Questão 8 ~~Pág. 89~~ Pag 108 / /

insere - $O(1)$

remove - $O(K)$ sendo K o nº de elementos em D.

O máximo de inserções será de n elementos, então a operação remove só poderá mover n elementos da pilha D para a pilha E, portanto no máximo será um consumo de $O(n)$

$O(n) + O(n) = 2\,O(n) = O(n)$

Seja $T(n)$ o custo amortizado de todas as operações para n elementos, $T(n) = O(n)$ então o custo amortizado de cada operação será

$\frac{T(n)}{2} = O(1)$

Questão 6 Não sei Fazer

Árvore B

LISTA 2 - ALAIR

Questão 16 - Regra de Horner

$P_n(x) = a_n x^n + \dots + a_1 x^1 + a_0 x^0$

$q_{n-1}(x) = a_n x^{n-1} + \dots + a_2 x + a_1$ e $p_n(x) = x\,q(x) + a_0$

 / /

Algo (A, p, r, x)
  Se p = r
    então Se r = 0
      devolva A[r]
    se r = 1
      devolva A[r].x
    i ← 2 v ← x
    enquanto i < r
      v ← v.x
      i ← i + 1
    devolva A[r].v
  senão q ← ⌊(r+p)/2⌋
    devolva Algo(A, p, q, x) + Algo(A, q+1, r, x)

Quantas adições faz?

Sendo $n = r - p + 1$ adições $n-1$

Quantas multiplicações faz?

$\sum_{k=1}^{n} k - 1 = \frac{n.(n-1)}{2} = \frac{n^2-n}{2}$

Regra Horner

$q_{n-1}(x) = a_5 x^4 + a_4 x^3 + a_3 x^2 + a_2 x + a_1$

$= \sum_{k=1}^{n-2} k - 1 = \frac{1}{2} \cdot (n-2)\cdot(n-3)$

$= \frac{1}{2}\, n^2 - 3n - 2n + 6 = \frac{n^2 - 5n + 6}{2}$

$P_n(x) = x \cdot q(x) + a_0$

$= 1 + \frac{1}{2} n^2 - 5n + 6$ multiplicações

adições

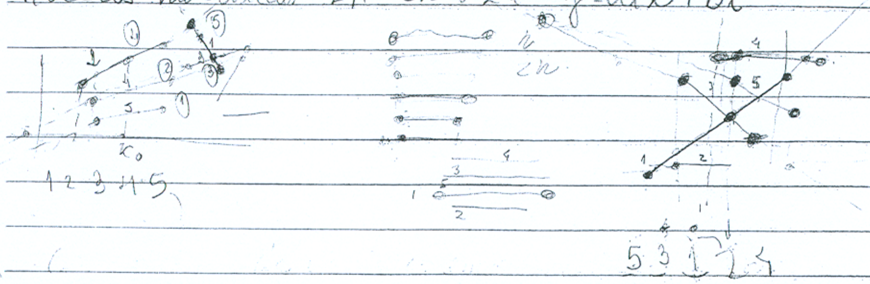
$n-1$ adições

O algoritmo faz $\frac{n^2-n}{2}$ enquanto que o método faz $1 + \frac{1}{2} n^2 - 5n + 6 < \frac{n^2-n}{2}$

 2

Questão 19 - Retas

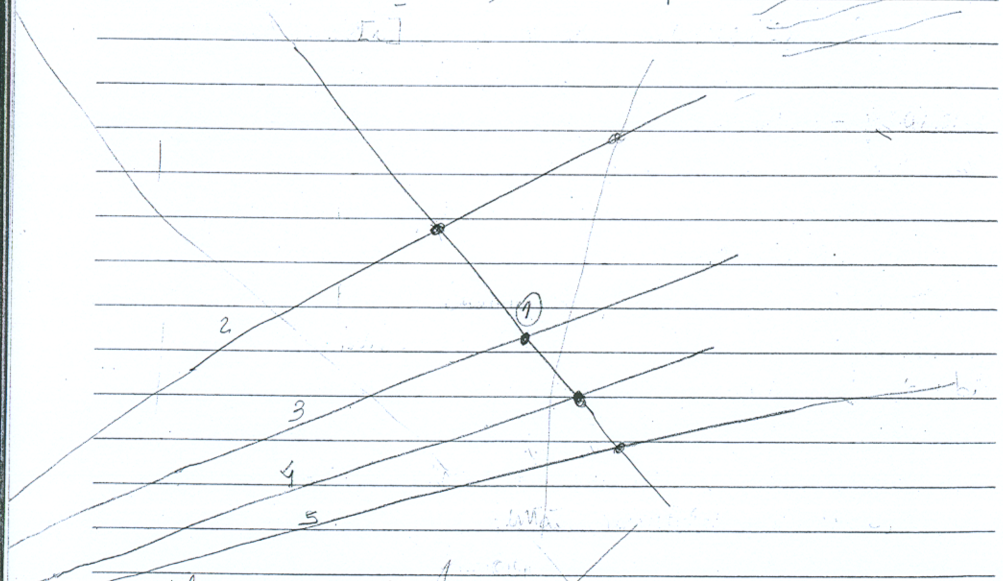
n retas não verticais $L_1 \ldots L_n$. L_i: $y = a_i x + b_i$



$t = 2 \quad 3 \quad 2 \quad 3 \quad 4$

$K = 3 \quad 4 \quad 5$

Inv 0	Inv 1	Inv 2	Inv 3	Inv 4
$-\infty$	5	3	1 2	4



Leva $O(n)$ pois o segundo loop (linha 7) no máximo executa 1 vez devido a ordenação preliminar

---

Seja n um inteiro positivo. Prove que $\left\lfloor \frac{\lfloor n/2 \rfloor}{2} \right\rfloor = \lfloor n/4 \rfloor$ e $\left\lfloor \frac{\lfloor n/a \rfloor}{b} \right\rfloor = \lfloor n/(ab) \rfloor$

