SCC 201/501 — Introdução à Ciência de Computação II (ICMC/USP)

Lista de Exercícios 1: Análise de algoritmos

Professor: Moacir Pereira Ponti Jr. *PAE(s): Pâmela/Paulo Henrique*

1. Seja a seguinte definição: "Dadas duas funções, $f(n)$ e $g(n)$, diz-se que $f(n)$ é da ordem de $g(n)$ ou que $f(n)$ é $\mathcal{O}(g(n))$, se existirem inteiros positivos a e b tais que $f(n) \leq a * g(n)$ para todo $n \geq b$." Verifique se as seguintes proposições estão corretas:

   (i) $7 \in \mathcal{O}(n)$

   (ii) $n \in \mathcal{O}(1)$

   (iii) $n + 7 \in \mathcal{O}(n)$

   (iv) $n + 7 \in \mathcal{O}(1)$

   (v) $n^2 + 2 \in \mathcal{O}(n)$

   (vi) $n + 2 \in \mathcal{O}(n^2)$

   (vii) $3n^3 + n \in \mathcal{O}(n^3)$

   (viii) $2n^4 \in \mathcal{O}(n^4)$

   (ix) $n^4 \in \mathcal{O}(2n^4)$

   (x) $3n^4 + 2n^3 \in \mathcal{O}(2n^4)$

   (xi) $2n^4 \in \mathcal{O}(3n^4 + 2n^3)$

   (xii) $log\ n \in \mathcal{O}(1)$

   (xiii) $log\ n + 1 \in \mathcal{O}(log\ n)$

   (xiv) $log\ n + 1 \in \mathcal{O}(n)$

   (xv) $log\ n + 1 \in \mathcal{O}(n^2)$

   (xvi) $log\ n + 1 \in \mathcal{O}(n^3)$

   (xvii) $n \cdot log\ n \in \mathcal{O}(1)$

   (xviii) $n \cdot log\ n + 1 \in \mathcal{O}(log\ n)$

   (xix) $n \cdot log\ n + 1 \in \mathcal{O}(n)$

   (xx) $n \cdot log\ n + 1 \in \mathcal{O}(n^2)$

   (xxi) $n \cdot log\ n + 1 \in \mathcal{O}(n^3)$

   (xxii) $2log\ n \in \mathcal{O}(n \cdot log\ n)$

   (xxiii) $3n \cdot log\ n \in \mathcal{O}(log\ n)$

   (xxiv) $2n + n \in \mathcal{O}(2^3)$

   (xxv) $n^2 \in \mathcal{O}(2^n)$

   (xxvi) $100n^4 \in \mathcal{O}(2^n)$

   (xxvii) $100n^4 \in \mathcal{O}(n^n)$

(xxviii) $2^n \in \mathcal{O}(100n^4)$

(xxix) $2^n \in \mathcal{O}(n^n)$

(xxx) $n^n \in \mathcal{O}(2^n)$

(xxxi) $n^{100} \in \mathcal{O}(n^n)$

(xxxii) $n(n+1)/2 \in \mathcal{O}(n^3)$

(xxxiii) $n(n+1)/2 \in \mathcal{O}(n^2)$

(xxxiv) $n(n+1)/2 \in \Theta(n^3)$

(xxxv) $n(n+1)/2 \in \Omega(n)$

2. Ordene as seguintes funções por suas taxas de crescimento: n, $\sqrt[2]{n}$, $n^1, 5$, n^2, $nlog\ n$, $n{\cdot}log\ log\ n$, $n(log\ n)^2$, $nlog\ n^2$, $2/n$, 2^n, $2^{n/2}$, 37, $n^2log\ n$, n^3, $(n-2)!$, $5log\ (n+100)^10$, 2_{2n}, $0,001n^4+3n^3+1$, $log^2\ n$, $\sqrt[3]{n}$, 3^n.

3. Para cada uma das seguintes funções, determine a classe $\Theta(g(n))$ a qual a função pertence. (Use $g(n)$ mais simples possível)

   (i) $(n^2+1)^10$
   (ii) $\sqrt{10n^2+7n+3}$
   (iii) $2nlog\ (n+2)^2+(n+2)^2log\ n/2$
   (iv) $2^{n+1}+3^{n-1}$
   (v) $\lfloor log\ n \rfloor$

4. Compare as duas funções n^2 e $\frac{2^n}{4}$ para vários valores de n. Determine quando a segunda se torna maior que a primeira.

5. Prove que todo polinômio de grau k, $p(n)=a_kn^k+a_{k-1}n^{k-1}+\cdots+a_0$, com $a_k>0\forall k$ pertence a $\Theta(n^k)$.

6. Prove que as funções exponenciais a^n têm diferentes ordens de crescimento para diferentes valores de a.

7. Prove as seguintes proposições ou dê um contra-exemplo:

   (i) $t(n)\in\mathcal{O}(g(n))\Rightarrow g(n)\in\Omega(t(n))$
   (ii) $\Theta(\alpha g(n))=\Theta(g(n))\forall\alpha>0$
   (iii) $\Theta(g(n))=\mathcal{O}(g(n))\bigcap\Omega(g(n))$

8. Encontre a ordem de crescimento dos seguintes somatórios:

   (i) $\sum_{i=0}^{n-1}(i^2+1)^2$
   (ii) $\sum_{i=2}^{n-1}(log\ i^2)$
   (iii) $\sum_{i=1}^{n}(i+1)\cdot 2^{i-1}$
   (iv) $\sum_{i=0}^{n-1}\sum_{j=0}^{i-1}(i+j)$

9. Considere o seguinte algoritmo, sendo $n \geq 0$ e inteiro:

```
  (i) s ← 0
 (ii) para i ← 1 até n faça
(iii)    s ← s + i * i
 (iv) devolve s;
```

   (i) Qual a resposta dada ao executar este algoritmo?
   (ii) Qual é a operação básica?
   (iii) Quantas vezes essa operação é executada?
   (iv) Qual é a classe de eficiência desse algoritmo?
   (v) Existe um algoritmo melhor que responda ao mesmo problema? Descreva-o ou mostre que tal algoritmo não existe.

10. Considere um computador com *clock* de 2GHz, que realiza cada operação relevante em 1 ciclo. Estime, apenas com esses dados, o tempo necessário para que ele execute um algoritmo que realiza $(n^2 - n)/2$ operações relevantes, considerando que há 4M dados de entrada.

11. Idem, usando um algoritmo que realiza n^3 operações relevantes.

12. Idem, usando um algoritmo que realiza 2^n operações relevantes.

13. Idem, usando um algoritmo que realiza n^n operações relevantes.

14. Idem, para um computador com *clock* de 100MHz, ordenando a mesma seqüência, usando um algoritmo que realiza $4/3 * n \, log \, n$ operações relevantes. Analise os resultados.

Referências

[1] Horowitz, E., Sahni, S. Rajasekaran, S. *Computer Algorithms*, Computer Science Press, 1998.

[2] Nakamiti, G., *Listas de Exercícios de Estruturas de Dados II*, Engenharia de Computação. PUC-Campinas, 2007.

[3] Tenenbaum, A. M., Langsam, Y., Augestein, M. J., *Estruturas de Dados Usando C*. Makron Books, 1995.

[4] Levitin, A. V., *Introduction to the Design and Analysis of Algorithms*. Pearson Addison-Wesley, 2007.

[5] Parte deste material foi adaptado das listas de exercício do Prof. João Luís Garcia Rosa, ICMC/USP.