Listex 8

João Pedro Machado Silva, BV3032477

Exercício 3:

a. A multiplicação de números inteiros grandes é uma operação fundamental na computação e em várias aplicações práticas, como criptografia e processamento de sinais. O método ingênuo de multiplicação, conhecido como multiplicação de papel e lápis, possui complexidade de tempo $O(n^2)$, onde n é o número de dígitos dos números a serem multiplicados. Esse método se torna ineficiente para números com muitos dígitos.

O Algoritmo de Karatsuba, proposto por Anatolii Alexeevich Karatsuba em 1960, melhora a eficiência da multiplicação de inteiros grandes. Este algoritmo divide os números em partes menores e utiliza a técnica de divisão e conquista para reduzir o número de multiplicações necessárias. A complexidade de tempo do Algoritmo de Karatsuba é O(n^log2 3)≈O(n^1.585), tornando-o mais eficiente do que o método tradicional.

A lógica do Algoritmo de Karatsuba envolve uma abordagem inovadora para multiplicação. O algoritmo divide os números a serem multiplicados em duas partes: uma parte mais significativa e outra menos significativa. Em seguida, ele calcula três produtos intermediários em vez dos quatro produtos necessários na multiplicação convencional. Esses produtos intermediários são combinados de maneira inteligente para formar o resultado final. A fórmula usada pelo algoritmo é baseada na observação de que os produtos cruzados podem ser obtidos de forma mais eficiente através de uma combinação de adições e subtrações dos produtos intermediários. Esta técnica de redução do número de multiplicações de quatro para três resulta em uma melhora significativa na eficiência computacional, especialmente para números com muitos dígitos, fazendo do Algoritmo de Karatsuba uma solução elegante e poderosa para a multiplicação de grandes inteiros.

b.

Karatsuba (u, v, n)

1 se n ≤ 3

```
devolva u×v e pare
m := ⌈n/2⌉
p := ⌊u/10^m⌋
q := u mod 10^m
r := ⌊v/10^m⌋
s := v mod 10^m
pr := Karatsuba (p, r, m)
qs := Karatsuba (q, s, m)
y := Karatsuba (p + q, r + s, m+1)
uv := pr×102m + (y − pr − qs)×10m + qs
devolva uv
```

c. T(n) = 1, para n=1
T(n) = 3T(n/2) + n, para n>1

a= 3, b = 2 e f(n) = n (obedece às restrições.)
n^(logb a)= n^(log2 3) = n^(1,5…) > n -> Caso 1 -> **T(n) ∈ O(n^(1,585)).**

d. Implementado.

e. Sim, Além do algoritmo de Karatsuba, que é eficiente para a multiplicação de números inteiros grandes com complexidade O(n^1.585), existem outros algoritmos notáveis com diferentes complexidades. O algoritmo de Strassen, que é usado para multiplicação de matrizes, tem uma complexidade de O(n^2.81). O algoritmo de Toom-Cook, com suas variantes como Toom-3 e Toom-4, pode alcançar complexidades em torno de O(n^1.465) dependendo da variante utilizada. Para números ainda maiores, o algoritmo de Schönhage-Strassen é muito eficiente, com uma complexidade de O(nlogn nlogn logn)=. Outro método avançado é o baseado em Transformada Rápida de Fourier (FFT), que também tem complexidade O(n^logn). Entre esses, o algoritmo de Schönhage-Strassen e o baseado em FFT são geralmente os mais eficientes para números muito grandes.