Seja $K := \left\lfloor \frac{\lfloor n/2 \rfloor}{2} \right\rfloor$. Vamos provar que $K = \left\lfloor \frac{n}{4} \right\rfloor$

$K \leq \frac{\lfloor n/2 \rfloor}{2} < K+1$

$2K \leq \lfloor n/2 \rfloor < 2K+2$

$2K \leq n/2 < 2K+2$, pois $n/2 \leq \lfloor n/2 \rfloor + 1 \leq 2K+2$

$K \leq n/4 < K+1$

$\lfloor n/4 \rfloor = K$ //

Seja $K := \left\lfloor \frac{\lfloor n/a \rfloor}{b} \right\rfloor$. Vamos provar que $K = \lfloor n/ab \rfloor$

$K \leq \lfloor n/a \rfloor / b < K+1$

$bK \leq \lfloor n/a \rfloor < bK+b$

$bK \leq n/a < bK+b$, pois $n/a \leq \lfloor n/a \rfloor + 1 \leq bK+b$

$K \leq n/a.b < K+1$

$K = \left\lfloor \frac{n}{ab} \right\rfloor$ //

Questão 1 - L2 - Alair

Sejam $n \in \mathbb{Z}$ e $x \in \mathbb{R}$. Prove que

a) $\lfloor x \rfloor < n \iff x < n$; $n < \lceil x \rceil \iff n < x$

$\Rightarrow$ Seja $\lfloor x \rfloor < n$ então

Suponha $\lfloor x \rfloor = n-1$, $n-1 \leq x < n$

Portanto $x < n$ //

$\Leftarrow$ Seja $x < n$ então

$\lfloor x \rfloor \leq x$

$\lfloor x \rfloor \leq x < n$

Assim $\lfloor x \rfloor < n$ //

$\Rightarrow$ Seja $n < \lceil x \rceil$ então

Suponha $\lceil x \rceil = n+1$

$n < x \leq n+1$, Portanto $n < x$

$\Leftarrow$ Seja $n < x$

Como $x \leq \lceil x \rceil$ então

$n < x \leq \lceil x \rceil$

Assim $n < \lceil x \rceil$

/ /

b) $n \le \lfloor x \rfloor \Leftrightarrow n \le x$; $\lceil x \rceil \le n \Leftrightarrow x \le n$

$\Rightarrow$ Seja $n \le \lfloor x \rfloor$ então Suponha $n = \lfloor x \rfloor$,
$n \le x < n+1$ portanto $n \le x$

$\Leftarrow$ Seja $n \le x$, $n \le x < n+1$
Portanto $n = \lfloor x \rfloor$

$\Rightarrow$ Seja $\lceil x \rceil \le n$ então suponha $n-1 < \lceil x \rceil \le n$
$n-1 < x \le n$ pois $n-1 \le \lceil x \rceil - 1 < x$
então $x \le n$ //

$\Leftarrow$ Seja $x \le n$ então suponha $n-1 < x$
$n-1 < x \le n$
$n-1 < \lceil x \rceil \le n$, pois $x \le \lceil x \rceil \le n$
portanto $\lceil x \rceil \le n$

Questão 5 - nem $f(n) = O(g(n))$ e nem $g(n) = O(f(n))$ Perguntar
$f(n) \le c.g(n)$ $g(n) \le c.f(n)$
$f(n) > c.g(n)$ $g(n) > c.f(n)$
$f(n) > c.g(n) > c.f(n)$
$f(n) > c.g(n) > f(n)$
absurdo! pois $f(n) = f(n)$

Questão 6 - Prove que, para todas as funções $f, g : N \to N$
a) $af(n) + b = O(f(n))$ (a e b constantes)
Vou mostrar que $af(n) + b \le c.f(n)$ para $n \ge N_0$
$af(n) + b \le af(n) + f(n)$
$= (a+1)f(n)$
$= c.f(n)$ //

Para que $af(n) + b = O(f(n))$ devem existir constantes positivas
c e N_0 tal que $af(n) + b \le c.f(n)$, $\forall n \ge N_0$
$a + \frac{b}{...} \le c$ $a + 0 \le c$ Portanto basta tomar

/ /

b) $O(f(n)) + O(f(n)) = O(f(n))$
Vou mostrar que $c_1.f(n) + c_2 f(n) \le c_3.f(n)$
$c_1 f(n) + c_2 f(n) = (c_1 + c_2) f(n)$ suponha $c_1 + c_2 \le c_3$
$\le c_3 f(n)$

c) $O(O(f(n))) = O(f(n))$
Seja $g(n) = O(f(n))$ e seja $h(n) = O(g(n))$
Vou mostrar que $h(n) = O(f(n))$
$g(n) \le c_1 f(n)$ para $n \ge N_0$ e $h(n) \le c_2.g(n)$ para $n \ge N_1$
$h(n) \le c_2.(c_1.f(n))$
$h(n) \le c_3.f(n)$
$h(n) = O(f(n))$ //

d) $O(f(n)).O(g(n)) = O(f(n).g(n))$
Seja $h(n) = O(f(n))$ e $y(n) = O(g(n))$
Vou mostrar que $h(n).y(n) \le c_3.f(n).g(n)$
Como $h(n) \le c_1 f(n)$ e $y(n) \le c_2 g(n)$
$c_1.f(n).c_2 g(n) = c_1.c_2 f(n).g(n)$
$= c_3.f(n).g(n)$ //

e) $O(f(n).g(n)) = f(n).O(g(n))$
Seja $y(n) = O(g(n))$
Vou mostrar que $f(n).y(n) \le c_3.(f(n).g(n))$
$y(n) \le c_1.g(n)$ para $n \ge N_0$
$f(n).y(n) \le f(n).c_1.g(n)$
$= c_1.f(n).g(n)$
$\le c_3.f(n).g(n)$ //

