

TP1: Multiplication d'entiers et système PLaTon

L'objectif de cette séance est de s'exercer aux compétences suivantes :

- lire un algorithme (itératif ou récursif) en pseudo-code ;
- exécuter à la main un algorithme (itératif ou récursif) sur un exemple jouet ;
- évaluer de manière intuitive l'efficacité d'un algorithme ;
- écrire du code C avec des exigences strictes sur les entrées et sorties (PLaTon).

1 Multiplication d'entiers

L'objectif de ce TP est de comparer différents algorithmes de multiplication d'entiers afin d'évaluer leur efficacité. Pour étudier cette tâche d'un point de vue algorithmique, nous devons restreindre les opérations que nous pouvons utiliser.

Problème : multiplier deux entiers donnés en n'utilisant que les opérations suivantes :

1. la multiplication de deux chiffres (tables de multiplication) ;
2. l'addition '+' ;
3. la multiplication par 10 (et ses puissances 100, 1000, ...) ;
4. la division entière par 10 : quotient '/' et reste '%'.
La division entière est toujours représentée par '/'.

Exercice 1: Algorithme Naïf. Considérez l'algorithme suivant :

Algorithme 1 MultiplicationNaïve

Données: deux entiers positifs a et b

Résultat: le produit de a par b

```
1:  $acc \leftarrow 0$ 
2: for  $i$  de 0 à  $b - 1$  do
3:    $acc \leftarrow acc + a$ 
4: end for
5: return  $acc$ 
```

1. Exécutez à la main MULTIPLICATIONNAIVE pour les valeurs 32 et 13 et présentez le résultat de cette exécution de la manière la plus claire possible.

Pour cela, vous pouvez vous aider du tableau suivant (à remplir) pour représenter l'état des variables au cours des itérations de l'algorithme.

$i :$		0	1	2	3	4	5	6	7	8	9	10	11	12
$acc :$														

2. Est-il possible (raisonnable) d'exécuter à la main MULTIPLICATIONNAIVE pour des grandes valeurs de a et b , par exemple $a = 2342$ et $b = 3224$?

Combien de fois l'opération d'addition située à l'intérieur de la boucle serait-elle répétée ?

Exercice 2: Comme à l'école primaire. Par la suite, si a est un entier, on désigne par a_i le $(i + 1)$ -ème chiffre de a en partant de la droite de sorte que si a s'écrit avec n chiffres on a :

$$a = \sum_{i=0}^{n-1} a_i 10^i$$

Par exemple si $a = 2342$, on a $a_3 = 2$, $a_2 = 3$, $a_1 = 4$, $a_0 = 2$. Plus graphiquement, on peut dessiner le tableau suivant pour représenter les $\{a_i\}$:

i	3	2	1	0
a_i	2	3	4	2

Considérez l'algorithme suivant :

Algorithme 2 MultiplicationEcolePrimaire

Données: deux entiers positifs a et b , comportant (au plus) n chiffres

Résultat: le produit de a par b

```

1:  $p \leftarrow 0$ 
2: for  $i$  de 0 à  $n - 1$  do
3:    $acc \leftarrow 0, r \leftarrow 0$ 
4:   for  $j$  de 0 à  $n - 1$  do
5:      $x \leftarrow b_i \cdot a_j + r$  // multiplication de 2 chiffres + retenue précédente
6:      $r \leftarrow x/10$  // mémoriser la retenue pour l'itération suivante
7:      $acc \leftarrow acc + (x \% 10) \cdot 10^j$ 
8:   end for
9:    $acc \leftarrow acc + r \cdot 10^n$ 
10:   $p \leftarrow p + acc \cdot 10^i$ 
11: end for
12: return  $p$ 
```

1. Exécutez à la main MULTIPLICATIONECOLEPRIMAIRE pour les valeurs de 2342 et 3224 et présentez le résultat de cette exécution de la manière la plus claire possible.

Pour cela, vous pouvez vous aider du tableau suivant pour représenter l'état des variables au cours des itérations de l'algorithme.

i	3	2	1	0
a_i	2	3	4	2

i	3	2	1	0
b_i				

	init					reset					
i		0	0	0	0		1	1	1	1	
j		0	1	2	3		0	1	2	3	
x											
r	0										
acc	0										
p	0					9368					

	reset					reset					fin
i		2	2	2	2		3	3	3	3	
j		0	1	2	3		0	1	2	3	
x											
r											
acc											
p	56208					524608					7550608

2. Combien de fois l'opération de multiplication élémentaire de 2 chiffres située à l'intérieur de la double boucle est-elle répétée (lorsque $a = 2342$ et $b = 3224$) ? Comparez avec votre réponse à l'Exercice 2.

