

Méthodes de Conception d'Algorithmes

C. TRABELSI & E. MARTINS & M. FRANÇOIS

TD3 -- Stratégie Gauche-Droite / Droite-Gauche

Avant propos

On parle de stratégie gauche-droite ou stratégie droite-gauche, lorsque l'on est face à un problème asymétrique : on ne prend pas le même chemin selon notre approche du problème.

Par exemple, on ne fait pas la même chose lorsqu'on désire laver une pile d'assiettes en commençant par celle du haut ou par celle du fond. De la même manière, lorsque vous tenez un guichet ce n'est pas la même chose de servir le dernier client arrivé ou le premier de la file d'attente.

Nous allons illustrer ce constat aujourd'hui en résolvant le problème du **Miroir d'un nombre**.

1 Problème du Miroir d'un nombre

Nous allons nous intéresser aujourd'hui à la fonction `miroir`, qui inverse l'ordre des chiffres des nombres positifs.

Par exemple le miroir du nombre 1234567 est 7654321.

Une définition plus formelle de la fonction `miroir` dit que, pour un entier $n \geq 0$, où $n = d_k d_{k-1} \dots d_1 d_0$ (en base 10), son image miroir est définie par :

$$\text{miroir}(n) = d_0 d_1 \dots d_{k-1} d_k$$

- Remplir le tableau suivant puis réponde aux questions :

Nombre n	<code>miroir(n)</code>	Nb chiffres de n ?
7654321		
10034		
45000		
1		
0		

- Est-ce que tous les nombres ont un miroir en :
 - Mathématiques ?
 - Informatique ?
- Est-ce que notre fonction `miroir` est involutive (i.e `miroir(miroy(n)) = n`) ?
Peut-on poser une convention afin qu'elle le devienne ?

EXERCICE 1. (Stratégie Gauche-Droite)

- 1. En parcourant votre nombre du chiffre de poids fort vers le chiffre de poids faible, écrivez sur papier l'algorithme qui permet de calculer le miroir d'un nombre. Vérifiez votre algorithme sur le nombre 81645.

Voici un exemple avec les différentes étapes :

$n = 1234$	$< - - - >$	$m = 0$
$n = 234$	$< - - - >$	$m = 1$
$n = 34$	$< - - - >$	$m = 21$
$n = 4$	$< - - - >$	$m = 321$
$n = 0$	$< - - - >$	$m = 4321$

Posez-vous les bonnes questions :

- comment sépare-t-on un chiffre des autres chiffres du nombre ?
- comment ajoute-t-on le chiffre récupéré au miroir temporaire ?
- comment passe-t-on à l'étape suivante ?
- de quelles informations intermédiaires a-t-on besoin pour répondre aux trois questions précédentes ?

- 2. Calculer la complexité en temps de l'algorithme :

	Valeur calculée	Landau	Appellation
dans le meilleur des cas			
dans le pire des cas			
en moyenne			

Remarque : la variable N représente la quantité de données, donc ici ce n'est pas la valeur des nombres mais la **quantité de chiffres dans un nombre**.

EXERCICE 2. (Stratégie Droite-Gauche)

- 1. En parcourant votre nombre du chiffre de poids faible vers le chiffre de poids fort, écrivez sur papier l'algorithme qui permet de calculer le miroir d'un nombre. Vérifiez votre algorithme sur le nombre 81645.

Voici un exemple avec les différentes étapes :

$n = 1234$	$< - - - >$	$m = 0$
$n = 123$	$< - - - >$	$m = 4$
$n = 12$	$< - - - >$	$m = 43$
$n = 1$	$< - - - >$	$m = 432$
$n = 0$	$< - - - >$	$m = 4321$

- 2. Calculer la complexité en temps de cet algorithme :

	Valeur calculée	Landau	Appellation
dans le meilleur des cas			
dans le pire des cas			
en moyenne			

Remarque : les complexités proposées sont vraies pour une implémentation respectant parfaitement l'algorithme donné. Pensez à éviter les calculs inutiles le plus possible.

- 3. Comparer les deux méthodes et conclure ?

Remarque : (Complexité asymptotique et Complexité usuelle)

On a vu que les deux algorithmes ont des complexités asymptotiques similaires, cependant pour des valeurs réalistes de N (entre 1 et 200 chiffres) la valeur du coefficient proportionnel devient non négligeable.

2 Mise en œuvre

EXERCICE 3. (Langage C)

- 1. Écrire en C le programme qui :
- tant qu'il y a des nombres entiers **non signés** sur le flux d'entrée,
 - calcule le miroir du nombre,
 - puis affiche "MIROIR(XXX) = YYY",
- où XXX est le nombre récupéré et YYY son miroir et revient à la ligne.
Voilà à quoi peut ressembler l'exécution du programme :

```
$ gcc -o EXEC test.c -Wall
$ ./EXEC
94
MIROIR(94) = 49
51068
MIROIR(51068) = 86015
-1452
ATTENTION : NOMBRE NEGATIF
888
MIROIR(888) = 888
1
MIROIR(1) = 1
$
```

- 2. Modifier votre programme pour qu'il puisse accepter uniquement les nombres composés de 6 chiffres, puis renvoi le miroir correspondant.
Voilà à quoi peut ressembler l'exécution du programme :

```
$ gcc -o EXEC test.c -Wall -lm
$ ./EXEC
524198
MIROIR(524198) = 891425
12587
ATTENTION : NOMBRE DE CHIRFFRES DIFFERENT DE 6
123456
MIROIR(123456) = 654321
5125478
ATTENTION : NOMBRE DE CHIRFFRES DIFFERENT DE 6
981204
MIROIR(981204) = 402189
$
```

- 3. (À faire chez vous) Écrire un programme qui calcule le miroir d'un nombre, lorsqu'il est représenté en binaire (*i.e.* base 2).