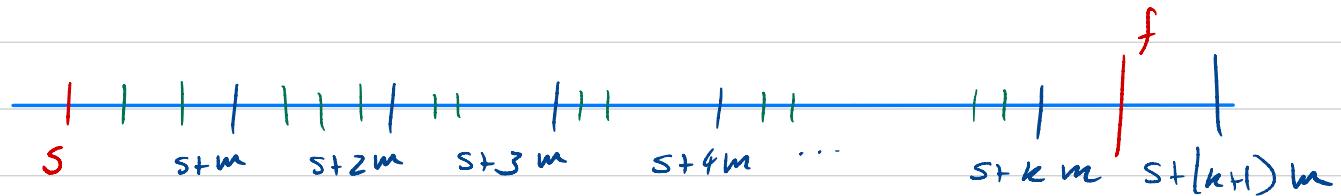

Práctica 8



Q1

16.2-4



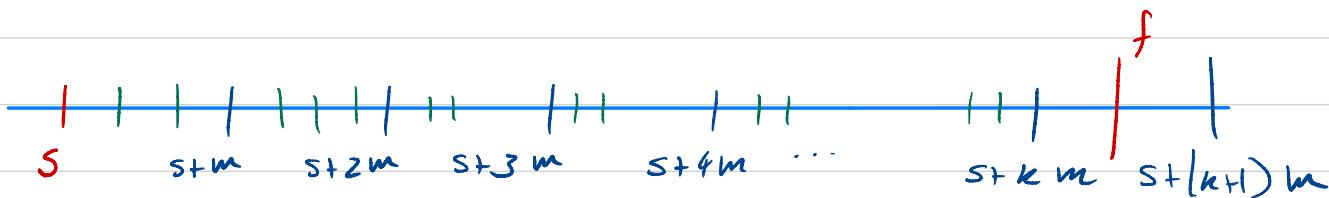
- A cada m milhas o Prof. Gekko deve abastecer-se de água.

Os postos de abastecimento estão indicados em verde.

- Problema: Em quantos postos de abastecimento deve o Prof. Gekko parar de modo a minimizar o nº de paragens?

Q1

16.2-4



- A cada m milhas o Prof. Gekko deve abastecer-se de água.

Os postos de abastecimento estão indicados em verde.

- Problema: Em \bar{x} postos de abastecimento deve o Prof. Gekko parar do modo a minimizar o nº de paragens?

- Formalização:

- Input: $\cdot m$ - distância máxima que o Prof. Gekko pode percorrer sem abastecer

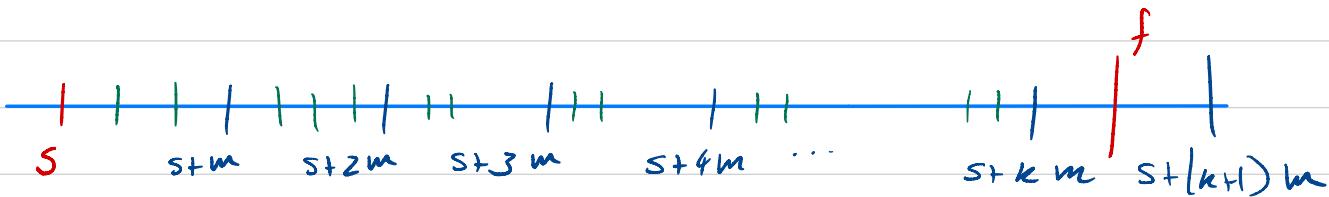
- $\cdot \bar{x}[1..n]$ - distâncias dos postos de abastecimento

- $\cdot s$ e f - distâncias de inicio e de fim

- Output - $\bar{g}[1..j]$ - distâncias dos postos de abastecimento escolhidos

Q1

16.2-4



- Input:
 - m - distância máxima que o Prof. Becko pode percorrer sem abastecer
 - $\vec{x}[1..n]$ - distâncias dos postos de abastecimento
 - s e f - distâncias de início e de fim

