



RECLUTAMIENTO – DESAFIO TECNICO

20 de agosto de 2020

Descripción

El siguiente es un desafío que tiene como objetivo captar los conocimientos y habilidades del postulador. Se tiene 3 días corrido para resolver el desafío y se cuenta desde el envío de el mismo.

Preguntas

1) Crear una función que devuelva una versión simplificada de una fracción

Ejemplos

`Simplificar("4/6") = "2/3"`

`Simplificar("10/11") = "10/11"`

`Simplificar("100/400") = "10/4";`

Notas: una fracción simplificada no tiene divisores comunes mínimos (excepto 1) entre el numerador y el denominador. Por ejemplo, 4/6 no esta simplificada, ya que tanto 4 y 6 comparten 2 como factor

Si una fracción puede ser transformada en entero, también debe tenerse en cuenta

2) Validar Nombres

Tener en cuenta las siguientes definiciones

- a) El termino ingresado pueden ser iniciales o palabras completas
- b) Una inicial es solo un caracter mas un punto
- c) Una palabra se comprende de 2 o mas caracteres, sin punto.

Un nombre Válido se puede escribir de alguna de las siguientes formas:

E. Poe

E. A. Poe

Edgard Allan Poe

Edgard A. Poe

Los siguientes ejemplos son inválidos:

Edgard (solo nombres o apellidos no son validos)

E Poe o E. A Poe (las iniciales deben contener un punto)

e. Poe o E. poe o e. a. Poe (sin mayúsculas)

E. Allan Poe (el segundo nombre completo, mientras que el primero es solo una inicial)

E. A. P. (el apellido no es una palabra completa)

Edg. A. Poe (la inicial debe ser una sola letra)

Reglas

a) Tanto las iniciales como las palabras completas deben estar capitalizadas (la primera letra en mayúsculas)

b) Las iniciales deben terminar en punto (.)

c) Solo habrán 2 o 3 términos en el ingreso. Es decir, o dos nombres y un apellido o solo un nombre y un apellido

- d) Si se ingresan dos nombres y un apellido, los dos primeros pueden ser iniciales, o solo el segundo. Nunca puede ser una inicial el primer nombre y no el segundo
- e) El apellido siempre debe ser una palabra completa

La tarea consiste en escribir una función que, siguiendo las reglas anteriores, devuelva si un nombre es válido (true) o no (false).

Ejemplos:

```
ValidName("E. Poe") → true
ValidName("E. A. Poe") → true
ValidName("Edgard A. Poe") → true
ValidName("Edgard") → false
// deben ser 2 o tres palabras
ValidName("e. Poe") → false
// capitalizacion
ValidName("E Poe") → false
// falta el punto detrás de la inicial
ValidName("E. Allan Poe") → false
// no es valido: inicial como primer nombre y palabra como segundo
ValidName("E. Allan P.") → false
// Apellido como inicial
ValidName("Edg. Allan Poe") → false
// Las palabras no pueden finalizar con punto (solo iniciales)
```