

## 0 OBJETIVOS

- Diseñar soluciones computacionales para problemas.
- Estimar costos de las soluciones planteadas.
- Implementar soluciones en *Java*.

Se premiarán las mejores soluciones y se castigarán las peores, en cuanto a eficiencia en tiempo y espacio.

## 1 CONDICIONES GENERALES

Hay tres problemas para resolver mediante soluciones implementadas en *Java*.

Para cada problema se pide:

- Análisis temporal y espacial.
- Una solución java.

## 2 PROBLEMAS

### A Subarreglo más largo con máximo de ceros

Sea  $a[0..n-1] : \text{int}$  un arreglo de números enteros y  $c : \text{nat}$ .

#### *Problema*

Encontrar la longitud del subarreglo más largo de  $a$  que tiene, a lo sumo,  $c$  ceros.

#### *Ejemplo:*

Supónganse  $n=15$ ,  $a = [12, 0, -3, 0, 0, 1, 2, 3, 4, 0, 4, 5, 23, 0, 100]$ .

Si  $c=3$ , la respuesta es 11. Si  $c=1$ , la respuesta es 8.

### B K-sima siguiente permutación

Sean  $r$  un número natural con representación decimal  $d_{n-1}d_{n-2}...d_0$ . Es decir, cada  $d_i \in 0..9$  y, además,  $r = (+i \mid 0 \leq i < n : 10^i * d_i)$ . Si existe, se define  $sp(r, 1)$ , la 1-siguiente permutación de  $r$ , como el menor número natural que tiene los mismos dígitos de  $r$  y que sea mayor que  $r$ .

Más generalmente, para  $k > 1$ , se define  $sp(r, k)$ , la  $k$ -siguiente permutación de  $r$ , como el menor número natural que tiene los mismos dígitos de  $r$  y que sea mayor que  $sp(r, k-1)$ .

## Problema

Dados  $r \geq 0$ ,  $k > 0$ , use debe producir  $sp(r, k)$ , si existe, o determinar que no existe.

*Ejemplo:*

Si  $n = 123542$ , entonces  
 $sp(123542, 1) = 124235$   
 $sp(123542, 2) = 124325$   
 $sp(123542, 3) = 124352$

Nótese que  $sp(543221, k)$  no existe, para  $k \geq 1$ .

## C Ríos más largos

En composición tipográfica, un "río" es una cadena de espacios formada por separaciones entre las palabras que se extiende por varias líneas de texto. Por ejemplo, la figura muestra varios ejemplos de ríos resaltados en rojo (en el texto todas las letras han sido remplazadas por el carácter 'x' para hacer que los ríos sean más visibles).

```

xxxx*xxx xxxxx xxxxxxxxxxxxxxxx*xxxxxxxxxxxxxxxxxxxxxx*x xxxxxxxx
xxx*xxxxxxxx*xx xxxxxxxx x*xxxx*xxxxxxxxxxx xxxx xx*xx xxxxxxxx
xx*xxxxxxxx*xxxxxxxx xxxxxx*xxxxxxxxxxx xxxxxxxx xx*xxxx xxxx
xxxxx xxxxx*xxxxxxxxxxx xxxxxx*x xxxxx xxxxxxxxxxxxxxxxxx xxxxxxxx
xxxxxxxx xx*xxxxxxxx xxxxxx*x xxxxxx xxxxxx xxxxxxxxxxxxxx xxxx
xx xxxx xxxxxx xxxxxx xxx xxxxxxxxxxxxxxxxxxxxxxx xxx xxxxxxxxxxxxxx

```

El texto que se considera se entiende escrito en una fuente monoespaciada (todas las letras y espacios tienen igual ancho) y en una columna alineada a la izquierda de cierto tamaño fijo, con exactamente un espacio separando palabras en cada línea (el texto no está alineado a la derecha). Un río se define como una secuencia de espacios en líneas consecutivas en las que la posición de cada espacio en la secuencia (excepto la primera) difiere a lo sumo en 1, desde la posición del espacio en la línea de arriba. Como excepción, ningún espacio en blanco al final de una línea puede aparecer en un río. Las palabras deben estar lo más ajustadas posible en líneas; ninguna palabra puede dividirse entre líneas. El tamaño de línea usado debe ser al menos tan largo como la palabra más larga en el texto.

