

Problema 2

Descripción

Una cadena de paréntesis es *balanceada* si cada paréntesis de apertura tiene un paréntesis de cierre correspondiente. Dada una cadena de paréntesis, decidir si es balanceada o no.

Entrada

Cadena de paréntesis.

Salida

La salida debe constar de dos líneas. La primera línea debe ser la cantidad de paréntesis de apertura en la cadena. La segunda línea debe indicar "Balanceada" en caso de que la cadena sea balanceada, "No balanceada" en caso contrario.

Ejemplo

entrada	salida
((()))()	4 Balanceada
)(())()	3 No balanceada

Solución:

Podemos contar los paréntesis de apertura y de cierre. Si en algún momento hay más paréntesis de cierre, significa que la cadena ya no es balanceada ya que no puede existir uno de apertura para ese paréntesis de cierre. Al final de procesar la cadena, si tenemos el mismo número de paréntesis de apertura y de cierre, podemos concluir que la cadena es balanceada.

```
#include<iostream>
using namespace std;
int main ()
{
    string str;
    cin >> str;
    int apertura = 0;
    int cierre = 0;
    bool noBalanceada = false;

    for (int i = 0; i < str.length(); i++)
    {
        if (str.at(i) == '(')
        {
            apertura++;
        }
        else
        {
            cierre++;
        }
    }
}
```

```
    if (cierre > apertura)
    {
        noBalanceada = true;
    }
}

cout << apertura << endl;

if (noBalanceada == true)
{
    cout << "No balanceada";
}
else if (apertura == cierre)
{
    cout << "Balanceada";
}
else
{
    cout << "No balanceada";
}
}
```

