Nombre \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ED-Ex3A1-20180314

(0.5 puntos) Haz un proyecto de nombre “ed3a1<tu apellido>” con dos clases:

* la clase MiCadenaTest que te adjuntamos y que se ha generado con JUnit
* la clase MiCadena que has de construir de forma que se ajuste a todas las pruebas de la clase MiCadenaTest y a las especificaciones que siguen.

La clase MiCadena lleva el control de una cadena de texto que se define como un campo String privado.

(1 punto) No tiene ningún constructor público y hemos de usar un método estático de tipo factoría de nombre “fabrica” que devolverá una instancia de la clase MiCadena siempre y cuando le pasemos como parámetro una cadena de longitud superior a 3 que dará valor al String del campo. Si la cadena que se le pasa al método es de longitud 3 o menor no hará la instancia y devolverá el valor null.

(0.2 puntos) Tiene un método “getCadena” que devuelve la cadena del campo.

(0.8 punto) Tiene un método estático “detectaFaltaNB” que devuelva la posición de la cadena que se le pasa como parámetro en la que se produce la falta de ortografía “n antes de b” (tanto en mayúsculas como en minúsculas). Si detecta la falta dice en qué posición está la n, pero si no la detecta devuelve el valor -1.

(1 punto) Tiene un método estático “corrigeFaltaNB” que devuelve una cadena igual a la que se le pasa como parámetro pero con todas las faltas “n antes de b” corregidas. La corrección la hará respetando los tipos: si la n estaba escrita en mayúscula pondrá la m mayúscula, si la n en minúscula la m en minúscula.

(0.5 puntos) Tiene un método “corrigeFaltaNB” sin parámetros que corrige todas las faltas “n antes de b” en la cadena del campo.

(1 punto) Tiene un método “estadistica” que devuelve una cadena con los pares “letra-veces que se repite en el campo” separados por un punto y coma y ordenados por la posición de la letra en el código de caracteres. Sólo saldrán los pares de letras que aparecen al menos una vez.