Est-il possible (raisonnable) d'exécuter à la main MULTIPLICATIONECOLEPRIMAIRE pour des grandes valeurs de a et b , par exemple avec des nombres comportant $n = 16$ chiffres ?

2 Exercices PLaTon

Comme décrit dans le premier cours (consultez les transparents pour plus de détails), nous utiliserons PLaTon pour tester certains des exercices obligatoires de chaque TP.

Cette section contient des descriptions des exercices, ainsi qu'un lien vers l'activité.

Bien qu'il y ait d'autres exercices sur la multiplication, c'est le moment idéal pour faire une pause et vous assurer que vous pouvez accéder à PLaTon et résoudre les exercices dans son format.

Remarque: si vous pensez que votre solution est correcte, mais que certains tests échouent, demandez à votre chargé de TP de vous aider à résoudre les problèmes.

Exercice 3: Accès PLaTon. Rendez-vous sur la page Moodle du cours et cliquez sur le lien « Accès PLaTon ».

Après avoir vérifié que vous avez accès au système, vous pouvez cliquer sur le lien suivant pour accéder à l'activité :

<https://platon.univ-eiffel.fr/player/activity/6abeec8c-a768-4c81-8d64-570fc4096844>

Exercice 4: Hello World (PLaTon). Écrivez un programme en C qui affiche "Hello World!".

Exercice 5: Somme (PLaTon). Écrivez un programme qui demande à l'utilisateur de saisir deux nombres entiers séparés par un espace, puis affiche leur somme.

Exercice 6: Carré (PLaTon). L'objectif de cet exercice est de s'entraîner à lire des entrées plus complexes, ce qui sera utile dans les exercices futurs. La tâche de notre programme consiste à demander à l'utilisateur plusieurs nombres entiers et à afficher leurs carrés.

3 Multiplication d'entiers (continuation)

Exercice 7: Vers l'algorithme de Karatsuba. Il est possible d'obtenir des algorithmes encore plus efficaces pour multiplier deux nombres. Ces algorithmes sont ceux qui sont implémentés dans nos ordinateurs.

Nous allons voir l'algorithme de Karatsuba¹ découvert en 1960. Il repose sur deux observations que nous exploiterons successivement pour améliorer l'efficacité de la multiplication.

Observation 1 Si a et b sont deux entiers comportant n chiffres, chacun peut être décomposé en deux parties de taille à peu près égale :

1. la partie droite comporte les $n/2$ chiffres situés le plus à droite du nombre ;
2. la partie gauche comporte les chiffres suivants.

La partie droite de a est désignée par a_d et sa partie gauche par a_g . La partie droite de b est désignée par b_d et sa partie gauche par b_g . Par exemple pour $a = 2342$ et $b = 3224$, nous avons :

$$a_g = 23, \quad a_d = 42, \quad b_g = 32, \quad b_d = 24.$$

On peut donc écrire :

$$a \cdot b = (a_g \cdot 10^{n/2} + a_d) \cdot (b_g \cdot 10^{n/2} + b_d)$$

En développant la partie droite de l'équation, on obtient :

$$a \cdot b = a_g \cdot b_g \cdot 10^{2 \cdot n/2} + a_g \cdot b_d \cdot 10^{n/2} + a_d \cdot b_g \cdot 10^{n/2} + a_d \cdot b_d$$

On observe dans la partie droite de l'équation que chacun des 4 termes est formé par :

- la multiplication de deux facteurs de taille inférieur à $n/2 + 1$ (par définition des décompositions de a et b) ;
- dont le résultat est multiplié par une puissance de 10 (et la multiplication par des puissance de 10 est facile car il s'agit d'un simple décalage de chiffres).

En factorisant, on peut simplifier l'équation précédente pour économiser une multiplication par une puissance de 10 :

$$a \cdot b = a_g \cdot b_g \cdot 10^{2 \cdot n/2} + (a_g \cdot b_d + a_d \cdot b_g) \cdot 10^{n/2} + a_d \cdot b_d$$

D'après cette équation on peut donc formuler l'algorithme récursif suivant pour effectuer le produit de a par b .

