîne de caractères représentant une molécule sous forme éclatée, un vecteur de `std::string` représentant la même forme éclatée. Par exemple, la fonction donnera le vecteur `[O;Nb;O;Nb;O]` à partir de la molécule `O-Nb-O-Nb-O`. Tester cette construction avec les molécules du fichier `"molecules.txt"`.
3. Écrire une méthode `std::string normalise() const` qui crée, à partir du vecteur d'atome, la forme normale de la molécule. Cette fonction devra donc :
 - trier les atomes selon leur nom
 - compter les occurrences de chaque atome dans la molécule
4. Tester cette fonction avec les molécules du fichier `"molecules.txt"`. Afficher le résultat de la forme normalisée de chacune des molécules dans un fichier `res_normalisation.txt`.

2 Normalisation des réactions chimiques

On souhaite maintenant normaliser l'écriture de réactions chimiques données sous forme éclatée. Par exemple, la réaction :

`Na-O-Na + H-O-H → Na-O-H + Na-H-O` sera normalisée en :

`Na2O + H2O → 2 NaOH`

1. Écrire la classe `Reaction` qui a deux attributs correspondant à l'ensemble des atomes de gauche et l'ensemble des atomes à droite de la réaction.
2. Écrire le constructeur créant, à partir d'une chaîne de caractères représentant une équation sous forme éclatée, deux vecteurs "gauche" et "droite" contenant respectivement la liste des molécules apparaissant à gauche et à droite de la flèche.
3. Écrire une méthode `std::string normalise() const` transformant une réaction donnée sous forme éclatée en une réaction normalisée. Tester votre fonction sur le fichier `"reactions.txt"`. Afficher le résultat de la forme normalisée de chacune des molécules dans un fichier `res_reactions.txt`.

Bonus

Surcharger l'opérateur de flux sortant (`operator<<`) pour les deux classes précédentes.