7) Prove ou disprove

a) $f(n) = O(g(n))$ implica que $\log(f(n)) = O(\log(g(n)))$, onde $\log(g(n)) > 0$ e $f(n) \geq 1$ para todo n sufi... grande

Seja $f(n) \leq c_1 \cdot g(n)$ vou provar que $\log(f(n)) \leq c_2 \cdot \log(g(n))$

$\log(f(n)) \leq \log(c_1 \cdot g(n))$
$= \log c_1 + \log(g(n))$
$= c_2 + \log(g(n))$
$\leq \log(g(n)) + \log(g(n))$
$= 2 \cdot \log(g(n))$
$= c_3 \cdot \log(g(n))$ //

b) $f(n) = O(g(n))$ implica que $2^{f(n)} = O(2^{g(n)})$

Seja $f(n) \leq c_1 \cdot g(n)$ para $n \geq n_0$

Vou provar que $2^{f(n)} \leq c_3 2^{g(n)}$ não é verdade.

Seja $f(n) = n^3$ e $g(n) = 2^n$

$n^3 \leq 1 \cdot 2^n$ para $\forall n \geq 1$

Mas não é verdade que $2^{n^3} \leq 1 \cdot 2^{2^n}$ pois

$2^{n^3} = 2^{3 \cdot n} = 8^n$ e $2^{2^n} = 4^n$

8) Seja k uma constante e $p(n) := \sum_{i=0}^{d} a_i n^i$ um polinômio em n de grau d, onde $a_d > 0$. Use as definições de notação assintótica para provar o seguinte.

a) Se $k \geq d$ então $p(n) = O(n^k)$

Vou provar que se $k \geq d$ então $\sum_{i=0}^{d} a_i n^i \leq c_1 \cdot n^k$ para $n \geq n_0$

$\sum_{i=0}^{d} a_i n^i = a_0 + a_1 n^1 + \dots + a_d n^d$
$\leq a_d n^d + a_d n^d + \dots a_d n^d$
$= (d+1) \cdot a_d n^d$ para $k \geq d$
$\leq \underbrace{(k+1) \cdot a_k}_{} n^k$
$= c_1 \cdot n^k$ //

b) Se $k > d$ então $p(n) = o(n^k)$

Vou provar que se $k > d$ então $\sum_{i=0}^{d} a_i n^i < c_1 \cdot n^k$

$\sum_{i=0}^{d} a_i n^i = a_0 + a_1 n + \dots + a_d n^d$
$\leq a_d n^d + a_d n^d + \dots + a_d n^d$
$= \underbrace{(d+1) a_d}_{} \cdot n^d$
$= c_1 \cdot n^d$ se $k > d$
$< c_1 \cdot n^k$ //

9) Prove que, para todo $n > 1$

a) $\sum_{i=1}^{n} i^k$ é $\Theta(n^{k+1})$

Vou provar que $c_1 n^{k+1} \leq \sum_{i=1}^{n} i^k \leq c_2 n^{k+1}$

primeiro vou provar que $\frac{1}{4} n^{k+1} \leq \sum_{i=1}^{n} i^k$ $\forall n > 1$

$\sum_{i=1}^{n} i^k = 1 + 2^k + \dots + n^k$ 2.2+
$= \sum_{i=1}^{\lfloor n/2 \rfloor} i^k + \sum_{i=\lfloor n/2 \rfloor+1}^{n} i^k$
$\geq \sum_{n/2+1}^{n} i^k$
$\geq \sum_{i=n/2+1}^{n} \left(\frac{n}{2}+1\right)^k$
$= \left(n - \frac{n}{2}\right) \cdot \left(\frac{n}{2}+1\right)^k$
$= n/2 \cdot \left(\frac{n^k}{2^k} + n + 1\right)$
$= \frac{1 n^{k+1}}{2 \cdot 2^k} + \frac{n^2}{2} + \frac{n}{2}$
$\geq \frac{1}{2 \cdot 2^k} n^{k+1}$

Vou provar que $\sum_{i=1}^{n} i^k \leq 1 \cdot i^k$

$\sum_{i=1}^{n} i^k \leq \frac{1}{2} n \cdot (1 + n^k)$ | $\leq \frac{1}{2} n^{k+1} + \frac{1}{2} n^{k+1}$
$= \frac{n + n^{k+1}}{2}$ | $= 1 \cdot n^{k+1}$
$= \frac{1}{2} n + \frac{1}{2} n^{k+1}$

b) $\sum_{i=1}^{n} \frac{i}{2^i} \leq 2$

Vou mostrar que $\sum_{i=1}^{n} \frac{i}{2^i} \leq 2$ para todo $n \geq 1$

$\sum_{i=1}^{n} \frac{i}{2^i} = \sum_{i=1}^{n} \sum_{j=i}^{n} \frac{1}{2^j} = \sum_{i=1}^{n} \frac{\left(\frac{1}{2}\right)^{n+1} - \left(\frac{1}{2}\right)^{i}}{\frac{1}{2} - 1} =$

$= -2 \cdot \left(\sum_{i=1}^{n} \left(\frac{1}{2}\right)^{n+1} - \sum_{i=1}^{n} \left(\frac{1}{2}\right)^{i}\right)$

$= -2 \cdot \sum_{i=1}^{n} \left(\frac{1}{2}\right)^{n+1} + 2\left(\sum_{i=0}^{n} \left(\frac{1}{2}\right)^{i} - 1\right)$

$= -2 \cdot n \cdot \left(\frac{1}{2}\right)^{n+1} + 2 \cdot \left(\frac{\left(\frac{1}{2}\right)^{n+1} - 1}{\frac{1}{2} - 1}\right) - 2$