## Problema

Dado un texto, se quiere determinar el ancho de línea que produce el río más largo de espacios para ese texto y, en este caso, el largo de dicho río. Si el río más largo se produce con más de un ancho de línea, se pregunta por el ancho más pequeño que lo consigue.

*Ejemplo:*

La siguiente tabla muestra el mismo texto configurado con dos anchos de línea diferentes.

Ancho de línea: 17	Ancho de línea: 25
El Amazonas es*el	El Amazonas es el río más
río más largo*de	largo de*Suramérica y el
Suramérica y*el	Magdalena*es el más largo
Magdalena es*el	que fluye*enteramente en
más largo que	Colombia
fluye enteramente	
en Colombia	
Río más largo: 4	Río más largo: 3

### 3 ENTRADA / SALIDA DE DATOS

En todas las soluciones que se presenten, la lectura de los datos de entrada se hace por la entrada estándar; así mismo, la escritura de los resultados se hace por la salida estándar.

Puede suponer que ninguna línea de entrada tiene espacios al principio o al final, y que los datos que se listan en cada línea están separados por exactamente un espacio.

El fin de la entrada de cada problema se identifica porque no hay más casos para leer. Esto se puede representar por una línea vacía en la entrada estándar o por el carácter de fin de archivo si la entrada está redirigida desde un archivo.

A continuación, para cada problema, se establecen parámetros que definen su tamaño y formato de lectura de los datos, tanto de entrada como de salida.

#### 3.1 Problema A: Subarreglo más largo con máximo de ceros

- Tamaño del problema:  $n, c$  (la dimensión del arreglo, el número de ceros aceptable)
- Condiciones de los casos de prueba:  $0 \leq n < 10^5, 0 < c < 100$ .

##### **Descripción de la entrada**

La entrada contiene varios casos de prueba. Cada caso de prueba se define con una línea que contiene  $n+2$  números enteros:

$n \ c \ z_0 \ z_1 \ \dots \ z_{n-1}$

que representan

$n$  : la dimensión de la lista de enteros

$c$  : el máximo número de ceros en un subarreglo

$z_i$  : los valores enteros que constituyen el arreglo,  $0 \leq i < n$ .

##### **Descripción de la salida**

Por cada caso para resolver, imprimir una línea de respuesta con la respuesta (un número entero).

##### **Ejemplos de entrada / salida**

Entrada	Salida
15 3 12 0 -3 0 0 1 2 3 4 0 4 5 23 0 100	11
15 1 12 0 -3 0 0 1 2 3 4 0 4 5 23 0 100	8

#### 3.2 Problema B: k-sima siguiente permutación

- Tamaño del problema:  $r, k$ .
- Condiciones de los casos de prueba:  $0 < r < 10^{1000}, 0 < k < 10^3$ .

##### **Descripción de la entrada**

La entrada contiene varios casos de prueba. Cada caso de prueba se define con una línea de la forma

$k \ d_{n-1} d_{n-2} \dots d_0$

donde  $k$  es un entero positivo y  $n$  es el número de dígitos de  $r$ ,  $r = (+i | \ 0 \leq i < n : 10^i * d_i)$ .

### Descripción de la salida

Por cada caso para resolver, imprimir una línea de respuesta.

La línea de respuesta debe darse con una línea que contenga una cadena de dígitos

$d'_{n-1}d'_{n-2}\dots d'_0$

que exprese  $sp(r, k)$  en notación decimal, cuando éste esté bien definido. En otro caso, la línea de respuesta debe contener un carácter asterisco alineado a la izquierda, i.e.,

\*

### Ejemplo de entrada / salida

Entrada	Salida
1 123542	124235
1 543221	*
4 345249689013	345249689301

### 3.3 Problema C: Ríos más largos

- Tamaño del problema:  $n$  (longitud del texto, contada desde el primero hasta el último carácter no blanco).
- Condiciones de los casos de prueba:  $0 < n \leq 2 \times 10^5$ .

