

Multiplicación de enteros grandes

Meza Rodriguez Fiorella Ivonne



Contenido

01

Problema

02

Diseño de
la solución

03

Complejidad

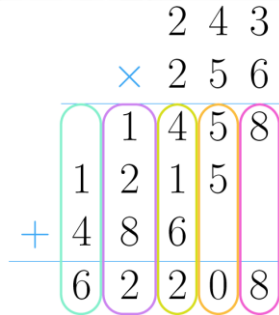
04

Código

01

Problema

El algoritmo tradicional consiste en colocar de forma vertical los números y realizar la multiplicación por cada dígito del número inferior con el número superior empezando desde las unidades, finalmente se obtiene el resultado sumando estos productos. Éste método tiene una complejidad de $O(n^2)$ y se observó que no funcionaba bien para números enteros grandes.



A hand-drawn illustration of a multiplication and addition problem. The numbers are arranged vertically. The multiplier is 243, and the multiplicand is 256. The multiplication is shown with a blue 'x' symbol. The result of the multiplication is 62208. The addition is shown with a blue '+' symbol. The digits are grouped into colored boxes: a green box for the first column (1, 4, 6), a purple box for the second column (1, 2, 2), a yellow box for the third column (4, 8, 2), an orange box for the fourth column (5, 1, 0), and a pink box for the fifth column (8, 5, 8). The final result 62208 is written at the bottom.

		2	4	3	
		×	2	5	6
		1	4	5	8
	1	2	1	5	
	4	8	6		
+	6	2	2	0	8

Orden de complejidad :
 $O(n^2)$

02 Diseño de la solución

En 1960 , el matemático ruso de 23 años llamado Anatoly Karatsuba descubre un algoritmo para multiplicar números enteros muy grandes.

El algoritmo de Karatsuba

Consideramos a 2 números enteros x e y

$$x = a * 10^{n/2} + b$$

$$y = c * 10^{n/2} + d$$

$$x * y = (a * 10^{n/2} + b) * (c * 10^{n/2} + d)$$

$$x * y = (a * 10^{\frac{n}{2}})(c * 10^{\frac{n}{2}}) + ad * 10^{\frac{n}{2}} + bc * 10^{\frac{n}{2}} + bd$$

$$x * y = ac * 10^{2(\frac{n}{2})} + (ad + bc) * 10^{\frac{n}{2}} + bd$$

Notamos :

$$(a + b)(c + d) - ac - bd$$

$$(ac + ad + bc + bd) - ac - bd = ad + bc$$

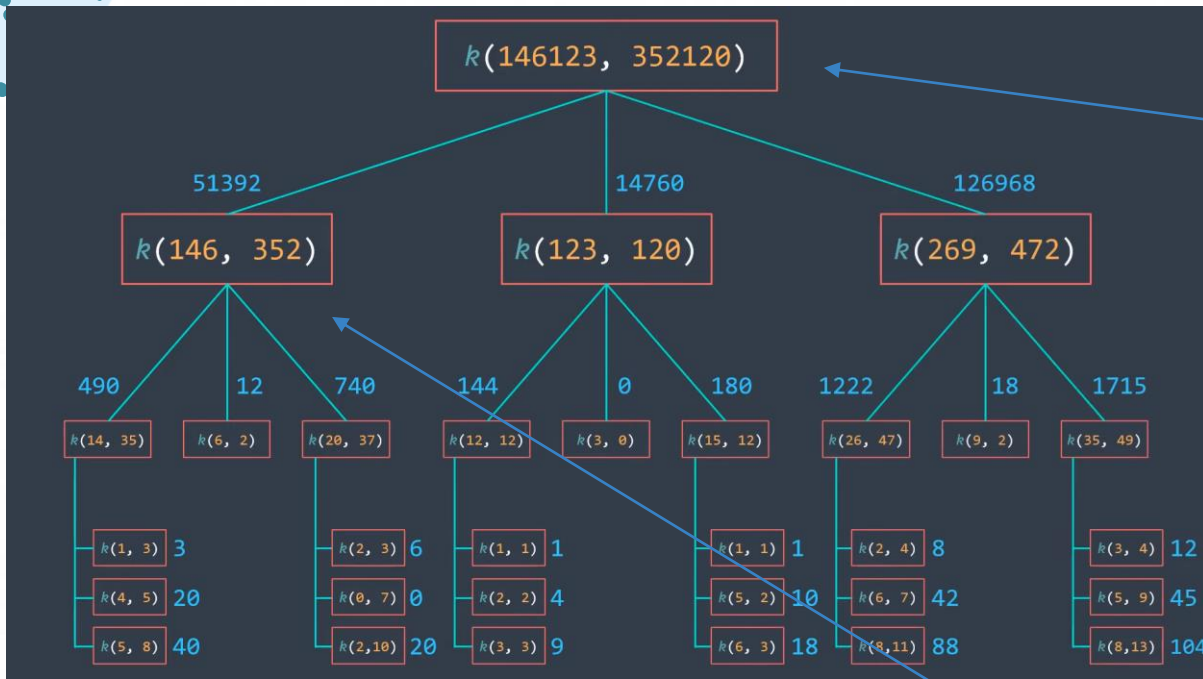
$$x * y = ac * 10^{2(\frac{n}{2})} + (ad + bc) * 10^{\frac{n}{2}} + bd$$

```
11  ac <- multiplicacion(a,c)
12  bd <- multiplicacion(b,d)
13  ad_sum_bc <- multiplicacion(a+b, c+d)-ac-bd
14  mult <- ac * (10 ** (2 * medio)) + (ad_sum_bc * (10 ** medio)) + bd
```