$= -\frac{2n}{2^{n+1}} + \left(\frac{2}{2^{n+1}} - 2\right) \times -2 - 2$

$= \frac{-2n}{2^{n+1}} - \frac{4}{2^{n+1}} + 4 - 2$

$= \frac{-2n - 4}{2^{n+1}} + 2$

≤ 2 item c: Pg - 209

Questão 10 - Suponha que $f: \mathbb{N} \to \mathbb{N}$ e $K \in \mathbb{N}$ são tais que $e^{f(n)} = O(n^K)$. Mostre que $f(n) = O(\log n)$

Seja $e^{f(n)} \leq c_1 \cdot n^K$, vou mostrar que $f(n) \leq c_2 \cdot \log n$

$\log_e e^{f(n)} \leq \log_e c_1 n^K$

$f(n) \leq \log_e c_1 + \log_e n^K$

$f(n) \leq \log_e c_1 + K \cdot \log_e n$

$\leq \log n + K \log n$

$= 2K \cdot \log n$ //

Questão 11 - Determine quanto valem as expressões

a) $1 \cdot 2 + 2 \cdot 3 + 3 \cdot 4 + \dots + n \cdot (n+1)$

$\sum_{k=1}^{n} k(k+1) = \sum_{k=1}^{n} k^2 + k$

$= \sum_{k=1}^{n} k^2 + \sum_{k=1}^{n} k$

$\geq \frac{1}{2} n \cdot (1+n^2) + \frac{1}{2} n \cdot (1+n)$

233

$= \frac{1}{2}(n + n^3) + \frac{1}{2}(n + n^2)$

$= \boxed{n + \frac{1}{2} n^3 + \frac{1}{2} n^2}$

$\geq n^3 + \frac{1}{2} n^3 + \frac{1}{2} n^3$

$= 2n^3$

$\boxed{\geq 2n^3}$

12 - Resolva as recorrências abaixo.

a) $T(n) = 8T(n/2) + \Theta(n^2)$

Seja $T(n) = a$ para $n = 1$ — $\frac{n}{2^i} = 1$, $i = \lg n$

$T(n) = 8T(n/2) + n^2 \quad n \geq 2^1, 2^2$

$T(n) = 8T(n/2) + n^2$

$= 8^2 T(n/4) + 8(n/2)^2 + n^2$

$= 8^3 T(n/8) + 8^2(n/4)^2 + 8(n/2)^2 + n^2$

$= 8^i T(n/2^i) + \left(\frac{8}{4}\right)^2 n^2 + \frac{8}{4} n^2 + n^2 \dots$

$= 8^i T(n/2^i) + n^2 \cdot \sum_{k=0}^{i-1} 2^k$

$= 8^{\lg n} T(1) + n^2 \cdot (2^{\lg n} - 1)/(2-1)$

$= 2^{3\lg n} \cdot a + n^2 \cdot 2^{\lg n} - n^2$

$= n^3 \cdot a + n^3 - n^2$

$= (1+a) n^3 - n^2$

fórmula fechada: $T(n) = (1+a) n^3 - n^2$

Vou provar por indução que a fórmula fechada está correta

Passo: $n \geq 2, 2^2$

Hipótese: $T(n/2) = (1+a)(n/2)^3 - (n/2)^2$

$T(n) = 8T(n/2) + n^2$

$\overset{H.I}{\leq} 8 \cdot \left((1+a) \cdot (n/2)^3 - (n/2)^2\right) + n^2$

$= 8 \cdot (n^3 + an^3/8) - 8 \cdot \frac{n^2}{4} + n^2$

$= n^3 + an^3 - 2n^2 + n^2 = (1+a) n^3 - n^2$ //

base: $n = 1$

$T(1) = a = (1+a)1^3 - 1 = 1 + a - 1 = a$ //

OK

b) $T(n) = 2T(n/2) + \Theta(n^3)$

Teorema Mestre:

Seja $a = 2$, $b = 2$ e $f(n) = n^3$

1 Se $n^3 \le n^{\log_2 2 - \epsilon}$ X

2 Se $c\,n^1 \le n^3 \le c_1 \cdot n^1$ X

3 Se $n^3 \ge n^{1+\epsilon}$ $\epsilon = 1$ e $a\,f(n/b) \le c \cdot f(n)$ para $c < 1$

$2 \cdot \left(\frac{n^3}{8}\right) \le \frac{1}{4} \cdot n^3$ OK

então $T(n) = \Theta\, n^3$ //

Detalhado: $T(n) = a$ $n = 1$

$T(n) = 2T(n/2) + n^3$ para $n \ge 2^1, 2^2$

$T(n) = 2T(n/2) + n^3$

$= 2^2T(n/4) + 2(n/2)^3 + n^3$

$= 2^3T(n/8) + 2^2(n/4)^3 + 2(n/2)^3 + n^3$

$= 2^iT(n/2^i) + n^3 \cdot \sum_{k=0}^{i-1} \frac{2^k}{2^{3k}}$

$= 2^{\lg n} a + n^3 \cdot \sum_{k=0}^{\lg n - 1} 2^{k-3k}$

$= n.a + n^3 \cdot \sum_{k=0}^{\lg n - 1} 2^{-2k}$

$= n.a + n^3 \cdot \sum_{k=0}^{\lg n - 1} \frac{1}{4^k}$

$= na + n^3 \cdot \frac{\left(\frac{1}{4}\right)^{\lg n} - 1}{\frac{1}{4} - 1}$

$= na + n^3 \cdot \left(-\frac{4}{3}\right) \cdot \left(\frac{1}{n^2} - 1\right)$