### Descripción de la entrada

La entrada contiene varios casos de prueba. Cada caso de prueba comienza contiene una línea con el texto que se debe analizar.

Cada palabra consiste solo en letras minúsculas y mayúsculas (eventualmente, acentuadas) y las palabras en la misma línea están separadas por un solo espacio. Ninguna palabra tiene más de 80 caracteres.

El final de un caso de prueba se indica con un carácter de fin de línea (*carriage return*) que, por supuesto, no hace parte de la entrada correspondiente. Para los ejemplos dados se utiliza <CR> para señalar el lugar en donde va el carácter que representa el fin de línea.

### Descripción de la salida

Para cada caso de prueba se debe producir una línea de la forma

$a \ r$

donde  $a$  es el ancho de línea más corto que produce un río de tamaño  $r$ , siendo  $r$  la longitud máxima de río que se puede encontrar en el texto variando el ancho dentro de los límites posibles..

### Ejemplo de entrada / salida

Entrada	Salida
El Amazonas es el río más largo de Suramérica y el Magdalena es el más largo que fluye enteramente en Colombia<CR>	16 5 10 5 15 6
Mambrú se fue a la guerra qué dolor qué dolor qué pena<CR>	
Carito me habla en inglés qué bonito se le ve Carito me habla en inglés que me dice yo no sé<CR>	

## 4 ENTREGABLES

El proyecto puede desarrollarse por grupos de uno o dos estudiantes de la misma sección. La entrega se hace por Sicua+ (una sola entrega por grupo de trabajo).

El grupo debe entregar, por Sicua+, un archivo de nombre `proyectoDAlgo.zip`. Este archivo es una carpeta de nombre `proyectoDAlgo`, comprimida en formato *ZIP*, dentro de la cual hay archivos fuente de soluciones propuestas y archivos que documentan cada una de las soluciones.

### 4.1 Archivos fuente de soluciones propuestas

Todos los programas implementados en *Java* deben compilar en *JDK 8*.

Para el problema  $X$ , siendo  $X \in \{A, B, C\}$ :

- Entregar un archivo *Java* (`.java`) con su código fuente de la solución que se presenta.
- Incluir como encabezado de cada archivo fuente un comentario que identifique el (los) autor(es) de la solución.
- Denominar `ProblemaX.java` el archivo de la solución que se presente.

Nótese que, si bien puede utilizarse un *IDE* como *Eclipse* durante el desarrollo del proyecto, la entrega requiere incluir solo un archivo `.java` por cada solución. El archivo debe poderse compilar y ejecutar independientemente (sin depender de ninguna estructura de directorios, librerías no estándar, etc.).

### 4.2 Archivos que documentan soluciones propuestas

Toda solución propuesta debe acompañarse de un archivo que la documente, con extensión `.doc`, `.docx` o `.pdf`. El nombre del archivo debe ser el mismo del código *Java* correspondiente. Por ejemplo, si incluyó un archivo `ProblemaB.java`, como solución para el problema B, debe incluirse un archivo `ProblemaB.docx` que lo documente.

Un archivo de documentación debe contener los siguientes elementos:

#### 0 Identificación

Nombre de autor(es)  
Identificación de autor(es)

#### 1 Algoritmo de solución

Explicación del algoritmo elegido. Si hubo alternativas de implantación diferentes, explicar por qué se escogió la que se implementó.

Deseable:

Anotación (contexto, pre- poscondición, invariantes, ...) para cada subrutina o método que se use.

#### 2 Análisis de complejidades espacial y temporal

Cálculo de complejidades y explicación de las mismas. Debe realizarse un análisis para cada solución entregada

#### 3 Comentarios finales

Comentarios al desempeño observado de la solución.

Téngase en cuenta que los análisis de 2 tienen sentido en la medida que la explicación de 1 sea clara y correcta. No se está exigiendo formalismo a ultranza, pero sí que, como aplicación de lo estudiado en el curso, se pueda describir un algoritmo de manera correcta y comprensible.