Ejemplo: Consideramos 2 números de 6 dígitos , donde x= 146 123 e y = 352 120

x: $\begin{matrix} a & b \\ 146 & | & 123 \end{matrix}$

y: $\begin{matrix} 352 & | & 120 \\ c & & d \end{matrix}$



$$n = 6$$

$$\text{mitad} = \frac{n}{2} = 3$$

$$ac = 51392$$

$$bd = 14760$$

$$(ad + bc) = (a + c)(b + d) - ac - bd = 60816$$

$$ac * 10^{2 * \text{mitad}} + (ad + bc) * 10^{\text{mitad}} + bd$$

$$= 51392 * 10^6 + 60816 * 10^3 + 14760$$

$$= 51\,452\,830\,760$$

$$n = 2$$

$$\text{mitad} = \frac{n}{2} = 1$$

$$ac = 3$$

$$bd = 20$$

$$(ad + bc) = (a + c)(b + d) - ac - bd = 40 - 20 - 3 = 17$$

$$ac * 10^{2 * \text{mitad}} + (ad + bc) * 10^{\text{mitad}} + bd$$

$$= 3 * 10^2 + 17 * 10^1 + 20 = 490$$

$$n = 3$$

$$\text{mitad} = \frac{n}{2} = 1$$

$$ac = 490$$

$$bd = 12$$

$$(ad + bc) = (a + c)(b + d) - ac - bd = 740 - 490 - 12 = 238$$

$$ac * 10^{2 * \text{mitad}} + (ad + bc) * 10^{\text{mitad}} + bd$$

$$= 490 * 10^2 + 238 * 10^1 + 12 = 51392$$

Pseudocódigo

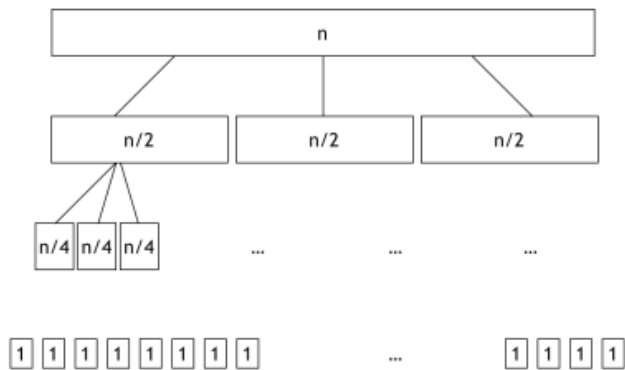
Caso base

```
1  Funcion mult <- multiplicacion( num1 , num2)
2  Si num1 < 10 O num2 < 10 Entonces
3      mult <- num1*num2
4  SiNo
5      n <- maximum (length(string(num1),string(num2)))
6      mitad <- ndiv2
7      a <- num1 div (10^mitad)
8      b <- num1 mod(10^mitad)
9      c <- num2 div (10^mitad)
10     d <- num2 mod(10^mitad)
11     ac <- multiplicacion(a,c)
12     bd <- multiplicacion(b,d)
13     ad_sum_bc <- multiplicacion(a+b, c+d)-ac-bd
14     mult <- ac * (10 ** (2 * mitad)) + (ad_sum_bc * (10 ** mitad)) + bd
15  FinSi
16
17  FinFuncion
```

03

Complejidad

Costo temporal de la función multiplicar:



$$T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{n}{2}\right) + T\left(\frac{n}{2}\right) + c * n$$

$$\rightarrow T(n) = 3 * T\left(\frac{n}{2}\right) + c * n$$

$$T\left(\frac{n}{2}\right) = 3 * T\left(\frac{n}{4}\right) + c * \frac{n}{2}$$

$$T\left(\frac{n}{4}\right) = 3 * T\left(\frac{n}{8}\right) + c * \frac{n}{4}$$

...