$= na - \frac{4}{3}\frac{n^3}{n^2} + \frac{4}{3}n^3$

$= na - \frac{4}{3}n + \frac{4}{3}n^3$

$\frac{n}{2^i} = 1$

$n = 2^i$

$\boxed{\log n = i}$

Fórmula fechada: $T(n) = na - \frac{4}{3}n + \frac{4}{3}n^3$

base: $n = 1$

$T(n) = a = a - \frac{4}{3} + \frac{4}{3} = a$

Passo: $n \ge 2^1, 2^2$

Hipótese: $T(n/2) = a(n/2) - \frac{4}{3}(n/2) + \frac{4}{3}(n/2)^3$

$T(n) = 2(T(n/2)) + n^3$

$\overset{H.I}{\le} 2 \cdot \left(\frac{an}{2} - \frac{4}{3}\frac{n}{2} + \frac{4}{3}\frac{n^3}{2^3}\right) + n^3$

$= an - \frac{4}{3}n + \frac{1}{3}n^3 + n^3$

$= an - \frac{4}{3}n + \frac{4}{3}n^3$ //

c) $T(n) = 7T(n/3) + \Theta(n^2)$ $n^2 \le n^{\log_3 7}$ $1+\epsilon$

$n^2 > n^{\log_3 7}$

Pelo teorema Mestre

Seja $a = 7$, $b = 3$ e $f(n) = n^2$

e como $n^2 \ge 1 \cdot n^{\log_3 7 + \epsilon}$ para $\epsilon > 0$ e

$7\left(\frac{n}{3}\right)^2 \le \frac{7}{9}n^2$ para $c < 1$

então $T(n) = \Theta\, n^2$ //

Detalhado: $T(n) = a$ $n = 1$

$T(n) = 7T(n/3) + n^2$ para $n \ge 3^1, 3^2 \ldots$

$T(n) = 7T(n/3) + n^2$

$= 7^2T(n/9) + 7(n/3)^2 + n^2$

$= 7^3T(n/27) + 7^2(n/9)^2 + 7(n/3)^2 + n^2$

$= 7^{\log_3 n}T(1) + \frac{7^2n^2}{3^4} + \frac{7n^2}{3^2} + n^2$

$= 7^{\log_3 n} a + \left(\frac{7}{3^2}\right)^2 n^2 + \frac{7}{3^2}n^2 + n^2$

$= 7^{\log_3 n} a + n^2 \cdot \sum_{k=0}^{\log_3 n - 1} \left(\frac{7}{3^2}\right)^k$

$= 7^{\log_3 n} a + n^2 \cdot \left(\left(\frac{7}{3^2}\right)^{\log_3 n} - 1\right) \times -\frac{9}{2}$

$= 7^{\log_3 n} a + n^2\left(-\frac{9}{2} \cdot \frac{7^{\log_3 n}}{9^{\log_3 n}} + \frac{9}{2}\right)$

$= 7^{\log_3 n} a - n^2 \cdot \frac{7^{\log_3 n} \cdot 9}{2 \cdot n^2} + \frac{9n^2}{2}$

$= 7^{\log_3 n} a - 7^{\log_3 n} \cdot \frac{9}{2n} + \frac{9n^2}{2}$

$\frac{n}{3^i} = 1$

$n = 3^i$

$\log_3 n = i$

Passo: $n \ge 3^1, 3^2 \ldots$

Hipótese: $T(n/3) = 7^{\log_3(n/3)} a - 7^{\log_3(n/3)} \cdot \frac{9}{2\frac{n}{3}} + \frac{9}{2}\left(\frac{n}{3}\right)^2$

$T(n) = 7(T(n/3)) + n^2$

$\overset{H.I}{\le} 7 \cdot \left(7^{\log(n/3)} a - 7^{\log n/3} \cdot \frac{9 \cdot 3}{2n} + \frac{9n^2}{9 \cdot 2}\right) + n^2$

$= 7\left(7^{\log n - 1} a - 7^{\log n - 1} \cdot \frac{3^3}{2n} + \frac{n^2}{2}\right) + n^2$

$= 7^{\log n} a - 7^{\log n} \cdot \frac{9}{2n} + \frac{7n^2}{2} + n^2$

$= 7^{\log n} a - 7^{\log n} \cdot \frac{9}{2n} + \frac{9}{2}n^2$ //

Fórmula fechada: $7^{\log_3 n} a - 7^{\log_3 n} \frac{9}{2n} + \frac{9n^2}{2}$

$= n^{\log 7} - n^{\log 7} \cdot \frac{9}{2n} + \frac{9}{2}n^2$

Vou provar que a fórmula está correta

Prova por indução em n:

base: $n = 1$

$T(n) = a = 7^0 a - 7^0 \cdot \frac{9}{2} + \frac{9}{2} = a.$

d) $T(n) = T(\lfloor 9n/10 \rfloor) + \Theta(n)$

$T(n) = a \quad n = 1$

$T(n) = T(\lfloor 9n/10 \rfloor) + n \quad n > \frac{10}{9}^1, \frac{10}{9}^2 \ldots$

