

Técnicas algorítmicas en ingeniería del software

Grado en Ingeniería del Software (UCM)

EXAMEN FINAL DE FEBRERO

Curso 2016/2017

Ejercicio 1. Colas con prioridad y montículos (2 puntos)

Las gorras de la competición

Queremos organizar una competición en la que participan N equipos, cada uno de los cuales cuenta inicialmente con S_i seguidores, $1 \leq i \leq N$. Cada vez que se juega un partido, el equipo que gana sigue compitiendo, y el perdedor desaparece de la competición. Y no hay más reglas para la organización de los partidos (quién debe jugar con quién) que la de tener al final un único vencedor.

Para cada partido, entregaremos una gorra conmemorativa a todos los asistentes. Sabemos que cada vez que un equipo pierde, todos sus seguidores pasan a serlo del equipo que los ha derrotado, y que a cada partido acuden todos los seguidores (actuales) de los dos equipos.

¿Puedes ayudarnos a organizar los partidos de forma que el número de gorras que tengamos que comprar sea lo menor posible?



Entrada

La entrada está compuesta por diversos casos de prueba, ocupando cada uno de ellos dos líneas: la primera contiene el número N de equipos, un entero entre 1 y 100.000, y la segunda contiene N números enteros entre 1 y 1.000.000, que representan los seguidores iniciales de cada equipo.

La entrada termina con un caso sin equipos ($N = 0$), que no debe procesarse.

Salida

Para cada caso de prueba se deberá escribir una línea con el menor número de gorras necesarias para realizar los partidos que lleven a tener un único vencedor.

Entrada de ejemplo

```
3
1 2 3
4
3 1 4 2
4
30 40 50 60
1
5
0
```

Salida de ejemplo

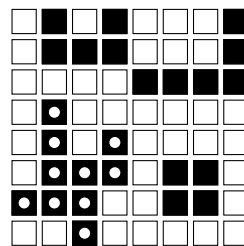
```
9
19
360
0
```

Ejercicio 2. Grafos y estructuras de partición (3 puntos)

Detección de manchas negras

Dado un *bitmap* de píxeles blancos y negros, queremos saber el número de *manchas negras* que contiene y el tamaño (número de píxeles) de la mancha negra más grande. Dos píxeles negros pertenecen a la misma mancha si se puede pasar de uno a otro atravesando solamente píxeles negros y moviéndose píxel a píxel solamente en horizontal o vertical.

Por ejemplo, en el siguiente dibujo (donde los píxeles se han representado mediante cuadrados) aparecen 4 manchas y la mancha más grande (marcada con puntos blancos) tiene 10 píxeles.



Entrada

La entrada estará compuesta por diversos casos de prueba. Para cada caso, la primera línea contendrá el número F de filas y el número C de columnas del bitmap (números entre 1 y 1.000). A continuación aparecerán F líneas, cada una con C caracteres. El carácter ‘-’ representa un píxel blanco y el carácter ‘#’ representa un píxel negro.

Salida

Por cada caso de prueba se escribirá una línea con el número de manchas y el tamaño de la mancha más grande.

Entrada de ejemplo

```
8 8
-#-#---#
-###---#
----####
-#-----
-#-#----
-###-##-
###--##-
--#-----
4 10
#-#-#-###-
#-#-#-#-#-
#-#-#-#-#-
#-#-#-#-#-
```

Salida de ejemplo

```
4 10
4 9
```

Ejercicio 3. Programación dinámica (3 puntos)

Subsecuencia común más larga

Si a una secuencia X de elementos (pongamos por ejemplo, caracteres) le quitamos algunos de ellos y dejamos los que quedan en el orden en el que aparecían originalmente tenemos lo que se llama una *subsecuencia* de X . Por ejemplo, **AAOA** es una subsecuencia de la secuencia **AMAPOLA**.

El término también se aplica cuando quitamos todos los elementos (es decir la secuencia vacía es siempre subsecuencia de cualquier secuencia) o cuando no quitamos ninguno (lo que significa que cualquier secuencia es siempre subsecuencia de sí misma).

Dadas dos secuencias X e Y , decimos que Z es una *subsecuencia común* de X e Y si Z es subsecuencia de X y de Y . Por ejemplo, si $X = \text{AMAPOLA}$ e $Y = \text{MATAMOSCAS}$, la secuencia **AAOA** es una de las subsecuencias comunes a X e Y *más larga*, con longitud 4, ya que no hay ninguna subsecuencia común a X e Y de longitud mayor que 4. También son subsecuencias comunes de longitud 4 **MAOA** o **AMOA**.

Lo que queremos es encontrar la longitud de las subsecuencias comunes más largas de dos secuencias de caracteres dadas.



Entrada

La entrada está compuesta por diversos casos de prueba, estando cada uno de ellos formado por una línea en la que aparecen dos cadenas no vacías (de no más de 1.000 caracteres de la A a la Z) separadas por un espacio.

Salida

Para cada caso de prueba se debe escribir la longitud de las subsecuencias comunes más largas de las dos cadenas leídas.

Entrada de ejemplo

```
AMAPOLA MATAMOSCAS
AAA BBBB
ATAMOS MATAMOSCAS
```

Salida de ejemplo

```
4
0
6
```

Normas de realización del examen

1. Debes programar soluciones para cada uno de los tres ejercicios, probarlas y entregarlas en el juez automático accesible en la dirección <http://exacrc/domjudge/team>.
2. En el juez te identificarás con el nombre de usuario y contraseña que has recibido al comienzo del examen. El nombre de usuario y contraseña que has estado utilizando durante la evaluación continua **no** son válidos.
3. Escribe tu **nombre y apellidos** en un comentario en la primera línea de cada fichero que subas al juez.
4. Puedes descargar el fichero <http://exacrc/docum.zip> que contiene material que puedes utilizar para la realización del examen (diapositivas de clase, implementación de las estructuras de datos, una plantilla de código fuente y ficheros de texto con los casos de prueba de cada ejercicio del enunciado).
5. Los ficheros con las implementaciones de las estructuras de datos están instalados en el juez, por lo que no es necesario subirlos como parte de tu solución (y conviene no hacerlo).
6. Los ejercicios están identificados con el nombre del tema de la asignatura en el que habrían aparecido si hubieran sido propuestos como parte de los ejercicios de la evaluación continua. Para obtener la máxima puntuación, las soluciones deberán seguir los criterios exigidos a los ejercicios de ese tema durante el curso (en cuanto a encapsulación, eficiencia, simplicidad, análisis de costes, etc.).
7. Tus soluciones serán evaluadas por el profesor independientemente del veredicto del juez automático. Para ello, el profesor tendrá en cuenta **exclusivamente** el último envío que hayas realizado de cada ejercicio.
8. Las notas de los ejercicios suman 8 puntos. La calificación obtenida en este examen será sumada a la obtenida por la evaluación continua (sobre 2 puntos) para obtener la calificación final de la asignatura.