



**DEPARTAMENTO  
DE COMPUTACION**

Facultad de Ciencias Exactas y Naturales - UBA

## TP2

---

### Algoritmos y Estructuras de Datos III

Integrante	LU	Correo electrónico
Ezequiel Aguerre	246/07	ezeaguerre@gmail.com
Juan Vanecek	169//10	juann.vanecek@gmail.com
Santiago Camacho	110/09	santicamacho90@gmail.com
Tomas Rodriguez	527/10	tomirodriguez.89@gmail.com



**Facultad de Ciencias Exactas y Naturales**  
**Universidad de Buenos Aires**

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

# Índice

<b>1. Problema 1: Robanúmeros</b>	<b>3</b>
1.1. Presentación del problema . . . . .	3
1.2. Resolución . . . . .	3
1.2.1. 1era solucion . . . . .	3
1.2.2. 2da solucion . . . . .	3
1.2.3. 3ra solucion . . . . .	3
1.3. Demostración . . . . .	4
1.4. Análisis de complejidad . . . . .	4
1.5. Test de complejidad . . . . .	4
1.6. Compilar y ejecutar . . . . .	4
<b>2. Problema 2: La centralita (de gas)</b>	<b>5</b>
2.1. Presentación del problema . . . . .	5
2.2. Resolución . . . . .	5
2.3. Demostración . . . . .	5
2.4. Análisis de complejidad . . . . .	5
2.5. Test de complejidad . . . . .	5
2.6. Compilar y ejecutar . . . . .	5
<b>3. Problema 3: Saltos en <i>La Matrix</i></b>	<b>6</b>
3.1. Presentación del problema . . . . .	6
3.2. Resolución . . . . .	6
3.3. Demostración . . . . .	6
3.4. Análisis de complejidad . . . . .	6
3.5. Test de complejidad . . . . .	6
3.6. Compilar y ejecutar . . . . .	6

# 1. Problema 1: Robanúmeros

## 1.1. Presentación del problema

## 1.2. Resolución

### 1.2.1. 1era solucion

En cada mano, agarrar la cantidad de cartas que sumen mas.

No funciona. Contraejemplo:  $[-1, -6, -10]$

### 1.2.2. 2da solucion

Sea  $v$  un vector de  $n$  enteros que son los valores de las cartas, y dada la funcion recursiva:

$$f(i, j, p) = \max \left( \underbrace{\max_{1 \leq k \leq j} \left\{ \left( \sum_{t=i}^k v_t \right) p + f(k, j, (p+1) \bmod 2) \right\}}_{\text{Lo mejor de la izquierda}}, \underbrace{\max_{1 \leq k \leq j} \left\{ \left( \sum_{t=k}^j v_t \right) p + f(i, k, (p+1) \bmod 2) \right\}}_{\text{Lo mejor de la derecha}} \right)$$

La solución al problema es  $f(1, n, 0)$  donde el 3er parámetro puede ser 0 o 1, y representa al jugador

### 1.2.3. 3ra solucion

La solucion es el valor  $\max(f_{\rightarrow}(1, n, yo), f_{\leftarrow}(1, n, yo))$  donde las funciones internas se definen como:

$$f_{\rightarrow}(i, j, p) = \begin{cases} \max_{1 \leq k \leq j} \left\{ \sum_{t=i}^k v[t] + \min(f_{\rightarrow}(k+1, j, p.next()), f_{\leftarrow}(k+1, j, p.next())) \right\} & \text{si } p = yo \\ \min_{1 \leq k \leq j} \{ \max(f_{\rightarrow}(k+1, j, p.next()), f_{\leftarrow}(k+1, j, p.next())) \} & \text{si } p = tu \end{cases}$$

$$f_{\leftarrow}(i, j, p) = \begin{cases} \max_{1 \leq k \leq j} \left\{ \sum_{t=k}^j v[t] + \min(f_{\rightarrow}(i, k-1, p.next()), f_{\leftarrow}(i, k-1, p.next())) \right\} & \text{si } p = yo \\ \min_{1 \leq k \leq j} \{ \max(f_{\rightarrow}(i, k-1, p.next()), f_{\leftarrow}(i, k-1, p.next())) \} & \text{si } p = tu \end{cases}$$

**1.3. Demostración**

**1.4. Análisis de complejidad**

**1.5. Test de complejidad**

**1.6. Compilar y ejecutar**

## 2. Problema 2: La centralita (de gas)

### 2.1. Presentación del problema

### 2.2. Resolución

### 2.3. Demostración

### 2.4. Análisis de complejidad

### 2.5. Test de complejidad

### 2.6. Compilar y ejecutar

### 3. Problema 3: Saltos en *La Matrix*

#### 3.1. Presentación del problema

#### 3.2. Resolución

#### 3.3. Demostración

#### 3.4. Análisis de complejidad

#### 3.5. Test de complejidad

#### 3.6. Compilar y ejecutar