$= T(9^2 n/10^2) + 9n/10 + n$

$= T(9^3 n/10^3) + 9^2 n/10^2 + 9n/10 + n$

$= T(9^i n/10^i) + n . \sum_{k=0}^{i-1} (\frac{9}{10})^k$

$= T(1) + n . ((\frac{9}{10})^{\log_{10/9} n} - 1) * (-10)$

$= a - 10n . (\frac{9}{10})^{\log_{10/9} n} + 10n$

$= a - 10 . n . \frac{n^{\log_{10/9} 9}}{n^{\log_{10/9} 10}} + 10n$

$= a - 10 . n . n^{\log_{10/9} 9/10} + 10n$

$= a - 10n . n^{-1} + 10n = a - 10 + 10n$

$\frac{9^i n}{10^i} = 1$

$9^i n = 10^i$

$n = (\frac{10}{9})^i$

$\log_{10/9} n = \log_{10/9} (10/9)^i$

$\log_{10/9} n = i$

Importante: $\log_{10/9} 9/10 = -1$

pois, $(\frac{10}{9})^c = \frac{9}{10}$

$= (\frac{10}{9})^{-1} = \frac{9}{10}$

Vou provar que $T(n) = 10n + a - 10$

Prova por indução em n:

base: $n = 1$

$T(n) = a = 10.1 + a - 10 = a$

passo: $n \geq (\frac{10}{9})^1, (\frac{10}{9})^i \ldots$

Hipótese: $T(9n/10) = 10 . \frac{9n}{10} + a - 10$

$T(n) = T(9n/10) + n$

$\overset{H.I}{\leq} 10 . \frac{9n}{10} + a - 10 + n$

$= 9n + a - 10 + n$

$= 10n + a - 10$ //

$\frac{\quad}{10} \quad \frac{\quad}{10/9}$

Pelo teorema Master temos que $a = 1$ e $b = 10/9$ e $f(n) = n$

temos que $n \geq n^{\log_{10/9} 1 + \epsilon}$ para $\epsilon > 0$ e $n \geq 1$ e

$9n/10 \leq 9/10\, n$ para $n \geq 0$ e $c = 9/10 < 1$

então $T(n) = \Theta(n)$ //

Não sei

e) $T(n) = T(\lfloor n/3 \rfloor) + T(\lfloor 2n/3 \rfloor) + \Theta(n)$

$T(n) = a \quad n \geq 1/2$

$T(n) = T(n/3) + T(2n/3) + n \quad$ pala $n \geq \frac{3}{2}^1, \frac{3}{2}^2, \ldots$

$= T(n/3) + T(2n/3) + n$

$= T(n/3^2) + T(2 . \frac{n}{3^2}) + n/3 + T(\frac{2n}{3^2}) + T(\frac{2^2 n}{3^2}) + \frac{2n}{3} + n$

$= T(n/3^2) + 2T(\frac{2n}{3^2}) + n/3 + \frac{2n}{3} + n + T((\frac{2}{3})^2 n)$

$= T(\frac{n}{3^3}) + T(\frac{2n}{3^3}) + \frac{n}{3^2} + 2(T(\frac{2n}{3^3}) + T(\frac{2}{3} . \frac{2n}{3^2}) + \frac{2n}{3^2}) + T((\frac{2}{3})^2 \frac{n}{3}) + T((\frac{2}{3})^2 \frac{2n}{3}) + (\frac{2}{3})^2 n + \frac{n}{3} + \frac{2n}{3} + n$

$= T(\frac{n}{3^3}) + 3T(\frac{2n}{3^3}) + 3T(\frac{2^2}{3^3} n) + T((\frac{2}{3})^3 n) + \frac{n}{3^2} + \frac{2^2}{3^2} n + \frac{2^2}{3^2} n + \frac{n}{3} + \frac{2n}{3} + n$

$= T(\frac{n}{3^i}) + i\,T(\frac{2}{3^i} n) + i\,T(2^2 \frac{n}{3^i}) + T(\frac{2^3 n}{3^i}) + \sum_{k=0}^{i-1} \frac{n}{3^k} + \sum_{k=1} k . (\frac{2}{3})^k . n$

$\leq T((\frac{2}{3})^i n) + \sum_{k=0}^{i-1} \frac{n}{3^k} + \sum_{k=1}^{i-1} k . (\frac{2}{3})^k . n$

$= T(1) + n . (3^{\log_{3/2} n} - 1/2) + n . (\sum_{k=1}^{\log_{3/2} n - 1} \sum_{j=k}^{\log_{3/2} n - 1} (\frac{2}{3})^k)$

$= a + n . (n^{\log 3} - 1/2) + n . (\sum^{\log n - 1} (\frac{2}{3})^{\log n} - (\frac{2}{3})^k / (\frac{2}{3}) - 1)$

$= a + n^{\log 3 + 1}/2 - n/2 + n . (\sum_{k=1}^{\log n - 1} \frac{3}{n} + 3 . \sum_{k=1}^{\log n - 1} (\frac{2}{3})^k)$

$= a + n^{\log 3 + 1}/2 - n/2 + n[(\log_{3/2} n - 1) . (-\frac{3}{n}) + 3 . \sum_{k=0}^{\log n - 1} (\frac{2}{3})^k - 1$

$= a + n^{\log_{3/2} 3 + 1}/2 - n/2 - 1 - 3\log_{3/2} n + 3 + 3n((\frac{2}{3})^{\log_{3/2} n} - 1/-3)$

$= a + n^{\log 3 + 1}/2 - n/2 + 2 - 3\log_{3/2} n - 3^2 n(\frac{1}{n} - 1)$

$= a + n^{\log 3 + 1}/2 - \frac{n}{2} + 2 - 3\log_{3/2} n - 3^2 + 3^2 n$

$+ T(\frac{1}{3^{\log n}}) \log n\; T(\frac{2}{3^{\log_{3/2} n}} n)$

$\frac{2}{n^{\log_{3/2} 3}} . n$

$\frac{\log_3 3}{\log_3 3/2}$

/ /

f) $T(n) = 2T(n/4) + \sqrt{n}$

$T(n) = a \quad n = 1$

$T(n) = 2T(n/4) + \sqrt{n} \quad n \geq 4^1, 4^2 \ldots$

$T(n) = 2^2 T(n/4^2) + 2\sqrt{n/4} + \sqrt{n}$

$= 2^3 T(n/4^3) + 2^2\sqrt{n/4^2} + 2\sqrt{n/4} + \sqrt{n}$

$= 2^i T\left(\frac{n}{4^i}\right) + \sum_{k=0}^{i-1} 2^k \cdot \sqrt{n/4^k}$

$= 2^{\log_4 n} T(1) + \sum_{k=0}^{\log n - 1} n^{1/2} \cdot 2^k / 4^{\frac{1}{2}k}$

