

Gabarito

①

a) CRCW forte

para $0 \leq i \leq n-1$ faça em paralelo

se $A[i]$ então

$I[i] := i$

senão

$I[i] := n+1$

para $0 \leq i \leq n-1$ faça em paralelo

$K[i] := -I[i]$

$K := -K$

complexidade: Tempo: $O(1)$

Processadores: $O(n)$

Memória: $O(n)$

	0	1	2	3	4	5	6	7
A	F	F	F	F	T	T	F	T

I	9	9	2	9	4	5	9	7
---	---	---	---	---	---	---	---	---

-I	-9	-9	-2	-9	-4	-5	-9	-7
----	----	----	----	----	----	----	----	----

$K := -2$ $\rightarrow K := (-1) * (-2)$

$K := 2$

b) escreva falso

(2)

UEMS

UNIVERSIDADE ESTADUAL DE MATO GROSSO DO SUL

para $0 \leq i \leq n-1$ faça em paralelo

[$C[i] := A[i]$

para $0 \leq i \leq n-1$ faça em paralelo

[se $A[i] = \text{True}$ então

para $0 \leq j \leq n-1, j < i$ faça em paralelo

[se $A[j] = \text{true}$ então

[$C[i] := \text{False}$

para $0 \leq i \leq n-1$ faça em paralelo

[se $C[i] = \text{true}$

[$K := i$

Complexidades: Tempo: $O(1)$

Processadores: $O(n^2)$

Costo: $O(n^2)$

	0	1	2	3	4	5	6	7
A	F	T	F	T	T	F	F	T

Exos: Inicialização

cada processador P_i define:

C	F	T	F	T	T	F	F	T
---	---	---	---	---	---	---	---	---

Passo 2: Eliminação Final

P₂: Verifica índices menores que 2.

A[1] = F // não encontrou true anterior

P₄: verifica

A[1] = F

A[2] = T

A[3] = F

} * Encontrou um true anterior

C[4] := F

P₅: verifica

A[1] = F

A[2] = T

A[3] = F

A[4] = T

} existe true anterior

C[5] := F

P₈: Verifica

A[1]

⋮

A[7]

} existe true anterior

C[8] := F

Após a eliminação, tem-se

C	F	T	F	F	F	F	F	F
---	---	---	---	---	---	---	---	---

(4)

Passo 3: Execução Concorrente

Cada processador faz: se $C[i] = \text{true}$ então

$$K := i$$

semente P2 encerra a escrita

$$K = 2$$

C) CREW

para $1 \leq i \leq n$ faça em paralelo

se $A[i] = \text{true}$ então

$$B[i] := 1$$

senão

$$B[i] := \infty$$

para $1 \leq n \leq \log n$ faça

para $1 \leq j \leq \frac{n}{2^n}$ faça em paralelo

$B[j] := \min(B[2j-1], B[2j])$

return $B[1]$

Complexidades: Tempo: $O(\log n)$

(5)

Processadores: $O(n)$

Custo: $O(n \log n)$

2) EREW

para $0 \leq i \leq n-1$ faça em paralelo

se $A[i] = \text{true}$ então

~~$B[i] := i$~~
 $B[i] := i$

senão

$B[i] := \infty$

para $1 \leq k \leq \log n$ faça

para $j = 0$ até $\frac{n}{2^k}$ faça em paralelo

$\text{esq} := B[(2j-1) * 2^{(n-1)}]$

$\text{dir} := B[(2j) * 2^{(n-1)}]$

$B[(2j) * 2^{(n-1)}] := \min(\text{esq}, \text{dir})$

$K := B[n]$

Complexidade

(6)

Tempo: $O(\log n)$

processadores: $O(n)$

custo: $O(n \log n)$

e) EREW ativo

seja $p = \frac{n}{\log n}$ processadores.

tamanho: $\log n$

para $0 \leq i \leq p-1$ faça em paralelo

$V[i] := \infty$

// varre sequencialmente o bloco de tamanho $\log n$

início: $i * \text{tamanho}$

fim: $\text{início} + \text{tamanho} - 1$

para início $\leq j \leq \text{fim}$ faça

se $A[j] = \text{true}$ então

$V[i] := j$

// como $p = \frac{n}{\log n}$ a árvore terá $\log(p)$ etapas.

para $1 \leq s \leq \log p$ faça

para $0 \leq i \leq \left(\frac{p}{2^s}\right) - 1$ faça em paralelo

$V[i] := \min(V[2i], V[2i+1])$

resultado: $V[0]$.

Teorema de Brent

(7)

// fim da funcao ImprimeR

para ser ótimo, precisamos ter custo: $O(n)$.

Se usarmos $p = n$ processadores com tempo $O(\log n)$
o custo será $O(n \log n)$

Aplicando o teorema para $p = \frac{n}{\log n}$

$$T_p = O\left(\frac{n}{\frac{n}{\log n}} + \log n\right) = O(\log n + \log n) = O(\log n)$$

$$\text{custo: } T_p \times p = O(\log n) \times O\left(\frac{n}{\log n}\right) = O(n)$$

Questão 2 $P_0 = 4, P_1 = 1, P_2 = 0, P_3 = 1, m = 0$
 $ml = 2$

Questão 3 3 itens ($m = 3$)
3 estágios ($k = 3$)

$$S_1(x) = x + 1$$

$$S_2(x) = x * 2$$

$$S_3(x) = x - 3$$

Entrada: $x = [1, 2, 3]$

cada item passa por:

$$S_3(S_2(S_1(x)))$$

item 1

$1 \rightarrow S_1 \rightarrow 2$

$2 \rightarrow S_2 \rightarrow 4$

$4 \rightarrow S_3 \rightarrow 1$

item 2

$2 \rightarrow S_1 \rightarrow 3$

$3 \rightarrow S_2 \rightarrow 6$

$6 \rightarrow S_3 \rightarrow 3$

item 3

$3 \rightarrow S_1 \rightarrow 4$

$4 \rightarrow S_2 \rightarrow 8$

$8 \rightarrow S_3 \rightarrow 5$

(8)

Wago: $Y = [1, 3, 5]$

Nº de Passos: $m + n - 1 = 3 + 3 - 1 = 5$

Tempo $T = 1$

Estágio 1: Item 1

$S_1(1) = 2$

Item	B1	B2	B3
1	2	-	-

Tempo $T = 2$

Estágio 1: Item 2

$S_1(2) = 3$

Estágio 2: Item 1

$S_2(2) = 4$

Buffer:

Item	B1	B2	B3
1	2	4	-
2	3	-	-

Tempo $T = 3$

Estágio 1: Item 3

$S_1(3) = 4$

Estágio 2: Item 2

$S_2(3) = 6$

Estágio 3: Item 1

$S_3(4) = 1$

Saída: $Y[1] = 1$

Buffer:

Item	B1	B2	B3
1	2	4	1
2	3	6	-
3	4	-	-

(9)

Tempo $T=4$ Estágio 2: Item 3
 $S_2(4)=8$ Estágio 3: Item 2
 $S_3(6)=3$ Saída: $Y[2]=3$ Buffers:

Item	B1	B2	B3
1	2	4	1
2	3	6	3
3	4	8	-

Tempo $T=5$ Estágio 3: Item 3
 $S_3(8)=5$ Buffers:

Item	B1	B2	B3
1	2	4	1
2	3	6	3
3	4	8	5

Saída: $Y[3]=5$