
Multiselección**X56972_es**

En el problema de la *multiselección* se nos da un (multi)conjunto X de $n > 0$ elementos y $p \geq 1$ rangos

$$1 \leq i_1 < i_2 < \dots < i_p \leq n$$

El objetivo es desarrollar un algoritmo que devuelva en orden creciente el i_1 -ésimo menor elemento de X , el i_2 -ésimo menor elemento de X , etc.

Un algoritmo eficiente de multiselección no ordena la entrada; tampoco aplica un algoritmo de selección a cada uno de los p rangos dados. Aunque la demostración es compleja, es relativamente fácil diseñar un algoritmo de multiselección cuyo coste promedio sea $\Theta(n \log p)$.

Entrada

La entrada consiste en el valor $n > 0$, seguido del valor $p \geq 1$, a continuación los p rangos i_1, \dots, i_p en orden creciente, y finalmente n enteros x_1, \dots, x_n que constituyen el (multi)conjunto X .

Salida

La salida son los p elementos de X designados por los rangos dados: el i_1 -ésimo menor, el i_2 -ésimo menor, y así sucesivamente hasta el i_p -ésimo menor, separados por espacios en blanco.

Observación

Vuestro programa deberá estar escrito en C++. Para la evaluación de la práctica se usarán juegos de pruebas públicos y privados, pero el *Jutge* sólo tendrá en cuenta los juegos de pruebas públicos para determinar si el programa pasa o no, y los resultados de los juegos de pruebas privados sólo se usarán para perfilar la nota de la práctica.

En el fichero entregado, el algoritmo deberá estar debidamente comentado. En particular debéis justificar su corrección.

El algoritmo **no** debe ordenar la entrada ni resolver la tarea aplicando un algoritmo de selección p veces, una vez con cada rango. En caso contrario la nota de la práctica será 0, independientemente de su funcionamiento correcto.

Se valorará que se desarrolle un algoritmo genérico (con `templates`) ya que el tipo de los elementos del conjunto X es irrelevante, lo único importante es que exista una relación de orden total sobre dichos elementos.

La fecha y hora límite de entrega se publicará en un aviso del Racó.

Ejemplo de entrada 1

20 3 1 2 20 3 4 5 7 8 1 2 6 9 10 1 4 7 9

Ejemplo de entrada 2

10 1 4 9 7 3 2 8 6 2 1 5 2

Ejemplo de salida 1

18 11 18 12 15 17

Ejemplo de salida 2

2

Ejemplo de entrada 3

11 11 1 2 3 4 5 6 7 8 9 10 11 4 9 7 3 2 8

Ejemplo de salida 3

15 2 2 5 2 4 5 6 7 8 9

Información del problema

Autor : Conrado Martinez

Generación : 2013-03-06 00:18:24

© *Jutge.org*, 2006–2013.

<http://www.jutge.org>