

Trazas de cobre

Contribución de Carlos Mendioroz

Descripción del problema

Una empresa de equipos electrónicos se encuentra desarrollando un sistema de iluminación. Dado que hay más de un equipo de personas abocado al desarrollo, han acordado una interesante interfaz entre los desarrolladores de los LEDs y los que realizan el control de encendido: en una circunferencia de la plaqueta de control quedan determinados puntos que corresponden a los distintos LEDs o a los interruptores de encendido.

Esta disposición les permite cierta flexibilidad, pero se han dado cuenta que ahora necesitan determinar cómo conectar estos puntos (siempre un LED con un control), y para hacerlo deben decidir cuáles unir. Una complicación es que estas uniones no pueden cruzarse pues harían un cortocircuito, y se requiere maximizar la cantidad de LEDs que funcionarán! Si, es increíble pero podría haber más (o menos) LEDs que puntos de control.

Para ayudar a los técnicos de la empresa, se te pide que escribas una función `conecta(cadena, pares)`

Donde:

cadena es una secuencia de a lo más

1000 letras "L" o "C" representando el orden y función de los puntos en la circunferencia. La numeración de puntos arranca en 1

pares es un vector de enteros donde se almacenarán las conexiones propuestas, colocando siempre el punto C delante de su correspondiente punto L.

La función devuelve un entero: el número de conexiones que se pudieron establecer y almacenar en `conecta`.

Ejemplo

Si **cadena** tuviera:

L L L L C C C L C

La función devolvería un **4** y **pares** podría contener:

5 3 7 8 6 2 9 1

Detalles de implantación

Debes enviar exactamente un archivo, llamado **pistas.pas**, **pistas.cpp** o **pistas.c**, que implemente la función **conecta** tal como está descrita arriba usando los siguientes encabezamientos:

En C/C++ : `int conecta((char cadena[], int pares[]);`

Debes incluir también un archivo de encabezamiento **pistas.h**

En Pascal : `type ArregloPares = array [1..2000] of longint;`

`function conecta (cadena : string ;`

`var pares: ArregloPares) : longint ;`

Recuerda usar la llave {\$H}

El evaluador local lee la entrada por consola la cual consta de una sola línea conteniendo la `cadena`.

Entrega por consola una línea con el valor de la función seguido de tantas línea como indica éste número, cada una conteniendo un par de números extraídos secuencialmente del vector `pares`, al como se indica al lado.

4
5 3
7 8
6 2
9 1