**Introduction :**

Le projet est réalisé sous la forme d’une classe. Le seul module nécessaire est maplotlib.

Le fichier python à exécuter se nomme projet.py.

L’appel de cette classe muni de deux paramètres (l’atome et l’action) provoquera un retour adapté.

L’appel d’exemple est réalisé à la ligne 619 du programme (Atomistique(5, "casesQuantiques"))

S

**Fonction \_\_init\_\_ :**

C’est la fonction basique d’une classe servant à définir ou à exécuter ce que l’on veut. Toutes les variables utiles au programme sont définies ici et en fonction de l’action choisie, l’appel d’une des fonctions est réalisé et affiché.

**Fonction nb\_electrons :**

L’atome peut être donné sous forme du nombre atomique ou bien comme une chaine de caractère. La fonction s’occupe de différencier les deux cas. Dans le cas d’un numéro atomique, l’exécution est plutôt simple et définit simplement le nombre d’électron comme le numéro atomique indiqué.

Dans le cas d’une chaine de caractère, le fait que le nom de l’atome soit composé d’une ou deux lettres est pris en compte pour le parcours de la chaine

Dans le cas d’une chaine de caractère avec ionisation, les deux cas d’écriture de l’ionisation sont différenciés. En fonction de cela, le parcours de la chaine de caractère est fait différemment afin de prendre en compte les ions.

Il y’a aussi une variable nom qui est retournée pour m’aider dans d’autres fonctions.

**Fonction structureComplete :**

*(Self.nb\_electrons\_r est le nombre d’électron pur sans chaine de caractère.)*

Fonction la plus simple du programme, une boucle est réalisée par saut de deux sur le nombre quantique de l’ordre de base jusqu’à atteindre le nombre d’électrons, et vérifie quelle est la sous couche afin de faire l’affectation nécessaire.

**Fonction structureAllegee :**

Cette fonction renvoie la structure allégée avec le gaz noble correspondant au nombre d’électrons de l’atome simplement en se plaçant à partir du nombre d’électrons du gaz noble en question puis en écrivant le reste, tout en prenant compte des exceptions à la règle de Klechkowski.

**Fonction structureValence :**

Cette fonction, en parcourant la structure de l’atome, joue du fait que le caractère soit oui ou non dans l’alphabet afin de vérifier l’élément qui suit que sera donc le nombre présent dans la sous- couche, permettant de vérifier si elle est complète. Même raisonnement pour le nombre quantique principal en adaptant le principe.

**Fonction casesQuantiques :**

Fonction la plus longue du programme, pas forcément la plus compliquée, mais la plus dense. D’abord, le tracé du tableau est réalisé en reprenant les dimensions fournies dans les consignes (quelquefois adaptées un petit peu). La hauteur du tableau est divisée pour pouvoir afficher chaque nombre quantique de façon équidistante, et les cases quantiques sont simplement dessinées avec une longueur de 1, donc plutôt simple à tracer avec quelques boucles pour faire des carrés.

Ensuite, pour le tracé des flèches, le remplissage des couches est vérifié au cas par cas et si remplies, la flèche est tracée à l’endroit adapté.

**Conclusion :**

Ce projet m’a vraiment permis de me replonger dans la conception et la réalisation d’un programme plutôt complexe, ce fut la première fois que je liais mes compétences en chimie et en programmation pour réaliser quelque chose de satisfaisant. Le fait qu’il y’ait un rendu visuel est très gratifiant et je suis plutôt fier de ma réalisation dans l’ensemble.

J’ai cependant quelques points que j’aurais aimé entreprendre différemment :

* Au commencement de la conception de ma fonction casesQuantiques, je n’avais pas pensé à faire mon programme de sorte que le tracé des cases soit fait en fonctions de la structure de l’atome. Mon programme fait d’abord le tracé, puis place les flèches, donc les cases vides reste présentes. Je trouve le rendu visuel peut être mieux pour comprendre la structure mais moins gratifiant du coté programmation, un affichage s’adaptant à la structure aurait apporté surement une complexité temporelle plus faible, ainsi qu’une réflexion plus importante de ma part.
* La fonction nb\_éléctrons, ayant été faite en première, est selon moi plus complexe qu’elle n’aurait pu l’être. Je regrette ce manque d’optimisation pour cette fonction.
* La réalisation d’une légende avec des couleurs pour les flèches, je n’ai pas trouvé de moyen de l’implémenter dans ma fonction casesQuantiques.