

ADA

Algoritmos  
voraces  
(Greedy)

# Analisis y Diseño de Algoritmos

Juan Gutiérrez

October 19, 2022

# Intervalos disjuntos

ADA

Algoritmos  
voraces  
(Greedy)

**Problema Max-Intervalos-Disjuntos.** Dada una secuencia de intervalos cerrados en la recta, encontrar un subconjunto de intervalos compatibles dos a dos de tamaño máximo.

# Intervalos disjuntos

ADA

Algoritmos  
voraces  
(Greedy)

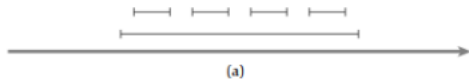


Figure 1: Tomada del libro Kleinberg, Algorithm Design

# Intervalos disjuntos

ADA

Algoritmos  
voraces  
(Greedy)

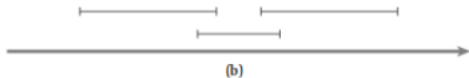


Figure 2: Tomada del libro Kleinberg, Algorithm Design

# Intervalos disjuntos

ADA

Algoritmos  
voraces  
(Greedy)

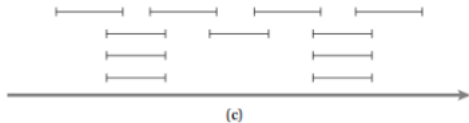


Figure 3: Tomada del libro Kleinberg, Algorithm Design

# Intervalos disjuntos

ADA

Algoritmos  
voraces  
(Greedy)

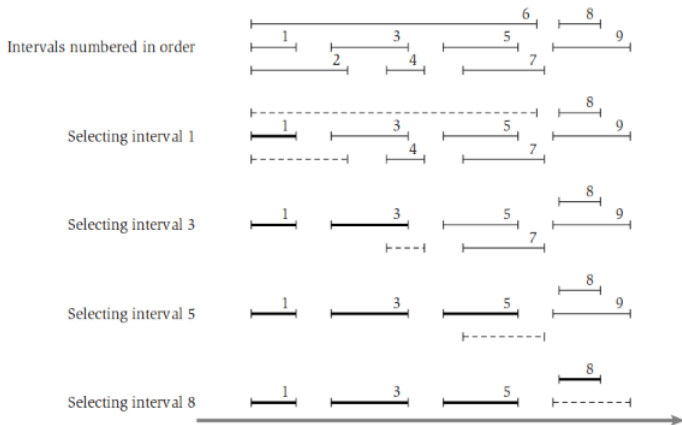


Figure 4: Tomada del libro Kleinberg, Algorithm Design

# Elementos de la estrategia Voraz

ADA

Algoritmos  
voraces  
(Greedy)

1. *Elección voraz*: debemos demostrar que siempre existe una solución óptima que contiene a la elección voraz
2. *Subestructura óptima*: debemos demostrar que la subsolución dejada es óptima para el subproblema dejado por la elección voraz

# Elementos de la estrategia Voraz

ADA

Algoritmos  
voraces  
(Greedy)

**Problema Max-Intervalos-Disjuntos.** Dada una secuencia de intervalos cerrados en la recta, encontrar un subconjunto de intervalos compatibles dos a dos.



# Elementos de la estrategia Voraz

ADA

Algoritmos  
voraces  
(Greedy)

*Recibe:* un conjunto  $\mathcal{I} = \{[s_1, f_1], [s_2, f_2], \dots, [s_n, f_n]\}$  de intervalos, ordenados de manera creciente por punta final

*Devuelve:* un subconjunto de intervalos compatibles dos a dos

MAX-INTERVALOS-DISJ-REC( $\mathcal{I}$ )

- 1: **if**  $\mathcal{I} = \emptyset$
- 2:   return  $\emptyset$
- 3:  $\mathcal{I}' = \mathcal{I} \setminus \{[s_i, f_i] : s_i \leq f_1\}$
- 4: **return**  $\{[s_1, f_1]\} \cup \text{MAX-INTERVALOS-DISJ-REC}(\mathcal{I}')$

# Elementos de la estrategia Voraz

ADA

Algoritmos  
voraces  
(Greedy)

**Lema 3.1** (Elección voraz). *Existe una solución óptima para el problema que contiene el intervalo  $[s_1, f_1]$ .*

# Elementos de la estrategia Voraz

ADA

Algoritmos  
voraces  
(Greedy)

**Lema 3.2** (Subestructura óptima). *Si  $X$  es una solución óptima al problema que contiene a  $[s_1, f_1]$  entonces  $X \setminus \{[s_1, f_1]\}$  es una solución óptima al subproblema dejado por la elección voraz.*

ADA

Algoritmos  
voraces  
(Greedy)

Gracias