Multiplicação eficiente de inteiros e de matrizes

Nota. Na resolução dos exercícios desta folha pode utilizar o Teorema que está no verso.

1. **Multiplicar inteiros**

   Considere-se o problema de multiplicar 2 inteiros x e y com comprimento (nº de bits) que não excede n (comprimento $2n$ na alínea (c)).

   (a) Qual a ordem de grandeza do algoritmo seguinte? Suponha que $|x| = |y| = n$. Exprima a sua resposta em termos de n.

```
prod1(x,y):
  s=0
  for i=1 to y:
    s=s+x
  return s
```

   (b) Qual a ordem de grandeza do algoritmo "escolar" da multiplicação (que é essencialmente o algoritmo normalmente usado nos CPUs)?

   (c) Sejam x e y com $2n$ bits divididos em 2 partes, x=$a \times 2^n + b$, $y = c \times 2^n + d$ com $|a| = |b| = |c| = |d| = n$. No algoritmo de Karatsuba a expressão fundamental que permite permite uma maior eficiência da multiplicação é

$$xy = ac(2^{2n} - 2^n) + (a + b)(c + d)2^n + bd(1 - 2^n)$$

   No cálculo de xy há 3 chamadas recursivas ao algoritmo. Admita que as computações adicionais necessárias ("carries" entre partes dos inteiros...) podem ser efectuadas em tempo $O(n)$.

   i. Descreva esquematicamente o algoritmo de Karatsuba, indicando as chamadas recursivas e as operações adicionais.

   ii. Deduza uma recorrência para a majoração do tempo de execução do algoritmo de Karatsuba e resolva-a.

   iii. Suponha que se usava a expressão $xy = ac \times 2^{2n} + (ad + bc)2^n + bd \times 2^n$ para a definição de um algoritmo semelhante ao de Karatsuba (4 chamadas recursivas). Mostre que, em termos de ordem de grandeza, não há melhoria relativamente ao algoritmo "escolar"; é também $O(n^2)$ (na realidade o algoritmo vai ser bastante mais lento).

2. **Multiplicar matrizes**

   Considere o método de Strassen para a multiplicação de matrizes. Os cálculos que permitem a redução de 8 para 7 do número de chamadas recursivas são os seguintes

$$\left\{\begin{array}{lllllll} M_1 & = & (A+D)(E+H) & \quad & M_2 & = & D(G-E) \\ M_3 & = & (B-D)(G+H) & & M_4 & = & (A+B)H \\ M_5 & = & (C+D)E & & M_6 & = & A(F-H) \\ M_7 & = & (C-A)(E+F) & & & & \\ R_{11} & = & M_1+M_2+M_3-M_4 & & R_{12} & = & M_4+M_6 \\ R_{21} & = & M_2+M_5 & & R_{22} & = & M_1-M_5+M_6+M_7 \end{array}\right.$$

   onde as matrizes a multiplicar são $\begin{bmatrix} A & B \\ C & D \end{bmatrix}$ e são $\begin{bmatrix} E & F \\ G & H \end{bmatrix}$ e o resultado é $\begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}$.

(a) Mostre que, no modelo uniforme, o algoritmo usual de multiplicação de 2 matrizes de dimensões $n \times n$ é de ordem $O(n^3)$. Quantas multiplicações elementares são efectuadas?

(b) Deduza uma recorrência para a majoração do tempo de execução do algoritmo de Strassen.

(c) Resolva a recorrência.

(d) Suponha que uma implementação do algoritmo clássico demora um tempo n^3 (em microsegundos) enquanto uma implementação do algoritmo de Strassen demora um tempo de $100n^{2.8}$. Determine o valor de n a partir do qual o método de Strassen se torna assintóticamente vantajoso.

(e) Num método recursivo mais directo fazem-se 8 multiplicações

$$R_{11} = AE + BG,\ R_{12} = AF + BH,\ R_{21} = CE + DG,\ R_{22} = CF + DH$$

Escreva a recorrência associada à eficiência deste método e resolva-a, mostrando que, em termos de ordem de grandeza, não há vantagem relativamente ao algoritmo usual.

Teorema. A solução de uma recorrência com a equação geral da forma $t(1) = c$, $t(n) = at(n/b) + cn^k$ onde c e k são inteiros positivos, a e b são inteiros com $a \geq 1$, $b \geq 2$ tem a seguinte ordem de grandeza

$$\begin{cases} t(n) \in O(n^{\log_b a}) & \text{se } a > b^k \\ t(n) \in O(n^k \log n) & \text{se } a = b^k \\ t(n) \in O(n^k) & \text{se } a < b^k \end{cases}$$