

# **Aula 4**

## **Programação gulosa**

# Introdução

- Método guloso de projeto de algoritmos
  - escolhe, em cada iteração, o objeto mais "apetitoso" que vê pela frente
  - "míope": ele toma decisões com base nas informações disponíveis na iteração corrente
  - sem olhar as consequências que essas decisões terão no futuro

# Introdução

- Essa estratégia nem sempre produz o resultado esperado!

# Introdução

- Prog. Gulosa
  - abocanha a alternativa mais promissora (sem explorar as outras)
  - é muito rápido
  - nunca se arrepende de uma decisão já tomada
  - não tem prova de correção simples

# Introdução

- Prog. Dinâmica
  - explora todas as alternativa (mas faz isso de maneira eficiente)
  - é um tanto lento
  - a cada iteração pode se arrepender de decisões tomadas anteriormente (ou seja, pode rever o "ótimo corrente")
  - tem prova de correção simples

# Problema do Pendrive

- Tenho  $n$  arquivos no computador
- Cada arquivo  $i$  tem  $t_i$  megabytes
- Quero gravar o maior número possível de arquivos num pendrive que tem capacidade para  $c$  megabytes



# Problema do Pendrive

- encontrar o maior subconjunto  $X$  do intervalo  $1..n$  que satisfaça a restrição:

$$\sum_{i \in X} t_i \leq c$$

- Obs: queremos maximizar  $|X|$  e não a soma:

$$\sum_{i \in X} t_i$$

# Exemplo

- Instância:
  - $n = 8$ ,  $t = (10, 15, 20, 20, 30, 35, 40, 50)$  e  $c = 90$  tem muitas soluções:

10	15	20	20	30	35	40	50
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	-	-	-	-
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	-	<input checked="" type="checkbox"/>	-	-	-
<input checked="" type="checkbox"/>	-	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	-	<input checked="" type="checkbox"/>	-	-
-	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	-	<input checked="" type="checkbox"/>	-	-



# Problema do Pendrive

- Heurística:
  - Escolher o menor dos arquivos que ainda não foram gravados

O algoritmo é guloso pois abocanha o menor arquivo sem antes pesar os prós e contras dessa escolha

# Problema do Pendrive

- Embora a estratégia pareça natural, não é óbvio que ela produz o resultado desejado

```
PENDRIVE ( $t, n, c$ )   $\triangleright t_1 \leq \dots \leq t_n$   
1   $i := 1$   
2  enquanto  $i \leq n$  e  $t_i \leq c$   
3       $c := c - t_i$    $\triangleright$  escolhe  $i$   
4       $i := i + 1$   
5  devolva  $\{1, \dots, i-1\}$ 
```

# Exercício

- Proponha uma estratégia gulosa para resolver o problema da mochila booleana.
  - Qual o resultado da sua estratégia para uma mochila com  $n = 4$  objetos e capacidade  $c = 5$

	1	2	3	4
$p$	4	2	1	3
$v$	500	400	300	450

- Qual a classe de complexidade da sua solução?