$= 2^{\log_4 n} \cdot a + n^{1/2} \cdot \sum_{k=0}^{\log n - 1} 2^k / 2^k$

$= n^{\log_4 2} \cdot a + \sqrt{n} \cdot (\log_4 n)$ // = Fórmula Fechada

$\frac{n}{4^i} = 1$

$n = 4^i$

$\log_4 n = \log_4 4^i$

$\log_4 n = i$

4

Vou provar que a Fórmula fechada está correta:

Prova por Indução em n:

base: $n = 1$

$T(n) = a = 1^{\log_4 2} \cdot a + \sqrt{1} \cdot \cancel{(\log_4 1)}^{0} = a$ ok

Passo: $n \geq 4^1, 4^2 \ldots$

Hipótese: $T(n/4) = (n/4)^{\log_4 2} \cdot a + \sqrt{n/4} \cdot \log_4(n/4)$

$T(n) = 2T(n/4) + \sqrt{n}$

$\overset{H.I.}{=} 2 \cdot \left((n/4)^{\log_4 2} \cdot a + \sqrt{n/4} \cdot (\log_4 n - 1)\right) + \sqrt{n}$

$= 2a\left(\frac{n^{\log_4 2}}{4^{\log_4 2}}\right) + 2 \cdot \frac{\sqrt{n}}{2} \cdot \log_4 n - 2\sqrt{n}/2 + \sqrt{n}$

$= 2a\left(n^{\log_4 2}/2\right) + \sqrt{n} \cdot \log_4 n - \sqrt{n} + \sqrt{n}$

$= a n^{\log_4 2} + \sqrt{n} \cdot \log_4 n$ //

NOME:

E-MAIL:

MATÉRIA:

PROFESSOR: SALA:

HORÁRIOS

HORÁRIO	SEGUNDA	TERÇA	QUARTA	QUINTA	SEXTA	SÁBADO

PROVAS/NOTAS

TRABALHOS

$2^k \cdot \frac{\sqrt{n}}{\sqrt{4^k}} \quad n^{1/2} \cdot \frac{2^k}{4^{k/2}}$

ANOTAÇÕES

/ /

13- Escreva um algoritmo que ordena uma lista de n itens dividindo-a em três sublistas de aproximadamente $n/3$ itens, ordenando cada sublista recursivamente e intercalando as três sublistas ordenadas. Analise seu algoritmo concluindo qual é o seu consumo de tempo.

Ordena (A, p, r)

1 se p < r
2 senão $q \leftarrow \lfloor r+p/3 \rfloor$
3 Ordena (A, p, q)
4 Ordena (A, q+1, 2q)
5 Ordena (A, 2q+1, r)
6 Intercala (A, p, r)

Tamanho da instância = $n = r - p + 1$

$T(n)$ = consumo do algoritmo

Linha	$T(n)$
1-2	$O(1)$
3,4,5	$T(n/3)$
6	$O(n)$

Recorrência:

$T(n) = 0 \quad n = 1$

$T(n) = 3T(n/3) + n \quad n \geq 3^1, 3^2, \ldots$

$T(n) = 3T(n/3) + n$

$= 3^2T(n/3^2) + 3 \cdot n/3 + n$

$= 3^3T(n/3^3) + 3^2 \cdot \frac{n}{3^2} + n + n$

$= 3^i T(n/3^i) + i n$

$= 3^{\log_3 n} T(1) + \log_3 n \cdot n$

$= 0 + n \log_3 n$ — Fórmula fechada

Vou provar que a fórmula está correta.

Prova por indução em n:

base: $n = 1$

/ /

$T(n) = 0 = 1 \cdot \log_3 1 = 0$

passo: $n \geq 3^1, 3^2$

Hipótese: $T(n/3) = n/3 \cdot \log_3 n/3$

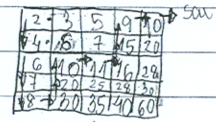
$T(n) = 3T(n/3) + n$

$\overset{H.I}{\leq} 3 \cdot (n/3 \cdot \log_3 n/3) + n$

$= n \cdot (\log_3 n - 1) + n$

$= n \log_3 n - n + n$

$= n \log_3 n$ //

14) 

Procura 12

Busca(A, x)

1 $i \leftarrow 1 \quad j \leftarrow 1$
2 enquanto $i < n$ e $x \geq A[i,j]$
3 se $x = A[i,j]$
4 então devolva "SIM"
5 senão $i \leftarrow i+1$
6 $i \leftarrow i-1$
7 enquanto $i \geq 1$ e $i \leq n$ e $j \geq 1$ e $j \leq n$
8 se $x = A[i,j]$ então devolva "SIM"
9 se $A[i,j] > x$
10 então $i \leftarrow i-1$
11 senão $j \leftarrow j+1$

Tamanho da instância: n

$T(n)$: consumo de tempo do algoritmo

Linha	$T(n)$
1	$O(1)$
2-5	$O(n)$
6	$O(1)$
7-11	$O(n)$
$T(n) = O(n)$	

Entre as linhas 7 e 11 os contadores andarão no máximo na diagonal por tratar-se de colunas e linhas em ordem não decrescente dessa forma percorrerão no máximo $2n$ elementos ou seja $O(n)$.