- Output - $\vec{t}[1..k]$ - distâncias das postos de abastecimento escolhidos

Escolha Greedy

$$g^* = \max \left\{ \vec{x}[k] \mid \vec{x}[k] \leq m \right\}$$

Subproblema:

$$(\vec{x}, m, s', f) \text{ onde } s' = g^*$$

Solução Óptima $y^* := \text{WaterFill}(\vec{x}, m, s', f)$
já pendedo

Q1

16.2-4Input: (\vec{x}, m, s, f) Output: $\vec{y} \subseteq \vec{x}$ Escolha Greedy

$$y^* = \max \left\{ \vec{x}[k] \mid \vec{x}[k] \leq m \right\}$$

- Há que provar que y^* faz parte de uma solução óptima.

Seja \vec{y} uma solução óptima; temos dois casos a considerar:

- $y^* \in \vec{y}$ (nada provar)
- $y^* \notin \vec{y}$

- Suponhamos que $y^* \notin \vec{y}$. Temos de construir a partir de \vec{y} uma outra solução óptima \vec{y}' tal que $y^* \in \vec{y}'$.

Q1

16.2-4

Escolha Greedy

$$j^* = \max \left\{ \vec{x}[k] \mid \vec{x}[k] \leq m \right\}$$

- Há que provar que j^* faz parte de uma solução óptima.

Sja \vec{j} uma solução óptima; temos dois casos a considerar:

- $j^* \in \vec{j}$ (nada mover)
- $j^* \notin \vec{j}$

- Suponhamos que $j^* \notin \vec{j}$. Temos de construir a partir de \vec{j} uma outra solução óptima \vec{j}' tal que $j^* \in \vec{j}'$.
Para tal basta trocar o 1º elemento de \vec{j} por j^* .

Formalmente:

$$\vec{j} = \langle j_1, j_2, \dots, j_j \rangle$$

$$\vec{j}' = \begin{matrix} \downarrow \\ \langle j^*, j_2, \dots, j_j \rangle \end{matrix}$$

Q1

16.2-4

- Suponhamos que $\vec{g}^* \notin \vec{J}$. Temos de construir a partir de \vec{J} uma outra solução óptima \vec{J}' tal que $\vec{g}^* \in \vec{J}'$.
Para tal basta trocar o 1º elemento de \vec{J} por \vec{g}^* .
Formalmente:

$$\vec{J} = \langle j_1, j_2, \dots, j_l \rangle$$



$$\vec{J}' = \langle \vec{g}^*, j_2, \dots, j_l \rangle$$

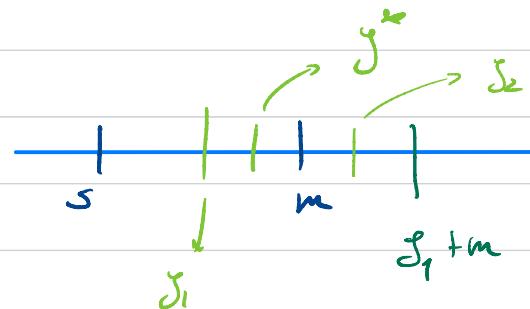
- Há ainda \vec{g} menor que \vec{g}^* :

- $|\vec{g}| = |\vec{g}'|$ ✓ (por construção)

- \vec{g}' é solução factível: $\forall 1 \leq i \leq l-1, \vec{g}'[i+1] \leq \vec{g}'[i] + m$

$i > z$ $\vec{g}'[i+1] = \vec{g}[i+1] \leq \vec{g}[i] + m$ $= \vec{g}'[i] + m$) porque \vec{g} é factível

$z=1$ $\vec{g}'[2] = \vec{g}[2] \geq \vec{g}[1] + m$
 $\geq \vec{g}^* + m$
 $= \vec{g}'[1] + m$



Q1

16.2-4WaterFill(\vec{x} , m, s, f)

```
let L = emptyList();
let next = s + m;
```

```
for i=1 to  $\vec{x}$ .size()
| if ( $\vec{x}[i] > next$ )
| | L.add ( $\vec{x}[i]$ )
| | next =  $\vec{x}[i]$  + m
```

return L

$$y^* = \max \left\{ \vec{x}[k] \mid \vec{x}[k] \leq m \right\}$$

Sub problema:

(\vec{x} , m, s', f) onde $s' = y^*$

Solução Óptima $y^* \leftarrow \text{WaterFill}(\vec{x}, m, s', f)$
 appended

Complexidade: $\Theta(n)$

Q2

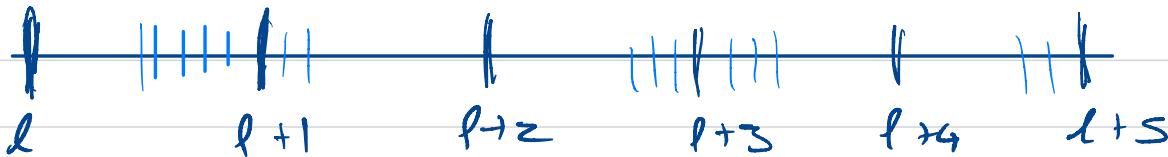
16.2.5



- Input: $\vec{x}[1..n]$ n pontos na recta real
 - Output: $\vec{y}[1..k]$ k pontos na recta real tais que:
 - ↓ Visão. $\exists 1 \leq j \leq k$. $\vec{x}[i] \in [\vec{y}[j], \vec{y}[j]+1]$
- Queremos encontrar uma solução \vec{y} tão pequena quanto possível.

Q2

16.2.5



Find Intervals (\vec{x})

- Input: $\vec{x}[1..n]$ n pontos na recta real
 - Output: $\vec{y}[1..k]$ k pontos na recta real tais que:
 - ↓ Visão. $1 \leq j \leq k$. $\vec{x}[i] \in [\vec{y}[j], \vec{y}[j+1]]$
- Queremos encontrar uma solução \vec{y} tão pequena quanto possível.

Escolha Greedy

$$y^* = \min \left\{ \vec{x}[i] \mid 1 \leq i \leq n \right\}$$

Subproblema: Find Intervals (\vec{x}')

com \vec{x}' vetor \vec{y} contém todos os elementos de \vec{x} maiores que y^*

Q2

16.2.5

Escolha Greedy

$$j^* = \min \{ \vec{x}[i] \mid 1 \leq i \leq n \}$$

Subproblema: Find Intervals (\vec{x}^i)

com \vec{x}^i vetor j contém todos os elementos
de \vec{x} maiores que $j+1$

- Há que provar que j^* faz parte de uma solução óptima.

Seja \vec{j} uma solução óptima; temos dois casos a considerar:

- $j^* \in \vec{j}$ (nada provar)
- $j^* \notin \vec{j}$

- Suponhamos que $j^* \notin \vec{j}$. Temos de construir a partir
de \vec{j} uma outra solução óptima \vec{j}' tal que $j^* \in \vec{j}'$.

Para tal basta trocar o 1º elemento de \vec{j} por j^* .

Formalmente:

$$\vec{j} = \langle j_1, j_2, \dots, j_j \rangle$$

$$\vec{j}' = \begin{matrix} \Downarrow \\ \langle j^*, j_2, \dots, j_j \rangle \end{matrix}$$

Q2

16.2.5

- Há ainda \vec{g}' que é solução \vec{g} :

- $|g'_i| = |g_i^*| \quad \checkmark \quad (\text{por comutatividade})$

- \vec{g}' é uma solução factível:

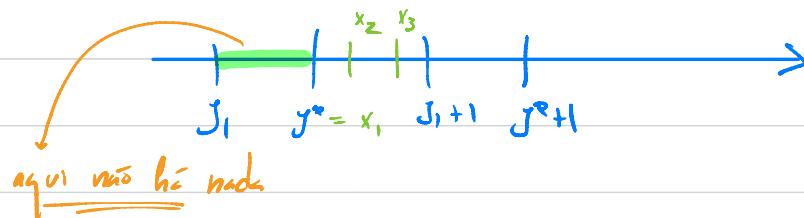
$$\forall i \in \mathbb{n}, \exists j_1 \leq j \leq k, \vec{x}[i] \in [\vec{g}[j], \vec{g}[j+1]]$$

- Trocamos o intervalo $[j_1, j_1+1]$ pelo intervalo $[j^*, j^*+1]$

- Como $[j_1, j_1+1]$ tem de incluir o elemento j^* , concluímos que: $j_1 \leq j^*$

Além disso, o intervalo $[x_1, j^*]$ não inclui nenhum elemento de \vec{x} .

