()[]{}<>

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos | 22.11 | Límite de memoria | 32MB |
| Límite de tiempo (caso) | 1s | Límite de tiempo (total) | 60s |
| Límite de entrada (bytes) | 10 KiB |  |  |

**Descripción**

Considera una cadena SS de caracteres, que esta conformada solo por 8 posibles caracteres, '<', '>', '(', ')', '{', '}', '[' , ']'. Como sabrás, estos caracteres regularmente se utilizan para agrupar distintas cosas, ya sean operaciones, oraciones, etc, sin embargo, cuando son usadas debe cuidarse que la agrupación tenga sentido, por ejemplo ")([]" "no tiene sentido" porque los primeros dos caracteres no agrupan nada. Un ejemplo de cadena que "tiene sentido" es [{}{()}] , ya que cada pareja de caracteres puede emparejarse con otra para darle sentido a la agrupación. Formalmente, la cadena de caracteres 'S' "tiene sentido" si se cumple cuando menos 1 de las siguientes condiciones:

a) SS="""" , es decir , es una cadena vacía.

b) SS es una concatenación de cadenas mas pequeñas que "tienen sentido".

c) SS ="'<'Q'>'" o SS="{Q}" o SS="[Q]" o SS="(Q)", donde Q es una cadena que "tiene sentido" mas pequeña que "S".

Tu tarea es la siguiente, dada una cadena de caracteres SS , que estará conformada solo por algunos de los siguientes caracteres '<', '>', '(', ')', '{', '}', '[' , ']', debes encontrar el mínimo numero de "reemplazos" que conviertan a SS en una cadena que "tiene sentido". Donde caracteres de apertura ('<' , '(', '{', '[' ) solo pueden cambiarse por otros caracteres de apertura, y caracteres de cerradura ('>', ')', '}' , ']' ) , solo pueden cambiarse por caracteres de cerradura.

**Entrada**

Una cadena de caracteres SS, sin espacios entre ellos.

**Salida**

Un solo entero, la mínima cantidad de "reemplazos" que se pueden hacer a SS para que "tenga sentido". Si es imposible lograr que SS "tenga sentido" con ninguna cantidad de reemplazos, debes imprimir "imposible".

**Ejemplo**

| **Entrada** | **Salida** | **Descripción** |
| --- | --- | --- |
| [) | 1 | De inicio la cadena no tiene sentido, pero cambiando el caracter ')' por un ']', ahora tiene sentido. |
| {<>} | 0 | La cadena ya tiene sentido, no es necesario hacer cambios. |
| {{ | imposible | Sin importar cual cambio se haga, no es posible hacer que la cadena tenga sentido. |

**Límites**

Sea NN la cantidad de caracteres que tiene SS.

En un conjunto de casos con valor del 2020 % la cadena esta conformada por caracteres de 1 solo tipo (donde los tipos son <> , () , {} , [] ), de aquí:

\*En 1010 % de los casos 1≤N≤1031≤N≤103

\*En 1010 % de los casos 1≤N≤1051≤N≤105

En un conjunto de casos con valor del 4040 % la cadena esta conformada por caracteres de solo 2 tipos, de aquí:

\*En 2020 % de los casos 1≤N≤1031≤N≤103

\*En 2020 % de los casos 1≤N≤1051≤N≤105

En un conjunto de casos con valor del 4040 % la cadena esta conformada por caracteres de 3 ó 4 tipos, de aquí:

\*En 2020 % de los casos 1≤N≤1031≤N≤103

\*En 2020 % de los casos 1≤N≤1051≤N≤105

*Fuente: CF*

Problema subido por: [José Tapia](https://omegaup.com/profile/Josetapia/)

