



# CENTRAL-EUROPEAN OLYMPIAD IN INFORMATICS

Sárospatak, Hungary  
28 July - 4 August 2005

Page 1 of 2

Español

Day 2: ticket

## Ticket Office (oficina de tickets)

**Input file:** ticket.in

**Output file:** ticket.out

**Source code:** ticket.pas/.c/.cpp

**100 points**

**Time limit: 0.5 sec**

**Memory limit: 64 MB**

Una oficina de venta de tickets vende tickets para conciertos. En lugar de vender tickets para asientos individuales vende manojos de tickets para un determinado número fijo de asientos consecutivos. La oficina ha recibido una gran cantidad de demandas. La demanda de un manajo de tickets se determina mediante el número más pequeño del asiento.

Sucede que la oficina no puede satisfacer todas las demandas. Además, si solo reserva los asientos exactamente como le piden las demandas, entonces un gran número de asientos pueden quedar vacíos. Por eso la oficina decide aplicar la siguiente estrategia de colocación y de precios. Si una demanda es aceptada y los asientos asignados son exactamente los que pide la demanda entonces se paga el precio completo (2 petaks por el manajo, un petak es el nombre de una moneda ficticia). Si una demanda es aceptada, pero los asientos asignados difieren de los demandados (basta que haya uno solo que difiera!), entonces se paga la mitad del precio (1 petak por el manajo).

El objetivo es maximizar las ganancias.

### Task

Escribe un programa que calcule el máximo de ganancia obtenible (Subtask A) y coloque los asientos de las demandas seleccionadas (Subtask B).

### Input

La primera línea del fichero de texto ticket.in contiene dos enteros,  $M$  y  $L$ .  $M$  ( $1 \leq M \leq 30000$ ) es el número de asientos y  $L$  ( $1 \leq L \leq 100$ ) es el número de asientos consecutivos en cada manajo. Los asientos están numerados de 1 a  $M$ . La segunda línea contiene un entero,  $N$  ( $1 \leq N \leq 100\,000$ ), el número de demandas. La tercera línea contiene  $N$  enteros, que definen las demandas. El  $i$ -ésimo número en la línea,  $z$  ( $1 \leq z \leq M-L+1$ ), significa que la  $i$ -ésima demanda de un manajo de asientos empieza en el asiento  $z$  y termina en el asiento  $z+L-1$ .

### Output

La primera línea del fichero de texto ticket.out contiene un entero,  $S$ , el máximo de ganancias (Subtask A). La segunda línea contiene un entero,  $Q$ , el número de demandas aceptadas. Las siguientes  $Q$  líneas describen la colocación de los asientos (Subtask B). Cada línea contiene un par de enteros,  $x$  y  $y$ . El par  $x$  y  $y$  significa que la demanda  $x$  obtiene los asientos que empiezan en el número  $y$ . Las líneas deben escribirse en orden creciente, respecto los números de los asientos. Si hay múltiples posibilidades, tu programa solo debe mostrar a una de ellas.

### Example

ticket.in	ticket.out
20 3	9
7	6
4 2 10 9 16 15 17	2 1
	1 4
	4 7
	3 10
	6 13
	5 16



# CENTRAL-EUROPEAN OLYMPIAD IN INFORMATICS

Sárospatak, Hungary  
28 July - 4 August 2005

Page 2 of 2

Español

Day 2: ticket

---