Concluímos portanto que ao trocare $[j_1, j_1+1]$ por $[j^*, j^*+1]$ não deixamos de incluir nenhum ponto de \vec{x} .



Q2

16.2.5

Compute Intervals (\vec{x})

```
let L = emptyList();
let last =  $\vec{x}[1]$ 
for i=2 to  $\vec{x}.size()$ 
| if ( $\vec{x}[i] > last + 1$ )
| | L.add( $\vec{x}[i]$ )
| | last :=  $\vec{x}[i]$ 
return L
```

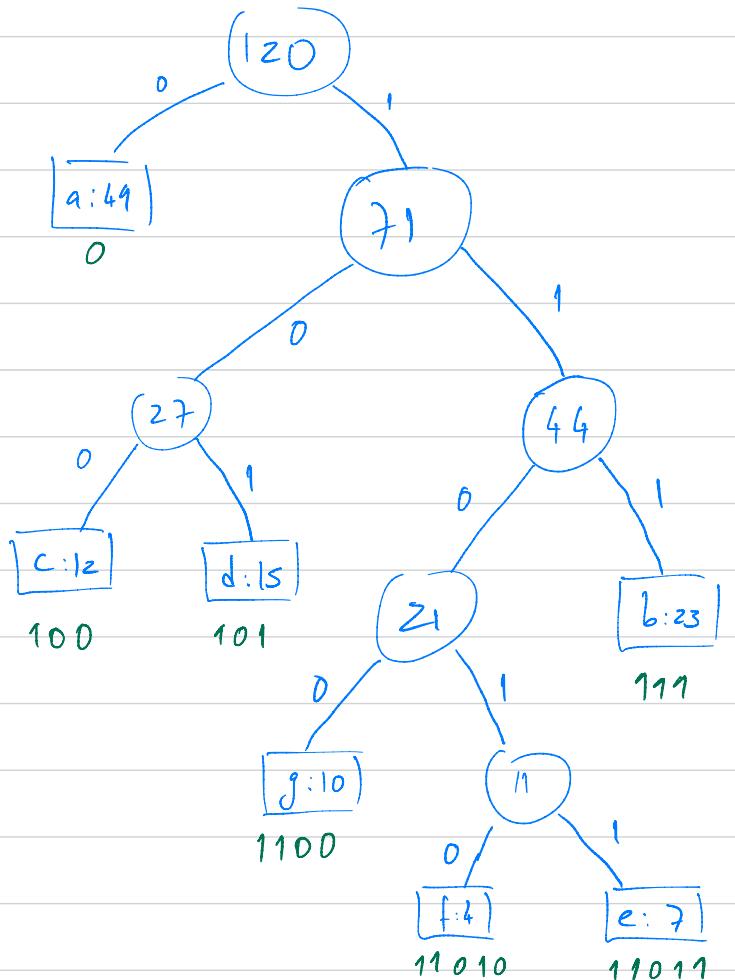
$\Theta(n)$

onde $n = \underline{\vec{x}.size()}$

Q3

T2 08/09 II.3

a:49, b:23, c:12, d:15, e:7, f:4, g:10



Q4

RZ 08/09 II.3

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
a _i	00	06	10	13	17	20	23	25	28	31	33	36	38	41	43	50	53	
d _i	5	5	3	6	1	3	2	6	2	4	5	3	3	4	4	6	2	

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
a _i	00	06	10	13	17	20	23	25	28	31	33	36	38	41	43	50	53	
d _i	5	5	3	6	1	3	2	6	2	4	5	3	3	4	4	6	2	
f _i	5	11	13	19	18	23	25	31	30	35	38	39	41	45	47	56	55	

