

TD de structures de données

Polytech'Lille IMA3

1. Ajouter au type `Eleve` son numéro NIP unique, sous la forme d'une chaîne de caractères (à 8 chiffres).
2. Montrer que pour optimiser l'accès aux informations d'un étudiant donné par son numéro NIP, il est judicieux que la classe soit ordonnée selon ce critère.
3. On souhaite donner la possibilité d'ajouter/supprimer des élèves dans une classe, dans la limite de `MAX_ELEVES`. Modifier la structure de données `Classe` pour offrir cette possibilité.
4. Etant donné une classe ordonnée par numéro NIP et un numéro NIP, écrire la fonction qui renvoie l'indice de rangement de l'étudiant correspondant dans la classe¹.
5. Ecrire l'opération d'ajout d'un élève (donné avec toutes ses informations) par insertion séquentielle ordonnée, afin de maintenir l'ordre des élèves par numéro de NIP et profiter de l'accès efficace précédent.

1. On supposera disposer d'une relation d'ordre alphanumérique sur les chaînes de caractères, applicable aux NIP, notée $<$, $=$, $>$. Cette relation d'ordre est en général fournie par les langages, notamment en C : fonction `strcmp` ("string compare") de la bibliothèque `<string.h>` de manipulation de chaînes.