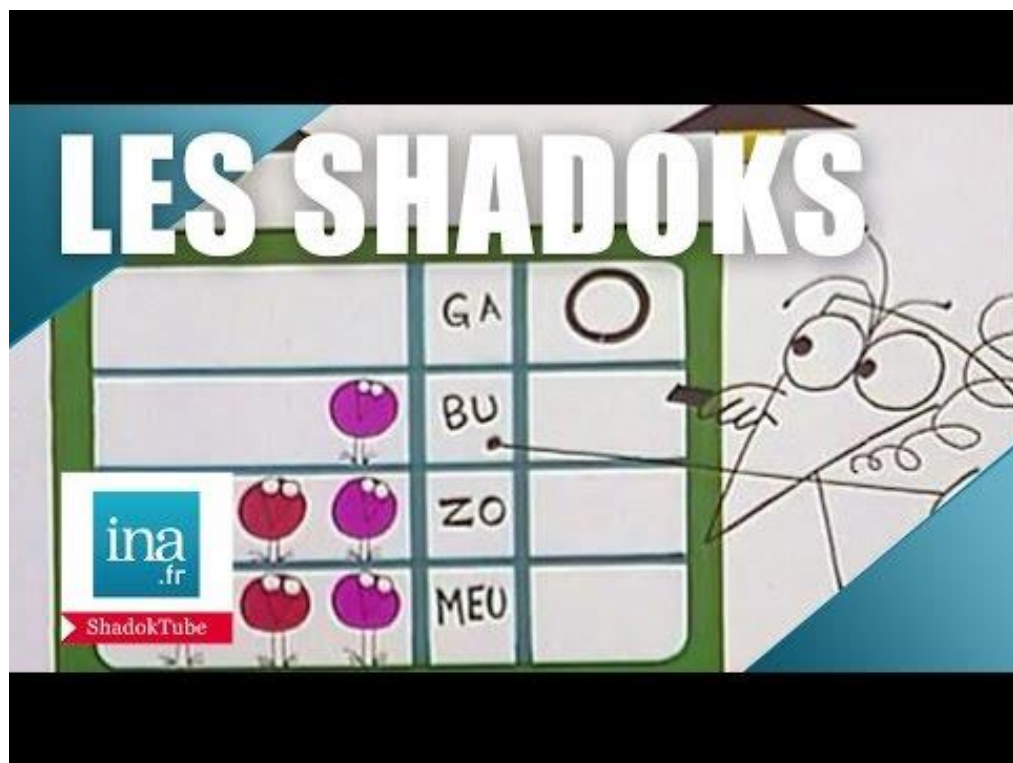


Compte Rendu du Mini-Projet d'ISN

1) Présentation du sujet

Nous avons comme demande que lors de l'entrée d'un nombre en base 10, le programme retourne le nombre en base Shadoks, comme montré dans cette vidéo :



Nous pouvions aussi, si nous le voulions faire la réciproque du programme, c'est-à-dire lors de l'entrée d'un nombre en base Shadoks, le programme devait retourner le nombre en base 10. Mais je n'ai préféré ne pas la faire et être sûr que le programme de base fonctionne.

Le programme doit être codé grâce à l'idle de Python 3 donc dans le langage de programmation Python. Nous avons eu près de 4 semaines pour finir ce projet.

2) Première essai

Dans un premier temps j'ai écrit un semblant d'algorithme au brouillon pour exposer mes idées et voilà ce que ça donne :

```
Entrez nombre -> N
N=S
def shadok(Chiffre)
    si chiffre== 0 ;
        sha= "GA"
    si chiffre== 1 ;
        sha= "BU"
    si chiffre== 2 ;
        sha= "ZO"
    si chiffre== 3 ;
        sha= "MEU"
Tant que != 0
    int(N//4) -> S
    N%4-> R
    sha + shadok(R) - > sha
Afficher (sha)
```

Après avoir fait cet algorithme en Python et l'avoir essayé, il y'avait un problème. En effet, lorsqu'on effectue une fonction, il faut utiliser la formule return, or je ne l'utilisais pas. Après avoir bloqué plusieurs minutes sur ce problème, j'ai réussi à le résoudre en ajoutant la fonction return et de ce fait en modifiant le code d'origine.

3) Les ajouts dans Python

```
1 n=eval(input("Entrez un nombre en base 10, à transformer en base Shadoks : "))
2 s=n # Je garde la variable n pour la phrase de fin
3 r=122 # Il fallait initialiser r avec un chiffre différent de 0
4 sha = str("") # Il faut indiquer qu'il s'agit de chaines de caractères
5 shs = str("") # Il faut indiquer qu'il s'agit de chaines de caractères
6
7 def shadok(chiffre):
8     if chiffre==0 :
9         return "Ga"
10    if chiffre==1 :
11        return "Bu"
12    if chiffre==2 :
13        return "Zo"
14    if chiffre==3 :
15        return "Meu"
16
17 if s==0:
18     sha = "Ga"
19
20 while s!=0:
21     r=s%4
22     s=s//4 #Cela nous permet d'avoir un nombre entier
23     shs=shadok(r)
24     sha = shs+sha #Cela permet d'indiquer les nombres dans le bon ordre
25 print(n,"en base shadoks donne",sha)
```

Après quelques ajouts pour que le programme soit fonctionnel avec Python, on arrive à ce code. J'ai testé celui-ci et vérifié les résultats grâce au site dcode.fr, qui permet de faire les conversions rapidement.

www.dcode.fr/numeration-shadok

J'ai commencé par tester le programme par 0 et ERREUR, je n'avais pas planifié cette entrée, donc j'ai rajouté les lignes 17 et 18.

J'ai dû coup re-testé le programme pour 0, 2, 7, 46, 81,1234, et 456789 et celui-ci était fonctionnel à chaque fois.

4) Conclusion

Ce mini-projet m'a permis de tester mes capacités dans un temps limité, d'examiner mes connaissances et ma capacité à créer un algorithme complexe. Le tout en faisant face aux imprévus en m'y prenant à l'avance.