



1.- (7 puntos) Se dispone de una tabla T con n filas y m columnas que representan las posibilidades de que ciertos trabajadores (n) realicen determinadas tareas (m). Si $T[i, j]=1$ entonces el trabajador i-ésimo puede realizar la tarea j-ésima. En caso contrario no puede realizarla. Cada tarea puede ser realizada por uno o ningún trabajador, y cada trabajador debe tener una tarea o ninguna. El objetivo es obtener una asignación de trabajadores con tareas, de forma que el número de tareas realizadas sea máximo. Utilizando la metodología de Backtracking, diseñar un algoritmo que resuelva este problema.

Deberá responderse a las siguientes cuestiones:

- Identificar y escribir el esquema de Backtracking a utilizar
- Identificar cada una de las operaciones del esquema en relación con los datos del problema
- Dar el algoritmo resultante

Ejemplo

	Tarea 1	Tarea 2	Tarea 3	Tarea 4	Tarea 5
Trabajador 1	1	0	0	0	0
Trabajador 2	0	1	1	1	0
Trabajador 3	0	1	0	0	1
Trabajador 4	1	0	1	1	0

Una solución óptima sería:

Trabajador 1 → Tarea 1 / Trabajador 2 → Tarea 2 / Trabajador 3 → Tarea 5 / Trabajador 4 → Tarea 3



2.- (3 puntos) Se plantea una situación similar a la del problema 1, pero en este caso la tabla T también con n filas y m columnas representa los beneficios (valores enteros) de que ciertos trabajadores (n) realicen determinadas tareas (m). Si $T[i, j]=s$ entonces el beneficio de que el trabajador i-ésimo realice la tarea j-ésima es s. Cada trabajador puede realizar varias tareas o ninguna pero las tareas deben ejecutarse todas y cada una sólo puede ser realizada por un trabajador. El objetivo es asignar trabajadores a las tareas, de forma que el beneficio global sea máximo. Utilizando la metodología Voraz, diseñar un algoritmo que resuelva este problema.

Deberá responderse a las siguientes cuestiones:

- Identificar y escribir el esquema voraz a utilizar
- Identificar cada una de las operaciones del esquema en relación con los datos del problema
- Dar el algoritmo resultante

Ejemplo

	Tarea 1	Tarea 2	Tarea 3	Tarea 4	Tarea 5
Trabajador 1	1	2	6	6	5
Trabajador 2	3	5	1	4	0
Trabajador 3	4	1	8	5	1
Trabajador 4	2	7	7	2	3

Una posible solución a este problema sería:

Trabajador 3 → Tarea 1 / Trabajador 4 → Tarea 2 / Trabajador 3 → Tarea 3 / Trabajador 1 → Tarea 4
Trabajador 1 → Tarea 5