



DM 4

Informatique

James Donb

Simon Dauguet
simon.dauguet@gmail.com

Pour le Vendredi 25 Avril 2025

On renverra le code dans un fichier dont le nom devra être "DM4.Nom.Prénom.py" (attention, l'extension n'est pas toujours affiché par défaut sur les machines fonctionnant sous windows (c'est nul windows)).

On s'interdira, uniquement dans les deux prochaines questions, l'usage de la fonction `int()`.

L'agent secret James Domb 008 est parti précipitamment en laissant derrière-lui le fichier texte `DM4-image.txt` et l'image `DM4-image.bmp`. Saurez-vous découvrir le message qu'il voulait vous transmettre ?

1. Définir la fonction `hex2dec(v:str) -> int`, qui convertit le caractère hexadécimal `v` en entier. Par exemple, `hex2dec("7")` renverra 7 tandis que `hex2dec("a")` renverra 10.
2. En déduire une fonction `hexa2dec(val:str) -> int`, qui convertit un hexadécimal `val` en entier. Par exemple, `hexa2dec("7a")` renverra 122.
3. Définir la fonction `decodeTxt(txt:str) -> str`, qui convertit le texte `txt` composée d'une succession de code hexadécimal en une chaîne de caractères lisible. Par exemple, `decodeTxt("6d707369")` renverra "mpsi".
4. Le chiffrement de César consiste à décaler toutes les lettres d'un décalage de d . Par exemple, si $d = 5$, 'A' devient 'F', 'B' devient 'G', ..., 'U' devient 'Z', 'V' devient 'A', 'W' devient 'B', etc.

Proposer une fonction `lettreCesar(let:str, d:int) -> str`, qui prend une lettre majuscule ou un espace et un entier d , puis qui renvoie un espace si `let` en est un ou une lettre majuscule décalée de d si `let` était une lettre majuscule.

On rappelle que `ord('A')` = 65 et `%26` renvoie le reste de la division euclidienne par 26.

5. Définir la fonction `cesar(txt:str, d:int) -> str`, qui prend un texte `txt` ne comportant que des majuscules ou des espaces et décale chacune des lettres de d . Par exemple, `cesar("ABU VW", 5)` renverra "FGZ AB".
6. Proposer alors une fonction `decodeCesar(txt:str, mot:str) -> list`, qui renvoie dans un tableau tous les décalages d pour lequel le texte `txt` contient `mot`.

En déduire la dernière phrase du fichier texte.

7. Proposer une fonction `bit(nb:int, n:int) -> int` qui renvoie le bit associé au poids 2^n d'un nombre `nb`. Par exemple, puisque $201 = \overline{11001001}^2 = 2^7 + 2^6 + 2^3 + 2^0$, `bit(201,5)` devra renvoyer 0 et `bit(201,3)` devra renvoyer 1.
8. Proposer une fonction `selectImage(mat:list, f:'function'=(lambda x : x)) -> list`, qui à partir d'une matrice composée d'entiers codés sur un octet, renvoie une nouvelle matrice de même taille composée de l'image par la fonction `f` de chacun de ces coefficients.

Par exemple, si $M = \begin{bmatrix} 21 & 123 & 12 \\ 43 & 44 & 181 \end{bmatrix}$, alors `selectImage(M, lambda x : x%2)` renverra $\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$.

9. Proposer une fonction `sautImage(mat:list, s:int)-> list`, qui renvoie une nouvelle matrice composée de toutes les lignes espacées de s indices à partir de la première ligne.

M=[[21, 122, 223],
[243, 144, 45],
[24, 125, 226],
[246, 147, 48],
[27, 128, 229]]

Par exemple, si `sautImage(M, 2)` renverra

[[21, 122, 223],
[24, 125, 226],
[27, 128, 229]]

10. Proposer une fonction `recomposeImage(mat:list, r:int)-> list` qui :

- Créé une décomposition binaire d'un nombre en partant de l'un des éléments de `mat` et en sélectionnant en comptant par pas de r -termes.
- Complète les décompositions binaires par des 0 à droite jusqu'à obtenir un code binaire sur un octet.
- Renvoie la liste des nombres ainsi codés.

Par exemple, `recomposeImage([[1,1,0,1,0,0,0,0,1]], 3)` renverra `[[192,128,32]]` car $192 = \overline{1100\ 0000}^2$ (on sélectionne les éléments d'indices 0, 3 et 6, donc [1,1,0], et on complète par cinq 0), $128 = \overline{1000\ 0000}^2$ (on sélectionne les éléments d'indices 1, 4, 7, donc [1,0,0], et on complète avec cinq 0) et $32 = \overline{0010\ 0000}^2$ (on sélectionne les éléments d'indices 2, 5, 8, donc [0,0,1], et on complète avec cinq 0).

11. Appliquer ces dernières fonctions à la matrice représentant l'image puis sauvegarder la matrice ainsi décodée afin d'obtenir le message caché par l'agent 001 à votre intention (message à préciser).