	1	2	3	5	4	6	7	9	8	10	11	12	13	14	15	17	16	
a _i	00	06	10	17	13	20	23	28	25	31	33	36	38	41	43	53	50	
d _i	5	5	3	1	6	3	2	2	6	4	5	3	3	4	4	2	6	
f _i	5	11	13	18	19	23	25	30	31	35	38	39	41	45	47	55	56	

$$x = \{1, 2, 5, 6, 7, 9, 10, 12, 14, 17\}$$

Q5

Tz 14/15 II.b

① Ordenemos os n objectos por ordem decrescente (\leq) de valor por unidade de peso

② Seguindo a ordem estabelecida, vamos colocando 50% de cada objecto em cada uma das caixas

③ Seja i o índice do objecto tal que p_i/z excede a capacidade residual das duas caixas.

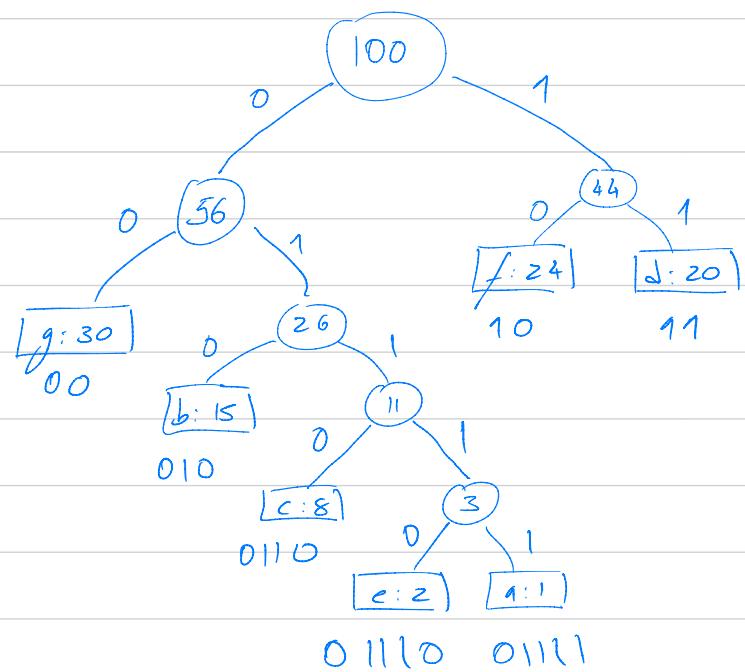
Quando atingimos o objecto i , colocamos em cada caixa apenas a fração de i suficiente para completar a capacidade da caixa.

$O(n)$

Total com ord.: $O(n \lg n)$

Q6 R2 14/15 II.b

[a:1] [b:15] [c:8] [d:20] [e:2] [f:24] [g:30]



Q7 RZ 15/16 - II.a

Input: $\underbrace{\langle d_1, \dots, d_n \rangle}_{\text{dólarinhos pn ordem crescente}}, K$ → valor do troco

Formalização: $\min \sum_{i=1}^n x_i$
 $\sum_{i=1}^n x_i \cdot d_i = K$

Importante:
$$\boxed{\forall 1 \leq i < n. \quad d_{i+1} \geq 2 \times d_i}$$

Q7 RZ 15/16 - II.a

Input: $\underbrace{\langle d_1, \dots, d_n \rangle}_{\text{domínios em ordem crescente}}, k$ → valor do troco

Formalização: $\min \sum_{i=1}^n x_i$
 $\sum_{i=1}^n x_i \cdot d_i = k$

Importante: $\boxed{\forall 1 \leq i < n. \quad d_{i+1} \geq 2 \times d_i}$

Escolha Greedy

$$x_n = \left\lfloor \frac{k}{d_n} \right\rfloor$$

Subproblema: $(\langle d_1, \dots, d_{n-1} \rangle, k - \left\lfloor \frac{k}{d_n} \right\rfloor)$

