

# Semana 9.- Algoritmos voraces II

## Tiempo mínimo en el sistema

- Tenemos  $n$  tareas que precisan de un tiempo de ejecución  $t_i$
- Todas están disponibles para ser ejecutadas por un único procesador en secuencia.
- Queremos **minimizar el tiempo medio de estancia de una tarea en el sistema**.
- $T_i$  es el tiempo en el sistema de la tarea  $i$ , que es la suma de su tiempo de ejecución  $t_i$  mas los de las tareas que se ejecutan antes.

### Ejemplo:

$$n = 3, t_1 = 5, t_2 = 10, t_3 = 3$$

- ▶ Si el orden de ejecución es 1, 2, 3:

$$T = 5 + (5 + 10) + (5 + 10 + 3) = 38$$

- ▶ Si el orden de ejecución es 1, 3, 2:

$$T = 5 + (5 + 3) + (5 + 3 + 10) = 31$$

- ▶ Si el orden de ejecución es 3, 1, 2:

$$T = 3 + (3 + 5) + (3 + 5 + 10) = 29$$

La planificación óptima consiste en atender las tareas por **orden creciente de ejecución**.

## Tareas con plazo y beneficio

- Tenemos  $n$  tareas, cada una requiriendo una unidad de tiempo en ejecutarse, con un plazo  $p_i$  y un beneficio  $b_i$ .

- Si la tarea se realiza antes de su plazo, se obtiene beneficio.
- No todas las tareas tienen por qué realizarse.
- Planificar las tareas para que se maximice el beneficio.

### Ejemplo:

$i$	$p_i$	$b_i$	Planificación	Beneficio total
1	2	30	[1, 3]	$30 + 25 = 55$
2	1	35	[2, 1]	$35 + 30 = 65$
3	2	25	[2, 3]	$35 + 25 = 60$
4	1	40	[3, 1]	$25 + 30 = 55$
			[4, 1]	$40 + 30 = 70$
			[4, 3]	$40 + 25 = 65$

Decimos que un conjunto de tareas es factible si existe alguna secuencia admisible, que permita realizar todas las tareas en su plazo.

Una permutación  $(i_1, i_2, \dots, i_k)$  es admisible si:

$$\forall j : 1 \leq j \leq k : p_{i_j} \geq j$$

La estrategia voraz pasa por considerar las tareas de mayor a menor beneficio y si cada tarea sigue siendo factible al ser añadida.

### Ejemplo:

$i$	$p_i$	$b_i$
1	4	40
2	1	35
3	1	30
4	3	25
5	1	20
6	3	15
7	2	10

$S = \emptyset$

$S = \{1\}$                        $[1]$                       40

$S = \{1, 2\}$                        $[2, 1]$                       75

$S = \{1, 2, 3\}$                       3 rechazada

$S = \{1, 2, 4\}$                        $[2, 1, 4]$                       100

$S = \{1, 2, 4, 5\}$                       5 rechazada

$S = \{1, 2, 4, 6\}$                        $[2, 4, 6, 1]$                       115

$S = \{1, 2, 4, 6, 7\}$                       7 rechazada