P2: Asignación de examenes a aulas

La Escuela Técnica Superior de Ingeniería Informática y de Telecomunicaciones (ETSIIT) está planificando los exámenes finales para las distintas asignaturas. Se deben programar N exámenes en un día determinado, y para cada examen, el profesorado ha especificado el tiempo requerido para su realización y la hora de inicio ideal.

El centro cuenta con m aulas disponibles, donde m es mayor que N, lo que garantiza la viabilidad de programar los exámenes. Sin embargo, cada vez que se utiliza un aula, el centro debe asegurar su vigilancia, lo que implica contratar a un vigilante distinto para cada aula utilizada.

Se solicita diseñar un algoritmo que, de manera eficiente en términos de tiempo, permita garantizar que los exámenes se realicen con el menor costo posible para la escuela.

n exámenes

Cada examen → inicio, duración (struct).

m aulas

m>n

↳ Vamos por casos

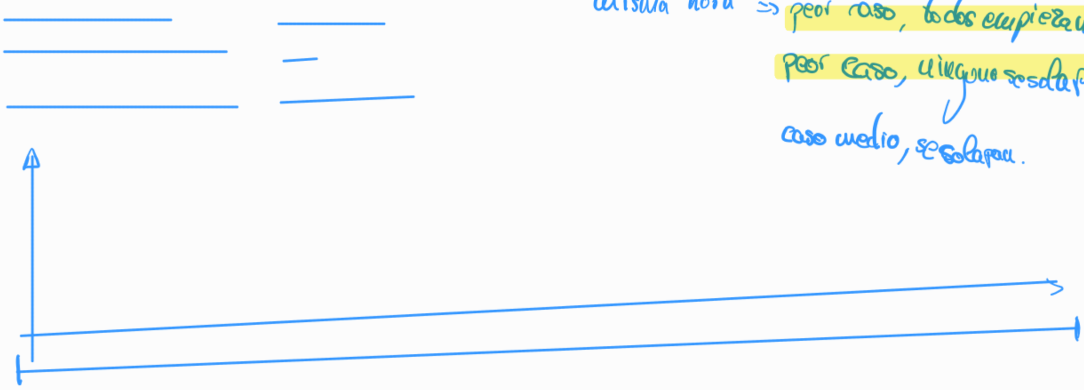
Busco: 1 → pocas aulas

2 → solapar lo menor posible.

↳ Voy, por huevos, a solapar los que tengan la misma hora ⇒ peor caso, todos empiezan a la misma hora

peor caso, ninguno se solapa ↳ asignación trivial

caso medio, se solapan.



Ta claro

Posibles criterios:

·) Por hora de inicio (siempre hay que tenerlo en cuenta)

·) ¿Y por hora de finalización? ↳ Bingo ↳ ¿Cómo controlo más aulas

↳ Coger secuencias no solapables.

Es decir, ir de forma secuencial por aulas

Primera Solución
- Hora de fin
- Sacar array con horas de fin
- No solapar más de la hora de fin.

Conjunto de Candidatos → Exámenes

Conjunto de Seleccionados → Aquellos exámenes que tienen la misma hora de fin.

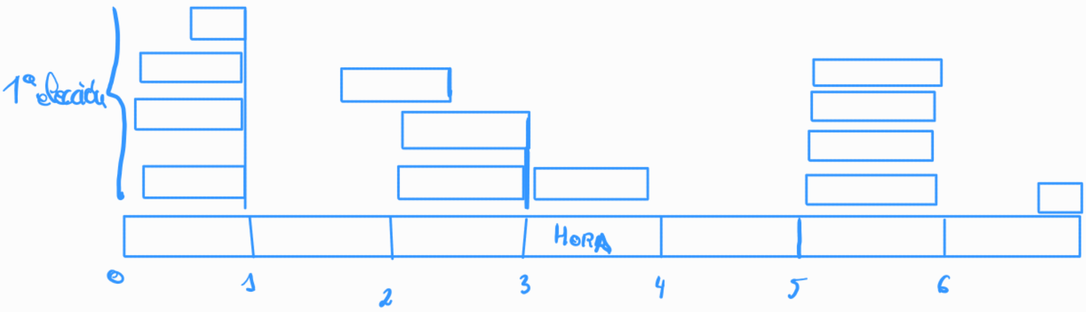
Función solución → Determina que hay solución si se han asignado todos los exámenes