[Reportar contenido inapropiado en este problema.](https://omegaup.com/arena/problem/ParentesisExtravagantes)

| **Enviado** | **GUID** | **Estatus** | **Porcentaje** | **Lenguaje** | **Memoria** | **Tiempo** | **Detalles** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| [Nuevo envío](https://omegaup.com/arena/problem/ParentesisExtravagantes#problems/new-run) | | | | | | | | |
| 2019-01-16 18:29:35 | 9393b421 | Respuesta correcta | 100.00% | cs | 15.92 MB | 3.71 s |  |  |
| 2019-01-16 18:02:36 | 2f1c1b0f | Error de ejecución | 68.55% | java | 0.00 MB | 5.71 s |  |  |
| 2019-01-16 17:41:38 | 4fa5630a | Error de ejecución | 68.55% | cpp | 3.27 MB | 0.19 s |  |  |
| 2019-01-16 16:06:43 | 21ebd975 | Error de ejecución | 68.55% | cs | 16.17 MB | 3.89 s |  |  |
| **Envíos** | | | | | | | |

**Opiniones de los usuarios**

**Calidad**

3.3

👥 6 votos en total

Muy bueno

**2**

Bueno

**4**

Regular

**0**

Malo

**0**

Muy malo

**0**

**Dificultad**

1.3

👥 6 votos en total

Muy fácil

**1**

Fácil

**3**

Medio

**1**

Difícil

**1**

Muy difícil

**0**

| **Usuario** | **Lenguaje** | **Memoria** | **Tiempo** | **Enviado** |
| --- | --- | --- | --- | --- |
| [Huvok](https://omegaup.com/profile/Huvok) | cpp11 | 3.30 | 0.03 | 2018-07-09 15:08:42 |
| [LeoVertiz](https://omegaup.com/profile/LeoVertiz) | cpp | 2.84 | 0.03 | 2018-05-06 20:38:51 |
| [Pino](https://omegaup.com/profile/Pino) | cpp11 | 3.27 | 0.03 | 2018-08-25 02:29:50 |
| [omichzarkito0](https://omegaup.com/profile/omichzarkito0) | cpp11 | 3.21 | 0.04 | 2018-08-28 04:31:42 |
| [5an7y](https://omegaup.com/profile/5an7y) | cpp11 | 3.77 | 0.06 | 2017-04-12 23:17:57 |
| [MiguelSPro](https://omegaup.com/profile/MiguelSPro) | c | 1.57 | 0.06 | 2018-10-03 05:05:11 |
| [MCwarrior](https://omegaup.com/profile/MCwarrior) | cpp | 3.29 | 0.10 | 2017-11-01 22:17:40 |
| [ascher](https://omegaup.com/profile/ascher) | cpp11 | 3.11 | 0.10 | 2018-01-18 04:55:14 |
| [Josetapia](https://omegaup.com/profile/Josetapia) | cpp11 | 3.70 | 0.11 | 2017-04-01 10:24:04 |
| [JorgeGlz0820](https://omegaup.com/profile/JorgeGlz0820) | cpp | 3.29 | 0.11 | 2017-12-13 13:18:51 |
| **Mejores envíos aceptados** | | | | |

<https://omegaup.com/arena/problem/ParentesisExtravagantes#problems>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

class Principal

{

static string EvaluarParentesis(string s)

{

string ap = "<({[";

string ci = ">)}]";

Dictionary<char, char> dic = new Dictionary<char, char>();

for (int i = 0; i < ap.Length; i++)

{

dic[ap[i]] = ci[i];

}

Stack<char> stack = new Stack<char>();

int cambios = 0;

for (int i = 0; i < s.Length; i++)

{

if (ap.Contains(s[i]))

{

stack.Push(s[i]);

}

else if (ci.Contains(s[i]))

{

if (stack.Count == 0)

{

//Console.WriteLine("imposible");

//break;

return "imposible";

}

char ab = stack.Pop();

if (dic[ab] != s[i])

{

cambios++;

}

}

}

if (stack.Count > 0)

{

//Console.WriteLine("imposible");

return "imposible";

}

//else

//{

// Console.WriteLine(cambios);

//}

return cambios.ToString();

}

static void Main(string[] args)

{

//string s = "{<>}";

//string s = "{{";

//string s = Console.ReadLine().Trim();

//string s = "()()[dddd333][333]333[}[333}{{3333]>";

string s = Console.ReadLine().Trim();

//string s = "[[}}";

Console.WriteLine(EvaluarParentesis(s));

//<> , () , {

//} , []

Console.ReadLine();

}

}