PUC-Rio — Departamento de Informática
Data: 19 de outubro de 1996 — Período: 1996.2

ESTRUTURAS DISCRETAS

Primeira Prova

---

1. (1 ponto) Defina o complemento de uma relação α, representado por $\overline{\alpha}$, como a relação tal que $a\,\overline{\alpha}\,b$ se e somente se $(a,b) \notin R_\alpha$. Demonstre que $(\overline{\alpha})^{-1} = \overline{(\alpha)^{-1}}$.

2. (1 ponto) Determine se as relações abaixo, definidas pelos pares ordenados (m,n) de inteiros positivos, possuem cada uma das seguintes propriedades: reflexividade, simetria, antissimetria e transitividade (cada 3 erros eliminam um acerto).

   (a) m é divisível por n
   (b) $m+n \geq 50$
   (c) $m+n$ é par
   (d) $m+n$ é ímpar
   (e) mn é par
   (f) m é uma potência de n
   (g) $m+n$ é um múltiplo de 3
   (h) m é maior do que n

3. (1 ponto) Determine quais dos seguintes conjuntos definem uma relação de equivalência e quais definem uma relação de compatibilidade:

   (a) $A = \{(x,y) \mid x \text{ e } y \text{ são pessoas da mesma idade}\}$
   (b) $B = \{(x,y) \mid x \text{ e } y \text{ são primos}\}$

4. (1 ponto) Considere três algoritmos cujas complexidades de pior caso sejam dadas por $T_1(n) = n^3$, $T_2(n) = 3^n$ e $T_3(n) = 3 \times n!$. Suponha que $n = 10$ é o tamanho do maior problema que pode ser resolvido por cada um destes algoritmos com o computador mais rápido atualmente disponível, em determinado tempo de processamento. Qual seria o tamanho do maior problema que poderia ser resolvido na prática por estes mesmos algoritmos, utilizando-se de um computador 10 vezes mais rápido e mantendo-se constante o tempo de processamento disponível e utilizado? Compare estes algoritmos em termos dos benefícios que tiram de avanços tecnológicos.

5. (3 pontos) Seja $a = (a_1, a_2, \cdots, a_n)$ um vetor ordenado em ordem crescente, que armazena n números inteiros. Considere o algoritmo de busca binária que, dado um valor inteiro x, determina se existe $i \in \{1, 2, \cdots, n\}$ tal que $a_i = x$. (Aliás, qual é mesmo sua complexidade?) Especifique um algoritmo de "busca ternária", baseando-se na partição do intervalo de busca em três partes iguais. Justifique o princípio de funcionamento do algoritmo. Analise e avalie sua complexidade, justificando. Compare os algoritmos de busca binária e de busca ternária em termos de suas complexidades. Você considera que seria possível propor alguma generalização desta idéia? Discuta suas implicações.

6. (3 pontos) Considere os três algoritmos `Alg-1`, `Alg-2` e `Alg-3` abaixo apresentados. Sejam $T_1(n)$, $T_2(n)$ e $T_3(n)$, respectivamente, as funções que descrevem suas complexidades.

```
Algoritmo Alg-1(n)
     i := 1; j := 0
     for k := 1 to n do
          j := i + j
          i := j - i
     end-for
     return{j}
end-Algoritmo

Algoritmo Alg-2(n)
     if n <= 1
     then return{n}
     else return{Alg-1(n-1)+Alg-1(n-2)}
end-Algoritmo

Algoritmo Alg-3(n)
     i := 1; j := 0; k := 0; h := 1
     while (n > 0) do
          if resto(n/2) = 0
          then t := j*h
               j := i*h + j*k + t
               i := i*k + t
          end-if
          t := h*h
          h := 2*k*h + t
          k := k*k + t
          n := inteiro(n/2)
     end-while
     return{j}
end-Algoritmo
```

   (a) Utilizando a especificação do algoritmo `Alg-1`, analise (justificando) as funções que ele executa (em termos da interpretação dos valores retornados para cada valor de n).

   (b) Na verdade, estes três algoritmos executam exatamente a mesma função. Analise e avalie cuidadosamente a complexidade de cada um deles, comparando-os em termos de sua eficiência. Atenção: **não** é necessário calcular a complexidade do algoritmo `Alg-2`, que é dada por $T_2(n) = O(((1+\sqrt{5})/2)^n)$, bastando deixar indicada a equação de recorrência que deveria ser resolvida para a obtenção de $T_2(n)$.
   Bônus: 1 (um) ponto extra pela dedução formal de $T_2(n)$ a partir da equação acima mencionada.

Observações:

- Prova sem consulta de qualquer espécie.
- Os algoritmos devem ser justificados e especificados em pseudo-linguagem, não podendo ser utilizadas linguagens de programação específicas.