Tendremos la siguiente expresión:

$$T(n) = 3^i T\left(\frac{n}{2^i}\right) + \left(\frac{3}{2}\right)^{i-1} cn + \left(\frac{3}{2}\right)^{i-2} cn + \dots + cn$$

Hacemos :

$$\frac{n}{2^i} = 1 \rightarrow n = 2^i$$

$$i = \log_2 n$$

$$T(n) = 3^{\log_2 n} + cn \left[\frac{\left(\frac{3}{2}\right)^i - 1}{\frac{3}{2} - 1} \right]$$

$$T(n) = 3^{\log_2 n} + 2nc \left[\left(\frac{3}{2}\right)^{\log_2 n} - 1 \right]$$

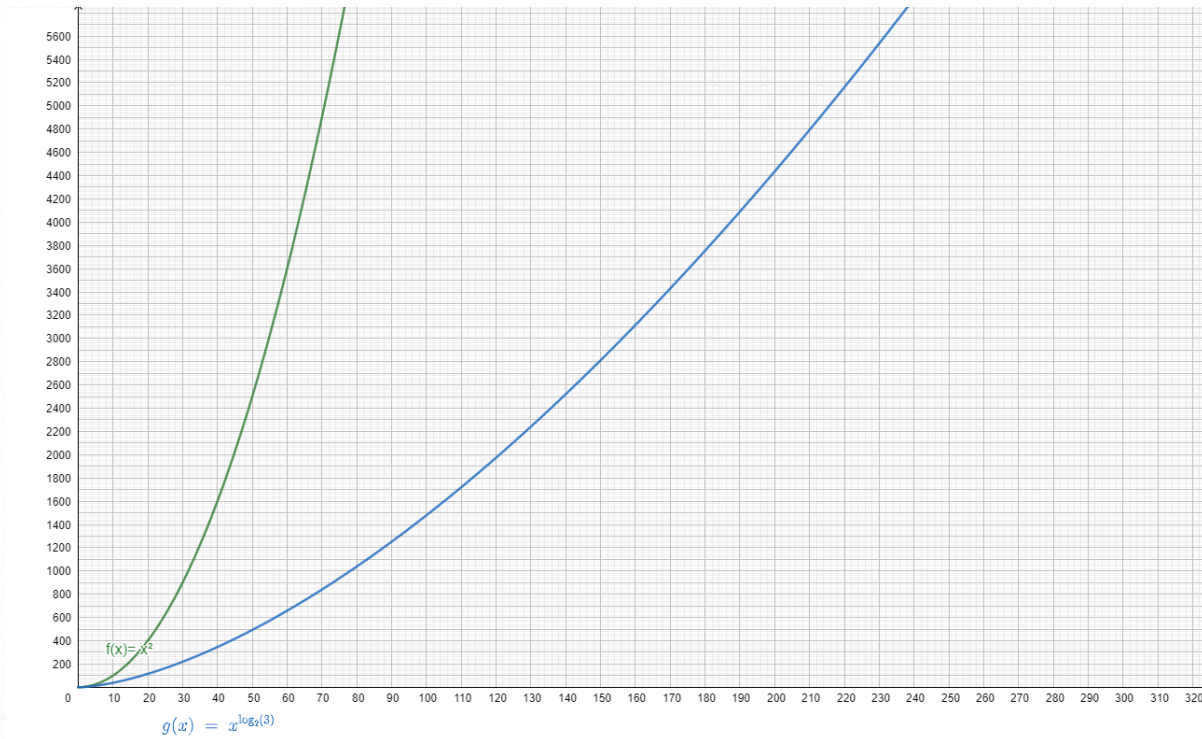
$$T(n) = 3^{\log_2 n} + 2cn \left(\frac{3}{2}\right)^{\log_2 n} - 2nc = n^{\log_2 3} + 2c * n * n^{\log_2 3 - 1} - 2nc$$

$$T(n) = 2cn^{\log_2 3} + n^{\log_2 3} - 2nc$$

$$T(n) \in O(n^{\log_2 3})$$

$$\log_2 3 \approx 1,585$$

Gráficas



04 Código

```
1 def multiplicacion(x, y):
2     if x < 10 or y < 10:
3         return x * y
4     else:
5         n = max(len(str(x)), len(str(y)))
6         mitad = n // 2
7
8         a = x // (10 ** (mitad))
9         b = x % (10 ** (mitad))
10
11        c = y // (10 ** (mitad))
12        d = y % (10 ** (mitad))
13
14        ac = multiplicacion(a, c)
15        bd = multiplicacion(b, d)
16        ad_sum_bc = multiplicacion(a+b, c+d)-ac-bd
17
18        return ac * (10 ** (2 * mitad)) + (ad_sum_bc * (10 ** mitad)) + bd
19
20 print("Resultado: ")
21 print(multiplicacion(146123,352120))
```

Resultado:
51452830760

Avances en los métodos de multiplicación.

Año	$M(n)$	autor
-2000?	$O(n^2)$	desconocido
1960	$O(n^{\log_2 3})$	Karatsuba
1966	$O(n \log n e^{\sqrt{2 \log_2 n}})$	Toom-Cook
1971	$O(n \log n \log \log n)$	Schönhage-Strassen
2007	$O(n \log n K^{\log^* n})$	Fürer
2018	$O(n \log n 4^{\log^* n})$	Harvey-van der Hoeven



Gracias