

Las tareas de la clase

Ha comenzado el curso y la profesora tiene que asignar los diversos cargos y tareas de la clase: quien será el delegado, quien anotará los deberes en la agenda, quien borrará la pizarra entre clases o quien se encarga de recoger y repartir los trabajos. Para ello ha solicitado a los alumnos que le indiquen sus preferencias. Darán una puntuación entre 1 y 10 a cada tarea. 10 indica que les gustaría mucho realizarla y 1 que no quieren hacerla. Si hay varias tareas que les gustan parecido, pueden darles la misma puntuación.

Ahora ha llegado el momento de hacer la asignación. La profesora va a **elegir 2 alumnos para cada puesto**, ya que a veces alguno de ellos falta y no quiere que la tarea quede sin responsable. Para intentar que participen el mayor número posible de alumnos y evitar que unos pocos acaparen todos los puestos, **ningún alumno ocupará más de t puestos**. Como quiere que los alumnos estén lo más contentos posible, intentará asignar los puestos según las preferencias que han formulado. **Busca la asignación tal que la suma de las preferencias expresadas por los alumnos seleccionados para los puestos sea máxima.**

Requisitos de implementación.

El problema se debe resolver utilizando la técnica de vuelta atrás. **Se debe podar el árbol de ejecución impidiendo realizar llamadas recursivas que no darán soluciones en base a la solución ya construida, pero no es necesario realizar estimaciones sobre las tareas que todavía no han sido tratadas.**

Explicar el algoritmo: como es la solución que se va construyendo y como la vamos a ir construyendo. Poner comentarios en el código.

Entrada

La entrada consta de una serie de casos de prueba. Cada caso de prueba consta de varias líneas. En la primera se muestra el número de tareas n que hay que realizar, el número de alumnos de la clase a y el número máximo de tareas que puede realizar un alumno t . En las a líneas siguientes se muestran n valores que representan las preferencias del alumno para cada tarea.

El número de tareas es un entero, $1 \leq n \leq 6$, el número de alumnos es un entero, $2 \leq a \leq 8$ y el número máximo de tareas que puede realizar un alumno $1 \leq t \leq 6$. Se garantiza que siempre existe una asignación posible, es decir $a * t > 2 * n$

Salida

Para cada caso de prueba se muestra en una línea la suma de las preferencias de los alumnos que realizarán las tareas.

Entrada de ejemplo

[a*n]	3 5 2	n: numTareasas a: numAlumnos t: numMaxTareasPorAlumno
	5 10 2	
	1 8 4	
	3 10 3	
	4 7 7	
	1 7 6	
	4 2 4	
	5 5 5 5	
	7 7 7 7	
	3 3 2	
	10 10 8	
	8 9 1	
	1 7 8	
	0 0 0	

Salida de ejemplo

```
42
48
50
```

Una asignación óptima

```
42
0 3 0 2 3 4
48
0 1 0 1 0 1 0 1
50
0 1 1 2 0 2
```