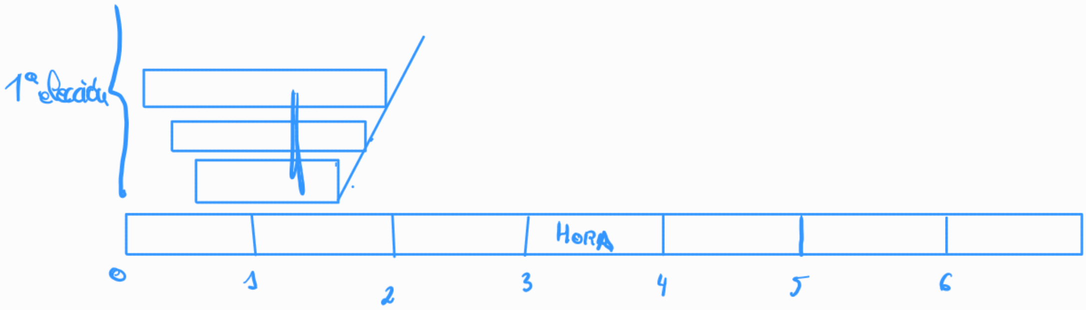
Función factibilidad → (1ª versión) No hay más exámenes solapados que aulas.

Función selección → Selecciona conjuntos de exámenes con la misma hora de fin (+ε?) no vale

Función Objetivo → Dar un list de list de pot. aparts ¿Seguro?

Función Optimalidad → Tiempo perdido es mínimo





¿Y si voy por aulas? → Probamos por inducción sobre N?

1ª elección, el examen con menor hora de fin y no solapamiento. Supongamos que E_{11} no forma parte de la solución optimal S ⇒ ∃ E_{11}' que es la primera elección de la solución ⇒ (igual que el anterior!) como E_{11}.hora_fin = min { E_{ij}.hora_fin, i ∈ Δ_N, j ∈ Δ_M } ⇒ la solución ¿qué criterio me lleva a que es la óptima?

1	√ 0	3	3	3	Aula 1
	√ 0	5	5	4	Aula 2
	√ 2	3	5	5	Aula 3
	√ 7	2	9	6	Aula 4
1	√ 1	2	3	2	Aula 5
	3	7	10	7	
	√ 1	1	2	1	

1	2	3	4
1 2	1 3	0 3	0 4
2 5	3 7		
7 9			

Elección: 1) (0,3) (0,5) (1,2), (1,1), (2,3)
2) (7,2)

1 2	1 3	0 3	0 5
2 5	3 10		
7 9			

Posible idea final

Selección → menor hora final
- → si solapa aula nueva
- → si no solapa aula vieja

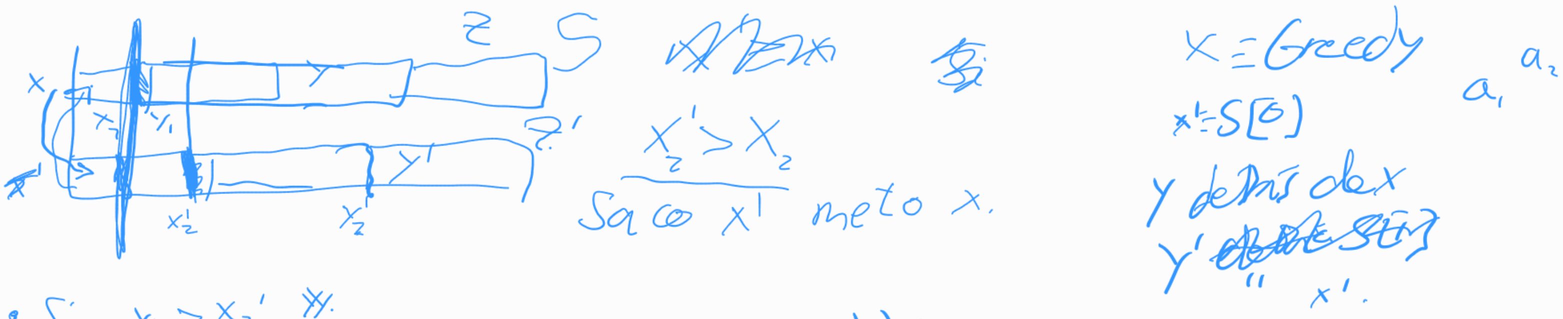
Optimal → menor nº aulas → solo las que solapen obligatoriamente.

solución → No hay exámenes sin asignar

Veamos que si tengo un conjunto de examenes $\mathcal{E} \Rightarrow \mathcal{E}_0 \in \mathcal{A}$ siendo $\mathcal{A}$ una solución optimal.

