UNIVERSIDADE FEDERAL DE MINAS GERAIS
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

PAA – Prova 3
2021/1

Bruno Augusto Alemão Monteiro
Matrícula: 2021665695

Belo Horizonte
09/08/2021

QUESTÃO 1

Bruno Alemão -

Questão 1

1) Par por blocos { a) Trecho com apenas "0" ou "1"
↳ comprimento par { b) Comprimento deve ser par

Ex: 0000 11 00 ok

0000 1111 00 ×
ok ok

Dado: vetor b de n posições com dígitos binários,

Determinar: Maior subsequência contígua par por blocos

$T(n) \in O(n)$

Caso base: entrada de duas posições. Se $b[1] == b[2]$, então é contígua par por blocos.

```
MSCPPB(b[ ], n)      n >= 2
Se  n == 2
 |  se b[1] == b[2]
 |   |  ret  V[ ]
Senão
 |  temp = {b[1]}
 |  subvetor = ∅
 |  resposta = ∅
 |  para i = 2 até n
 |   |  se b[i] == b[i-1]
 |   |   |  temp = temp U {b[i]}
 |   |  senão
 |   |   |  se temp.comprimento é par
 |   |   |   |  subvetor = subvetor U {temp}
 |   |   |   |  temp = ∅
 |   |   |  senão
 |   |   |   |  temp = temp - {b[i]}
 |   |   |   |  subvetor = subvetor + temp
 |   |   |   |  temp = ∅
 |   |  se subvetor.comprimento > resposta.comprimento
 |   |   |  resposta = subvetor
 |  ret resposta
```

QUESTÃO 2 a

Bruno Alemão

Q2 n tarefas

2 grupos de tarefas { 1° quinzena / 2° quinzena -> Dividir as tarefas de forma que duração total p/ cada quinzena seja similar.

i-ésima tarefa -> duração d_i

Conjunto $\{1, \ldots, n\}$ -> Q_1, Q_2 que minimize $\left| \sum_{i \in Q_1} d_i - \sum_{i \in Q_2} d_i \right|$

Guloso: ordenar decrescente (maior p/ o menor) e adicionar onde soma está menor.

a) $n = 7$; $Q_1 = Q_2 = \emptyset$

1 2 3 4 5 6 7

$d = [1, 11, 8, 5, 7, 3, 11]$ -> ordenado: $[11, 11, 8, 7, 5, 3, 1]$

Inicialmente $\sum_{i \in Q_1} d_i = \sum_{i \in Q_2} d_i = 0$

Iteração 1: $d[1] = 11 \rightarrow Q_1 = Q_1 \cup \{d[1]\} = [11]$ $\Sigma = 11$

2: $d[2] = 11 \rightarrow Q_2 = Q_2 \cup \{d[2]\} = [11]$ $\Sigma = 11$

3: $d[3] = 8 \rightarrow Q_1 = [11, 8]$ Σ 19

4: $d[4] = 7 \rightarrow Q_2 = [11, 7]$ Σ 18

5: $d[5] = 5 \rightarrow Q_2 = [11, 7, 5]$ Σ 23

6: $d[6] = 3 \rightarrow Q_1 = [11, 8, 3]$ Σ 22

7: $d[7] = 1 \rightarrow Q_1 = [11, 8, 3, 1]$ Σ 23

$$\left| \sum_{i \in Q_1} d_i - \sum_{i \in Q_2} d_i \right| = 0$$

$Q_1 = [11, 8, 3, 1]$ Foi encontrada uma solução ótima

$Q_2 = [11, 7, 5]$

QUESTÃO 2 b

Q2 - b)

b) Exemplo para o qual o algoritmo não funciona

$d = [9, 8, 7, 6, 5]$

$1 \rightarrow d[1] = 9 \rightarrow Q_1 = [9] \quad \Sigma\, 9$

$d[2] = 8 \rightarrow Q_2 = [8] \quad \Sigma\, 8$

$d[3] = 7 \rightarrow Q_2 = [8, 7] \quad \Sigma\, 15$

$d[4] = 6 \rightarrow Q_1 = [9, 6] \quad \Sigma\, 15$

$d[5] = 5 \rightarrow Q_1 = [9, 6, 5] \quad \Sigma\, 20$

Diferença = 5

A solução ótima seria $\begin{cases} Q_1 = [9, 8] : \Sigma\, 17 \\ Q_2 = [7, 6, 5] : \Sigma\, 18 \end{cases}$ Diferença = 1

Questão 3 a

Bruno Allemão

Q3 $S(i, j, K)$ bool

$\downarrow$

partição i primeiros tarefas

j soma dos elementos do primeiro conjunto

K soma dos elementos do segundo conjunto

a) como minimizar $\left| \sum_{i \in Q_1} d_i - \sum_{i \in Q_2} d_i \right|$

Se $S(i, j, k) =$ Verdadeiro $: \sum_{n=1}^{i} d_n = j + k$

$[\underbrace{9, 8,}_{j=17} \mid_{i} \underbrace{7, 6, 5}_{k=18}] \rightsquigarrow S(i=2, j=17, k=18) =$ Verdadeiro

e $\sum_{n=1}^{i} d_n = 17 + 18 = 35$

- Se estamos particionando o vetor d em dois conjuntos Q_1 e Q_2, Q_1 deverá ter ($\underline{a}$) elementos e Q_2, $(n - a)$ elementos.

- $S(i, j, K)$ só retorna verdadeiro se a soma de todos os n elementos for igual a $j + k$.
- Queremos minimizar $|j - K|$.

→ A solução recursiva envolveria somar para todas as partições possíveis os valores de j e K e escolher aquela partição que minimize $|j - K|$.

→ Idealmente gostaríamos que $j = K = \dfrac{\sum_{n=1}^{n} d_n}{2}$

- Para o caso base, a entrada teria 2 elementos e cada elemento iria para um conjunto

Questão 3 b e c

b) Dado entrada de $n \geq 2$ elementos

$$S(i, j, k) = \begin{cases} \text{Verdadeiro}, & \text{se } n == 2 \\ \text{Verdadeiro}, & \text{se } j + k == \sum_{r=1}^{n} d_r \\ \text{Falso}, & \text{se } j + k \neq \sum_{r=1}^{n} d_r \end{cases}$$

c) Recursivamente, as possibilidades de soma são obtidas

ou $\begin{cases} \text{incluindo o i-ésimo elemento no 1° grupo} \\ \text{incluindo no 2°} \end{cases}$

Então para cada elemento, temos 2 possibilidades.

Para n elementos, temos $2 \times 2 \ldots 2$ possibilidades (n vezes)

Então complexidade é da ordem $O(2^n)$