1. voir https://fr.wikipedia.org/wiki/Algorithme_de_Karatsuba

Algorithme 3 Mult1**Données:** deux entiers positifs a et b , comportant (au plus) n chiffres**Résultat:** retourne le produit de a par b

```

1: if  $n=1$  then
2:   return  $a \cdot b$                                 // Multiplication de deux chiffres
3: end if
4:  $a_d \leftarrow$  le nombre composé des  $n/2$  chiffres les plus à droite dans  $a$ 
5:  $a_g \leftarrow$  le nombre composé des chiffres suivants de  $a$ 
6:  $b_d \leftarrow$  le nombre composé des  $n/2$  chiffres les plus à droite dans  $b$ 
7:  $b_g \leftarrow$  le nombre composé des chiffres suivants de  $b$ 
8:  $p_1 \leftarrow \text{MULT1}(a_g, b_g, n/2 + n\%2)$ 
9:  $p_2 \leftarrow \text{MULT1}(a_g, b_d, n/2 + n\%2)$ 
10:  $p_3 \leftarrow \text{MULT1}(a_d, b_g, n/2 + n\%2)$ 
11:  $p_4 \leftarrow \text{MULT1}(a_d, b_d, n/2)$ 
12: return  $p_1 \cdot 10^{2 \cdot n/2} + (p_2 + p_3) \cdot 10^{n/2} + p_4$ 

```

1. Exécutez à la main MULT1 pour les valeurs de $a = 2342$ et $b = 3224$ et présentez le résultat de cette exécution de la manière la plus claire possible.
2. Combien de fois l'opération de multiplication élémentaire de 2 chiffres située à l'intérieur de la conditionnelle Si est-elle répétée (lorsque $a = 2342$ et $b = 3224$) ? Comparez avec votre réponse à l'Exercice 2.

Exercice 8: Algorithme de Karatsuba. La version efficace de l'algorithme de Karatsuba repose sur une seconde observation fondamentale.

Observation 2 Si a_d, a_g, b_g, b_d sont quatre nombres entiers quelconques, l'égalité suivante est vraie :

$$a_g \cdot b_g + a_d \cdot b_d - (a_g - a_d) \cdot (b_g - b_d) = a_g \cdot b_d + a_d \cdot b_g$$

D'après cette égalité, nous pouvons réécrire l'équation surlignée dans la partie précédente :

$$a \cdot b = a_g \cdot b_g \cdot 10^{2 \cdot n/2} + (a_g \cdot b_g + a_d \cdot b_d - (a_g - a_d) \cdot (b_g - b_d)) \cdot 10^{n/2} + a_d \cdot b_d$$

Cette nouvelle équation ne fait intervenir que 3 multiplications différentes au lieu de 4 précédemment. Cette différence peut sembler négligeable mais c'est bien elle qui conduit à une amélioration très significative de l'efficacité de la méthode.

Algorithme 4 Karatsuba**Données:** deux entiers positifs a et b , comportant (au plus) n chiffres**Résultat:** retourne le produit de a par b

```

1: if  $a$  et  $b$  sont négatifs then
2:   return KARATSUBA( $|a|, |b|, n$ )
3: else if un des deux nombres est négatif then
4:   return  $-$ KARATSUBA( $|a|, |b|, n$ )
5: end if
6: if  $n=1$  then
7:   return  $a \cdot b$ 
8: end if
9:  $a_d \leftarrow$  le nombre composé des  $n/2$  chiffres les plus à droite dans  $a$ 
10:  $a_g \leftarrow$  le nombre composé des chiffres suivants de  $a$ 
11:  $b_d \leftarrow$  le nombre composé des  $n/2$  chiffres les plus à droite dans  $b$ 
12:  $b_g \leftarrow$  le nombre composé des chiffres suivants de  $b$ 
13:  $p_1 \leftarrow$  KARATSUBA( $a_g, b_g, n/2 + n\%2$ )
14:  $p_2 \leftarrow$  KARATSUBA( $a_g - a_d, b_g - b_d, n/2 + n\%2$ )
15:  $p_3 \leftarrow$  KARATSUBA( $a_d, b_d, n/2$ )
16: return  $p_1 \cdot 10^{2 \cdot n/2} + (p_1 + p_3 - p_2) \cdot 10^{n/2} + p_3$ 

```

1. Exécutez à la main KARATSUBA pour les valeurs de $a = 2342$ et $b = 3224$ et présentez le résultat de cette exécution de la manière la plus claire possible.
2. Combien de fois l'opération de multiplication élémentaire de 2 chiffres située à l'intérieur de la conditionnelle est-elle répétée (lorsque $a = 2342$ et $b = 3224$) ? Comparez avec votre réponse votre réponse à la Question 3.2.
3. Est-il possible (raisonnable) d'exécuter à la main KARATSUBA pour des grandes valeurs de a et b , par exemple avec des nombres comportant $n = 16$ chiffres ? Combien de multiplications élémentaires seraient nécessaires dans ce cas ?
4. Pour aller encore un peu plus loin, essayez d'évaluer le nombre d'opérations nécessaires si a et b comportent chacun 2^k chiffres.
5. (Bonus) Implémentez une fonction (en C) pour multiplier deux nombres en utilisant l'algorithme de Karatsuba. Cette fonction doit prendre en entrée deux entiers donnés sous forme de chaînes de caractères, et retourner une chaîne représentant leur multiplication.