

Cadenas equilibradas

El objetivo de este control es resolver un problema utilizando TADs lineales vistos en clase (pilas y/o colas).

1) El problema

Una cadena de caracteres que contiene, entre otros símbolos, llaves, paréntesis y corchetes abiertos y cerrados se dice *equilibrada* si, para cada uno de los tres tipos de símbolos mencionados: (i) hay tantos abiertos como cerrados, y (ii) cada vez que aparece uno cerrado el último que apareció fue su correspondiente abierto.

Se debe construir un programa que procese un archivo de entrada que contiene una cadena por línea, y que escriba en el archivo de salida, para cada línea procesada, EQUILIBRADA cuando la cadena esté equilibrada y NO EQUILIBRADA cuando no lo esté.

Ejemplo de entrada / salida:

Entrada	Salida
28*(1+{2*3/[4-5]})/2	EQUILIBRADA
28*(1+{2*3/[4-5]})/2	EQUILIBRADA
[({})]	NO EQUILIBRADA
(({}))	NO EQUILIBRADA
({}))	NO EQUILIBRADA

2) Trabajo a realizar

Se proporciona el archivo `main.cpp` en el que se implementa toda la lógica de entrada / salida necesaria. Hay que añadir a dicho archivo la implementación de una función

```
bool esEquilibrada(const string& cadena);
```

que devuelva `true` si cadena es equilibrada, y `false` en otro caso, así como el resto de definiciones y directivas que se consideren oportuno, siempre y cuando no se modifique la función main.

La solución deberá utilizar también los TADs vistos en clase que se consideren oportunos.