Gestionnaire de mémoire

...

Clément BUON
Fanny BACHEY
Tifanny NGUYEN

Allocation/Gestion de la mémoire :

Notre mémoire est une bande sous la forme d'une liste et sera fixe. Elle permet de stocker des variables de 32 bits. (0 à 31 pour la première variable, 32 à 63 pour la deuxieme, 64 à 96 etc etc.)

32 bits car : 30 bâtons entourés de 0 (pour 29, l'entier maximal que nous prenons)

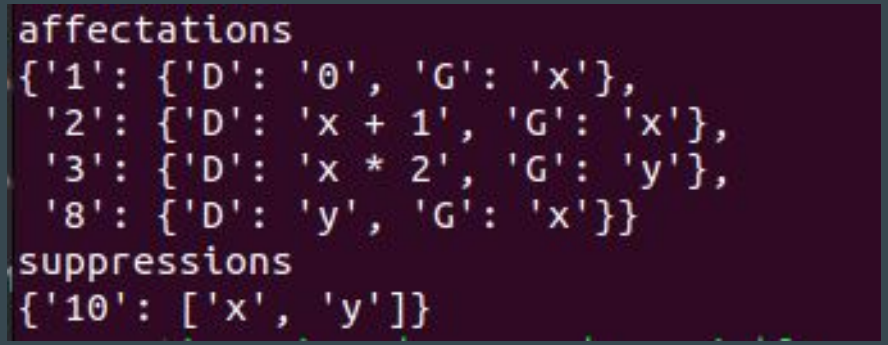
Lorsque la variable change de valeur, la bande s'actualise.

La composition de la mémoire est telle que les constantes occupent le début, suivies par les variables et enfin la mémoire vive.

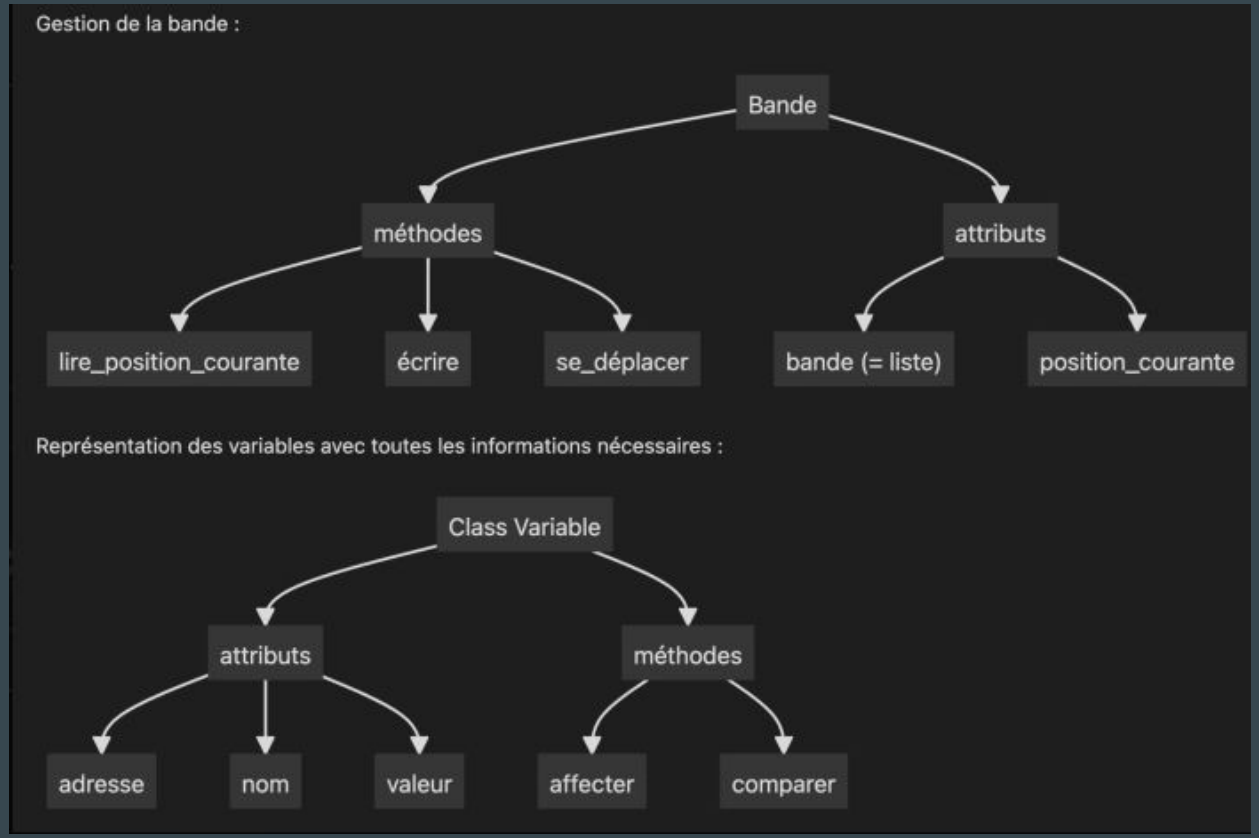
Représentation de la bande sur 32 bits.

Notre input :

Dictionnaire d'affectation et de suppression généré par le groupe 2



Notre module :



- On ajoute la classe constante héritée de la classe variable, cela permet d'allouer un espace pour celles-ci.
- Nous avons également ajouté plus de méthodes à bande.
- MODIFICATION MAJEURE : on ajoute une classe memory_manager qui nous permet de gérer les constantes et les variables.
- Enfin : un script principal met en pratique toutes ces classes =>

groupe3_gestionMemoire.py

Notre module :

- Parcours le script pour ajouter les constantes à la mémoire.
- Parcours une nouvelle fois pour y ajouter les variables.
- Ajout d'un module qui permet de récupérer l'adresse d'une variable (demandé par le groupe 2)
- On ne met pas seulement en mémoire les variables, il y a aussi les constantes et les opérateurs. De même, il pourrait être utile d'avoir une partie de mémoire "vive" pour le groupe 4.
- La mémoire est donc "divisée" en 3 parties : une partie pour les variables, une autre pour les constantes et enfin une pour la mémoire vive

Sortie que l'on renvoie au groupe 4 :

```
Étape n°1:
CONST_0 : adresse 0
x : adresse 32

Étape n°2:
CONST_0 : adresse 0
x : adresse 32
CONST_1 : adresse 64

Étape n°3:
CONST_0 : adresse 0
x : adresse 32
CONST_1 : adresse 64
CONST_2 : adresse 96
y : adresse 128
```

Voici ce qui est renvoyé à chaque itération. Mais d'autres méthodes permettent de récupérer d'autre informations isolées comme l'adresse d'une seule variable.

Conclusion :

C’était difficile.