15- MAIOR(A, n) Original / /

max ← A[1]
Para i ← 2 até n faça
 se max < A[i]
  então max ← A[i]

$T(n)$ = nº de comparações
$T(n) = \Theta(n)$

MAIORDIV(A, p, r) [3|5|10|8|9] $5+1/2 = 3$

se $p = r$
 então devolva A[r]
 senão $q \leftarrow \lfloor r+p/2 \rfloor$
  x ← MAIORDIV(A, p, q)
  y ← MAIORDIV(A, q+1, r)
  se x > y
   então devolva x
   senão devolva y

Tamanho da instância: $n = r-p+1$
$T(n)$ = número de comparações
Recorrência

$T(n) = 0 \quad n=1$
$T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + 1$
$T(n) = 2T(n/2) + 1$
$= 2^2 \cdot T(n/4) + 1 + 1$
$= 2^3 \cdot T(n/8) + 1 + 1 + 1$
$= 2^i \, T(n/2^i) + i$
$= 2^{\log n} T(1) + \log n$
$= n \cdot 0 + \lg n$
$= \lg n$

$i = \lg n$

Algoritmo original faz $O(n)$ comparações enquanto que a versão divisão e conquista faz no máximo $\lg n$ comparações.

LISTA 5- ALAIR / /

Questão 2. No pior caso o vetor está ordenado em ordem crescente. Dessa forma sua pilha de recursão se comportaria da seguinte forma:

$$\left.\begin{array}{ll} \text{QuickSort}(A, p, r) & = r-p+1 \\ \text{QuickSort}(A, p, r-1) & = r-1-p+1 = r-p \\ \text{QuickSort}(A, p, r-2) & = r-2-p+1 = r-p-1 \\ \vdots & \vdots \\ \text{QuickSort}(A, p, p) & p-p+1 = 1 \end{array}\right\} n$$

O espaço necessário para armazenar essa pilha de recursão será $r(n)$

Modificação

QuickSort(A, p, r)
 enquanto p < r faça
  q ← PARTICIONE(A, p, r)
  se q - p > r - q
   então QuickSort(A, q+1, r)
    r ← q
   senão QuickSort(A, p, q)
    p ← q+1

Justificativa
Pilha de recursão: $k = \log_2 n$

$\text{QuickSort}(A, p_1, r_1) = r-p+1 = n$
$\text{QuickSort}(A, p_2, r_2) = r_2 - p_2 + 1 = n/2$
$\text{QuickSort}(A, p_3, r_3) = r_3 - p_3 + 1 = n/2^2$
$\vdots$
$\text{QuickSort}(A, p_k, r_k) = r_k - p_k + 1 = \frac{n}{2^k} = 1$

Sendo a altura da pilha k, então $\frac{n}{2^k} = 1 \quad n = 2^k$
$k = \log_2 n$
O espaço diminuirá para $O(\log_2 n)$.

Questão 3-

X ← variável aleatória que representa a execução da linha 4

$X_i = \begin{cases} 1 & \text{se a linha 4 executar pelo menos } i \text{ vezes} \\ 0 & \text{cc} \end{cases}$

$X_i = Pr\{X_i = 1\} = \frac{1}{i}$ = Probabilidade de $v[i]$ ser máximo em $v[1..i]$

$E[X] = \sum_{i=1}^{n} X_i = \sum_{i=1}^{n} Pr\{X_i = 1\} = \sum_{i=1}^{n} \frac{1}{i} < 1 + \lg n$

$E[X] = O(\lg n)$

Questão 4-

X - variável aleatória que representa a execução das linhas 5 e 8

$X_i \begin{cases} 1, & \text{se as linhas 5 e 8 executam pelo menos } i \text{ vezes} \\ 0 & \text{cc} \end{cases}$

$X_i = Pr\{X_i = 1\}$ = Probabilidade da linha 5 executar + Probabilidade da linha 8 executar

Probabilidade da linha 5 executar = Probabilidade de $v[i]$ ser máximo em $v[1..i]$

Probabilidade linha 5 $= \frac{1}{i}$

Probabilidade linha 8 $= \frac{1}{i}$, já que $v[i]$ não é o maior

$Pr\{X_i = 1\} = \frac{1}{i} + \frac{1}{i} = \frac{2}{i}$

$E[X] = \sum_{i=1}^{n} X_i = \sum_{i=1}^{n} \frac{2}{i} = 2 \cdot \sum_{i=1}^{n} \frac{1}{i} \leq 2 . (\lg n + 1)$

$E[X] = O(\lg n)$

Questão 5-

X ← variável aleatória que representa nº de execuções da linha 6.

$X_i = \begin{cases} 1 & \text{se a linha 6 é executada pelo menos } i \text{ vezes} \\ 0 & \end{cases}$

$X_i = Pr\{X_i = 1\} = \frac{1}{i}$

$E[X] = \sum_{i=1}^{n} X_i = \frac{1}{1} + \frac{1}{2} + \dots + \frac{1}{n} \leq \lg n + 1$ //

$E[X] = O(\lg n)$

Y ← variável aleatória que representa nº de execuções da linha 7

$Y_i = \begin{cases} 1 & \text{se a linha 7 é executada} \\ 0 & \text{cc} \end{cases}$

$Y_i = Pr\{Y_i = 1\} = \frac{1}{i}$

$E[Y] = \sum_{i=1}^{n} Y_i = \frac{1}{1} + \frac{1}{2} + \frac{1}{n} \leq \lg n + 1$

$E[Y] = O(\lg n)$

PROVA 2002/1

Questão 1-

```
ORDENA(A, n)
1   j ← n-1   min ← n
2   enquanto j >= 0
3   K ← BUSCAMIN(A, 0, j)
4     Se K = 0
5       então FLIP(j, A, n)
6       senão se K > 0 e K < j
7             FLIP(K, A, n)
8             FLIP(j, A, n)
9   j ← j - 1
```

tamanho da instância: n

$T(n)$ = nº de chamadas de FLIP

linha	$T(n)$
1	$O(1)$
2-4	$O(n)$
5	$O(n)$
7-8	$2O(n)$
9	$O(n)$

consumo total $= T(n) = 2O(n) = O(n)$