Q7 RZ 15/16 - II.a

Input: $\langle d_1, \dots, d_n \rangle$, K
denominações em ordem crescente

Valor da moeda

Formalização: $\min \sum_{i=1}^n x_i$
 $\sum_{i=1}^n x_i \cdot d_i = K$

$$\forall 1 \leq i < n. \quad d_{i+1} \geq 2 \times d_i$$

Escolha Greedy

$$x_n = \left\lfloor \frac{k}{d_n} \right\rfloor$$

Subproblema: $(\langle d_1, \dots, d_{n-1} \rangle, k - \left\lfloor \frac{k}{d_n} \right\rfloor)$

Provar que: x é óptimo $\rightarrow x_n = \left\lfloor \frac{k}{d_n} \right\rfloor$

• Provamos o resultado por contradição.

Suponhamos que x é óptimo e $x_n \neq \left\lfloor \frac{k}{d_n} \right\rfloor$.

Há dois casos a considerar:

$$\text{I) } x_n > \left\lfloor \frac{k}{d_n} \right\rfloor$$

$$x_n > \left\lfloor \frac{k}{d_n} \right\rfloor \Rightarrow x_n \cdot d_n > k$$

∴ x não é solução

$$\text{II) } x_n < \left\lfloor \frac{k}{d_n} \right\rfloor$$

$$\Leftrightarrow x_n \leq \frac{k}{d_n} - 1$$

$$\Leftrightarrow x_n \cdot d_n \leq k - d_n$$

• Isto significa que há pelo menos duas unidades que têm de ser pagas com denominações inferiores a d_n . Ora, usando a denominação da última peça da unidade com uma única moeda. Usando uma denominação inferior a d_n , temos de utilizar pelo menos duas moedas; de onde concluímos a contradição

∴

Q7 RZ 15/16 - II.a

Compute change (\vec{d} , k)

let \vec{x} be a new array of size $n = \vec{d}$. size

for $i=1$ to n

$$\vec{x}[i] = \lfloor k / \vec{d}[i] \rfloor$$

$$k = k - \vec{x}[i] \times \vec{d}[i]$$

return \vec{x}

Q7

RZ 15/16 - II.a

Input: $\underbrace{\langle d_1, \dots, d_n \rangle}_{\text{dólaras em ordem crescente}}, k$ Value do troco
 $\sum_{i=1}^n x_i \cdot d_i = k$

Formalização:

$$\min \sum_{i=1}^n x_i$$

$$\sum_{i=1}^n x_i \cdot d_i = k$$

$$\forall 1 \leq i < n. \quad d_{i+1} \geq 2 \times d_i$$

Escolha Greedy

$$x_n = \left\lfloor \frac{k}{d_n} \right\rfloor$$

Subproblema: $(\langle d_1, \dots, d_{n-1} \rangle, k - \left\lfloor \frac{k}{d_n} \right\rfloor)$

Contra-exemplo

$$\forall 1 \leq i < n. \quad d_{i+1} \neq 2 \times d_i$$

Q7

RZ 15/16 - II.a

Input: $\langle d_1, \dots, d_n \rangle, k$ Value do troco
 denominações em ordem crescente

Formalização: $\min \sum_{i=1}^n x_i$
 $\sum_{i=1}^n x_i \cdot d_i = k$

$\forall 1 \leq i < n. d_{i+1} \geq 2 \times d_i$

Escolha Greedy

$$x_n = \left\lfloor \frac{k}{d_n} \right\rfloor$$

Subproblema: $(\langle d_1, \dots, d_{n-1} \rangle, k - \left\lfloor \frac{k}{d_n} \right\rfloor)$

Contra-exemplo

$\forall 1 \leq i < n. d_{i+1} \neq 2 \times d_i$

$$\vec{d} = \langle 1, 7, 8 \rangle$$

$$\underline{k} = 14$$

$$\rightarrow 1 \times 8 + 6 \times 1 \Rightarrow 7 moedas$$

$$2 \times 7 \Rightarrow 2 moedas$$

Q8

R2 16/17 I.a

[a:21] [b:20] [c:19] [d:18] [e:22]

