Exercícios - Corretude de Algoritmos Recursivos

Análise de Algoritmos - DINF - UFPR

Exercício 1. O algoritmo abaixo calcula $3^n - 2^n, \forall n \geq 0$. Prove que o algoritmo está correto.

Algoritmo $g(n)$
 se $n \leq 1$ **então**
  retorna n
 senão
  retorna $5g(n-1) - 6g(n-2)$

Exercício 2. O algoritmo abaixo calcula a multiplicação de números naturais. Prove que o algoritmo está correto.

Algoritmo $\text{mult}(y, z)$
 se $z = 0$ **então**
  retorna 0
 senão
  retorna $\text{mult}(2y, \lfloor z/2 \rfloor) + y(z \bmod 2)$

Exercício 3. O algoritmo abaixo calcula a exponenciação de números naturais. Prove que o algoritmo está correto.

Algoritmo $\text{power}(y, z)$
 se $z = 0$ **então**
  retorna 1
 senão se *z é ímpar* **então**
  retorna $\text{power}(y^2, \lfloor z/2 \rfloor) \cdot y$
 senão
  retorna $\text{power}(y^2, \lfloor z/2 \rfloor)$

Exercício 4. O algoritmo abaixo calcula a soma dos elementos do vetor $A[1..n]$. Prove que o algoritmo está correto.

Algoritmo $\text{sum}(A, n)$
 se $n \leq 1$ **então**
  retorna $A[1]$
 senão
  retorna $\text{sum}(A, n-1) + A[n]$

Exercício 5. O algoritmo abaixo devolve a posição m onde está o menor elemento de um vetor $v[a..b]$. Prove que o algoritmo está correto.

Algoritmo Minimo(v, a, b)
  se $a = b$ **então**
    retorna a
  $m \leftarrow$ Minimo$(v, a, b-1)$
  se $v[b] < v[m]$ **então**
    $m \leftarrow b$
  retorna m

Exercício 6. Considere o seguinte problema computacional:

Partição de Vetor

Entrada: (v, a, b) onde v é um vetor indexado por $[a..b]$.
Saída **:** escolhe um elemento qualquer x de v, e modifica v de forma a obter um índice $m \in [a-1..b]$ tal que

- $v[m] = x$
- $v[a] \leq x$, $v[a+1] \leq x$, ..., $v[m-1] \leq x$
- $v[m+1] > x$, $v[m+2] > x$, ..., $v[b] > x$

No final, devolve este índice m.

Visualmente, a saída da Partição de Vetor devolve m e altera o vetor assim:

i	$a..m-1$	m	$b..m-1$
$v[i]$	$\leq x$	x	$> x$

Considere que exista um algoritmo chamado **Particiona**(v, a, b) que resolve corretamente o problema da Partição de Vetor.

DICA: perceba que após a execução do **Particiona**(v, a, b), x fica na posição que deveria estar se o vetor estivesse ordenado.

O Quicksort é um algoritmo para ordenar um vetor $v[a..b]$, descrito abaixo:

Quicksort(v, a, b)

se $a < b$ **então**
  $m \leftarrow$ Particiona(v, a, b)
  Quicksort$(v, a, m-1)$
  Quicksort$(v, m+1, b)$

Prove que o Quicksort está correto.