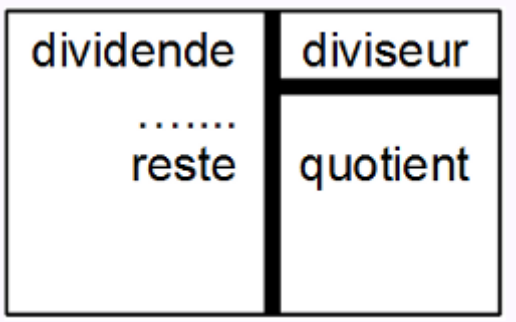
Document élève

Séquence
"Codage de l'information"

Activité :

Convertir un nombre entier (base 10) dans les bases decimale, hexadecimale et binaire

Pour mener à bien ce travail pratique, il a été découpé pour ne pas avoir la totalité du code à développer. Il est important de ne pas regarder les questions suivantes (qui apportent des élémentes de réponse) avant d'avoir traité la question en cours. En cas de difficultés, il vous est permis de consulter les aides puis le cas échéant le corrigé uniquement après en avoir fait la demande à l'enseignant.



Rappel : la commande CRTL + Q permet dans NotePad++ de commenter une à plusieurs lignes à la fois (très utile pour débugger un programme). Le logiciel Thonny peut également rendre de nombreux services pour comprendre comment s'exécute le code en mode débugger.

1. Convertir manuellement avec la division euclidienne, le nombre $123_{(10)}$ dans les bases 10, 16 et 2.
2. Chercher l'algorithme d'implémentation de l'opération précédente pour l'une des bases (par exemple la base 10 qui est plus simple).

   Conventions : On nommera les variables « dvd » pour le dividende, « dvs » pour le diviseur, « quo » pour le quotient et « rst » pour le reste.
   Aide : On utilisera une liste « chiffres » pour stocker les « restes » de la division (pour rappel, les « restes » sont forcément des symboles appartenant à la base).
   Corrigé : Élément de corrigé (algorithme.txt)

3. Implémenter en langage Python l'algorithme fourni (algorithme.txt), tout en commentant le code.

   Aide : Utiliser la méthode « reverse() » pour inverser les éléments de la liste.

   Aide niveau 1 → algo1-elève-elève
   Corrigé → algo1-corrige.py

4. Construire elementBase10 grâce à une boucle For pour que la liste se remplisse automatiquement (elle doit donc ressembler à ceci « elementBase10=[0,1,2,3,4,5,6,7,8,9] » sans avoir saisi les éléments de 0 à 9 à la main).

Pour l'instant votre code converti 123(10) en base 10, ce qui donne bien évidemment 123 ! Cela prouve au moins que sur quelle que chose de simple, cela fonctionne.

5. Modifier le code pour que 123 ne soit pas écrit à la main dans le code mais saisi au clavier. Valider le code pour différentes saisies.
   Aide : utiliser fonction « input() » et la fonctionnalité « f string » de Python.

6. Modifier le code pour que celui soit fonctionnel sur les bases 2 et 16 en plus de la base 10. Un message d'accueil proposera à l'utilisateur de choisir le mode de conversion (base 2, 10 ou 16). Il faudra saisir « d » pour la base 10 , « b » pour le base 2 et enfin « h » pour la base 16.
   Conseil : procéder par étapes en commençant à implémenter uniquement pour la base 2 (par exemple).