

Algorithmique – Programmation Objet – Python

TD n°3

Licence Informatique 2ème année
Université de Nice Sophia Antipolis

Les exercices qui suivent reprennent le cours magistral d'introduction à la programmation orientée objet. Nous concevrons des classes (et leurs méthodes) pour permettre à un programme d'effectuer des opérations arithmétiques précises sur des nombres de taille illimitée. Cela pourrait être utile, par exemple, pour effectuer des calculs pour la cryptographie à clé publique, ou pour résoudre des systèmes d'équations mal conditionnés.

1 Nombres Naturels

Concevez une classe `Naturel` des objets qui représentent des nombres naturels de n'importe quelle taille. Donnez la ou les structures de données utilisées pour représenter un nombre naturel ainsi que les opérations (méthodes) qui peuvent être appliquées aux objets de cette classe.

Suggestion Utilisez une représentation décimale des nombres, similaire à celle utilisée quand on fait des calculs arithmétiques à la main sur papier.

2 Un peu d'arithmétique

1. Écrivez l'algorithme de l'addition pour la classe que vous venez de concevoir. Inspirez-vous de l'algorithme que vous avez appris à l'école primaire et qu'on utilise pour faire les calculs à la main.
2. Analysez l'algorithme de l'addition: quel est sa complexité en temps en fonction des deux nombres sommés ? Quel est sa complexité en espace ?
3. Écrivez l'algorithme pour comparer deux nombres : supposez que vous deviez réaliser une méthode `entier compare(Naturel self, Naturel n)` qui renvoie 0 si le naturel représenté par l'objet est égal à n , -1 s'il est plus petit et $+1$ s'il est plus grand.
4. Faites l'analyse de la complexité de l'algorithme de comparaison : traitez le pire cas, le meilleur cas et le cas moyen ; discutez la pertinence de chacun des cas.
5. Proposez une représentation qui permette de réduire le nombre d'opérations de l'algorithme de l'addition d'un facteur de 100 par rapport à la représentation que vous avez utilisé jusqu'à ici.

3 Nombres Rationnels

1. Concevez une classe `Rationnel` pour représenter des nombres rationnels. Cherchez à (ré)utiliser la classe `Naturel` que vous avez conçu.
2. Donnez une ébauche de l'algorithme de l'addition pour cette nouvelle classe. Discutez sa complexité.
3. Donnez une ébauche de l'algorithme de comparaison.