

Vuelta atrás + voraces - Enero 2018

Secuencia de decisiones: $\langle x_1, x_2, \dots, x_E \rangle$ que indica el aula donde se realizará el examen i -ésimo.

Función objetivo: minimizar el número de aulas utilizadas. Debemos contabilizar las distintas aulas que se van a utilizar, teniendo en cuenta que en un aula pueden hacerse varios exámenes, lo que indica valores repetidos en la secuencia de decisiones.

Restricciones implícitas y explícitas:

$$(\forall i)(x_i \in \{1, 2, \dots, A\} : 1 \leq i \leq E)$$

La suma de los alumnos de los diferentes exámenes destinados a un mismo aula no puede exceder la capacidad de dicho aula.

Tipo de solución: óptima

Preparar - recorrido - nivel - k : $x[k] = 0$

Existe - hermano - nivel - k : $x[k] < A$

Siguiente - hermano - nivel - k : $x[k] += 1$

Solución: $k = E$

Función correcto: necesita la secuencia de decisiones x , la posición hasta la que se ha completado la secuencia de decisiones k , y los vectores Capacidad y NumAlumnos.

La función devuelve cierto si la suma de todos los alumnos que realizan su examen en su aula ($x[k]$) no supera la capacidad del aula; falso en caso contrario.

Funcion Correcto (x : tupla; k : entero; Capacidad $[1..A]$, NumAlumnos $[1..E]$: vector de enteros) retorna (b : booleano)

var i : entero {var

Total-alumnos = NumAlumnos $[k]$;

$i = 0$

mientras ($i < k-1$) hacer

$i = i + 1$;

si ($x[i] = x[k]$) entonces

Total-alumnos += NumAlumnos $[i]$;

{si

}mientras

si Total-Alumnos \leq Capacidad $[x[k]]$ entonces retorna cierto

si no retorna falso {si.

ffuncion

Función valor: necesita la secuencia de decisiones completa, o sea, $k = E$. La función valor produce el número total de aulas que han sido precisadas para la organización de los exámenes. Para ello se usa un vector auxiliar (Aula-usada) que almacenará un 0 en el caso de que el aula no haya sido usada (i) y 1 en caso contrario.

Funcion valor (x : tupla; k : entero) retorna (t : entero)

var i , Total-aulas: entero; Aulas-usadas $[1..A]$: vector de enteros; {var

Total-aulas = 0

para $i = 1$ hasta A hacer

Aula-usada $[i] = 0$;

}para

para $i = 1$ hasta k hacer

Aula-usada $[x[i]] = 1$;

}para

para $i = 1$ hasta A hacer

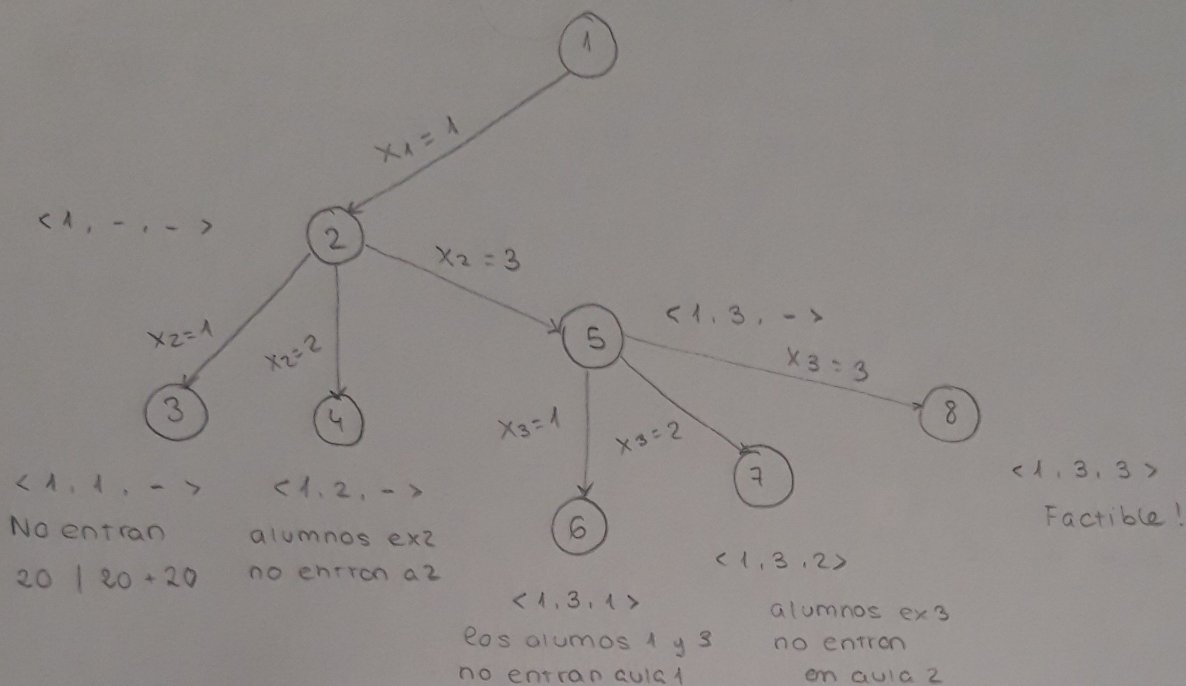
Total-aulas += Aula-usada $[i]$;

}para

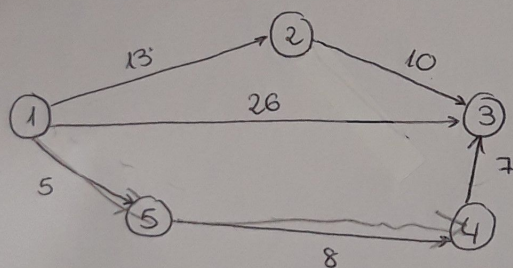
return Total-aulas

ffuncion

Capacidad [1..4] = {20, 10, 40, 10}
 NumAlumnos [1..3] = {20, 20, 20}



Algoritmo de Dijkstra



Etapa	Nodo selecc.	Nodos ya selecc.	Nodos aún no escog	Vector distancias (D)	Vector predec.
Inicial	-			$D[2..5]$	$P[2..5]$
1	5	{1, 5}	{2, 3, 4}	{13, 26, ∞ , 5}	{1, 1, -, 1}
2	2	{1, 5, 2}	{3, 4}	{13, 26, 13, 5}	{1, 1, 5, 1}
3	4	{1, 5, 2, 4}	{3}	{13, 23, 13, 5}	{1, 2, 5, 1}
				{15, <u>20</u> , 13, 5}	{1, 4, 5, 1}

La longitud del camino mínimo desde el nodo 1 hasta el nodo 3 está recogida en la posición 3 del vector, 20. Para saber por donde discurre dicho camino, hay que acudir a la posición 3 del vector p, que nos indica cual es el nodo predecesor.

4 \rightarrow 3 y vamos a p[4]
 5 \rightarrow 4 \rightarrow 3 p[5]
 1 \rightarrow 5 \rightarrow 4 \rightarrow 3