

Uma estratégia de conquista esperta

Infelizmente, a estratégia de decomposição que nós acabamos de ver ainda não produz um resultado interessante.

Vejamos.

Nós já sabemos que a multiplicação de X e Y dígito a dígito leva tempo n^2 .

De maneira análoga, as multiplicações envolvendo X_i, Y_j (que possuem $n/2$ dígitos cada um) levam tempo $\frac{n}{2} \cdot \frac{n}{2} = \frac{n^2}{4}$.

Mas, como são realizadas 4 multiplicações desse tipo, o tempo total é de

$$4 \cdot \frac{n^2}{4} = n^2$$

Ou seja, nós não ganhamos nada até aqui.

A observação chave, a seguir, é que a multiplicação de X e Y também pode ser escrita como

$$X \cdot Y = X_2Y_2 \cdot 10^n + (X_2Y_1 + X_1Y_2) \cdot 10^{n/2} + X_1Y_1$$

e aqui nós vemos que nós não precisamos dos resultados intermediários X_2Y_1 e X_1Y_2 separadamente, mas apenas da sua soma.

E agora, tudo o que é preciso é uma pequena esperteza ...

Veja o que acontece quando nós deixamos as potências de 10 de lado e multiplicamos

$$(X_2 + X_1) \cdot (Y_2 + Y_1) = X_2Y_2 + (X_2Y_1 + X_1Y_2) + X_1Y_1$$

Ou seja, o termo que nos interessa apareceu de novo.

E a observação importante aqui é que, se as somas do lado esquerdo são realizadas primeiro, então a coisa toda leva tempo $\frac{n}{2} + \frac{n}{2} + \frac{n^2}{4}$.

O único problema é que essa multiplicação nos dá o termo que nós desejamos misturado com X_2Y_2 e X_1Y_1 .

Mas, isso é um problema fácil de resolver

- basta calcular os termos X_2Y_2 e X_1Y_1 em separado
- e depois fazer a subtração

$$(X_2Y_1 + X_1Y_2) = X_2Y_2 + (X_2 + X_1) \cdot (Y_2 + Y_1) + X_1Y_1$$

Ora, mas os termos X_2Y_2 e X_1Y_1 já eram termos que eram calculados na solução original.

Portanto, não há perda de tempo (realmente) aqui.

Em resumo, a esperteza consiste em calcular apenas 3 substituições:

- X_2Y_2
- X_1Y_1
- $X_2Y_2 + (X_2 + X_1) \cdot (Y_2 + Y_1) + X_1Y_1$

(O procedimento também envolve algumas somas e subtrações, mas a sua contribuição para o tempo total não é relevante.)

Finalmente, uma vez que esses 3 termos foram calculados, a solução é obtida através de

$$X \cdot Y = X_2Y_2 \cdot 10^n + (X_2Y_1 + X_1Y_2) \cdot 10^{n/2} + X_1Y_1$$

Pseudo-código e análise de complexidade

Uma vez que as estratégias de divisão e conquista foram definidas, é imediato escrever o algoritmo que implementa essa solução:

```
Procedimento Mult-intDC ( X,Y : n dígitos )
{
    Se ( X e Y são pequenos ) Retorna ( X * Y )

    (X2,X1) <-- Quebra (X)
    (Y2,Y1) <-- Quebra (Y)

    C1 <-- Mult-intDC (X2,Y2)
    C2 <-- Mult-intDC (X1,Y1)
    C3 <-- Mult-intDC (X2+X1,Y2+Y1) - C1 - C2

    R <-- C1 * 10^n + C3 * 10^n/2 + C2
}
```

Examinando o pseudo-código, nós obtemos a seguinte equação de recorrência para o tempo de execução do algoritmo

$$T(n) = 3 \cdot T(n/2) + O(n)$$

que possui solução

$$T(n) = \Theta(n^{\log_2 3}) = \Theta(n^{1,585})$$