•) $\mathcal{E}_0 \rightarrow$ hora de fin más temprana

Supongamos $\nexists$ T solución óptima, $T \subseteq \mathcal{E} \mid \mathcal{E}_0 \notin T$. Sea $\mathcal{B}$ sol. óptima y $\mathcal{B}_0$ 1er examen (empieza) $\Rightarrow$ como greedy elige menor hora fin $\Rightarrow \mathcal{B}_0.\text{fin} > \mathcal{E}_0.\text{fin} \Rightarrow$



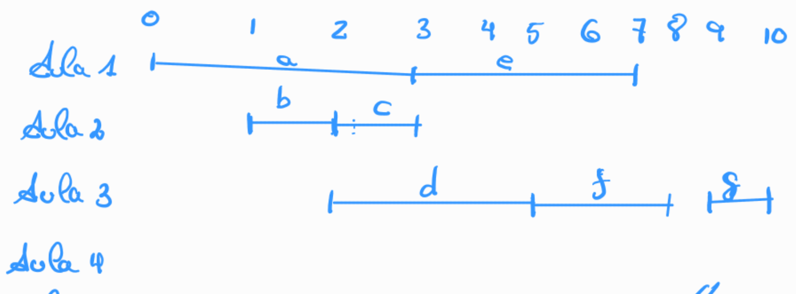
• Si $y_1 > x_2'$

• Si $y_1 < x_2'$

Inicio	Fin
5	3
2	1
1	2
3	7
9	10
2	5
0	3

orden →

Inicio	Fin	ident.
0	3	a
1	2	b
2	3	c
2	5	d
3	7	e
5	8	f
9	10	g



Terminada la implementación

Demostración:

$\mathcal{A}$ → conjunto de aulas

$\mathcal{E}$ → conjunto de exámenes

$\mathcal{E}_i$ → Examen i-ésimo (tras ordenación)

$\mathcal{A}_i$ → Aula i-ésima

$h_I(\mathcal{E}_i)$ hora inicio examen i-ésimo

$h_F(\mathcal{E}_i)$ hora fin examen i-ésimo

Primera elección

Supongamos que $\mathcal{E}_g$ es el examen que elije el algoritmo greedy como primer examen. Es claro que este examen cumple que $h_I(\mathcal{E}_g) = \min\{h_I(\mathcal{E}_i) \mid \mathcal{E}_i \in \mathcal{E}\}$. Supongamos ahora que no existe una solución S tal que el primer examen elegido sea $\mathcal{E}_g$.

Como todos los exámenes son elegidos es fácil ver que $\mathcal{E}_g$ debe ser el primero de la clase $\mathcal{A}_j$ con $j > 1$.

Sabemos que $h_I(\mathcal{E}_g) < h_I(\mathcal{E}_o)$, luego podemos intercambiarlo por $\mathcal{E}_o$; no obstante, hay dos casos

·) $h_F(\mathcal{E}_g) = h_F(\mathcal{E}_o)$, con el intercambio no obtenemos beneficio y el número de aulas es el mismo

·) $h_F(\mathcal{E}_g) \neq h_F(\mathcal{E}_o)$, con el intercambio, veremos que se produce solapamiento con $\mathcal{E}_g'$ → siguiente a $\mathcal{E}_g$ o $\mathcal{E}_o'$ → siguiente a $\mathcal{E}_o$. Sin embargo, realizando este intercambio entre $\mathcal{E}_g^{(n)}$ y $\mathcal{E}_o^{(n)}$ siguiendo los casos de nuevo hasta un momento en el que ya no haya solapamiento entre $\mathcal{E}_g^{(n+1)}$ y $\mathcal{E}_o^{(n+1)}$ obteniendo, en el peor de los casos la misma solución y el mismo tiempo desperdiciado $(\sum_{t \in T} t \mid T = \{h_F(\mathcal{E}_i^{(n)}) - h_I(\mathcal{E}_i^{(n+1)})\}$ en cada una de las aulas). Es decir, si no se produce el intercambio hasta el final, se obtendrá un tiempo desperdiciado menor haciendo que la solución sea mejor

En conclusión haciendo el intercambio obtenemos una solución a lo sumo igual a la anterior pero mejor. Contradicción con que S es la solución óptima.