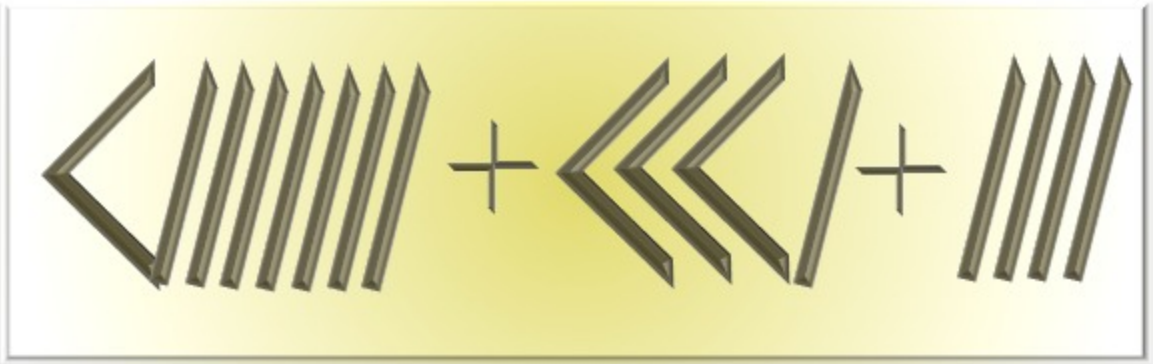


考古学者兼ITエンジニアであるあなたはパイザ遺跡の発掘に成功し、そこに数式のようなものが刻まれていることに気づきました。



あなたはこの数式の記法をパイザ記法と命名し、解釈を試みたところ、以下のルールを見つけ出しました。

- ・数式はすべて 2 個以上の整数の足し算となっている
- ・使用する整数は 1 から 99 までで、10 進法を用いる
- ・連続する "<" の数が整数の 10 の位を表し、それに続く連続する "/" (スラッシュ) の数が整数の 1 の位を表す
- ・"+" が足し算の記号を表す

このルールをもとに、パイザ記法で書かれた数式を解釈し、その結果を返すプログラムを作成してください。なお、上記のルールで解釈不能な入力は与えられません。

例)

```
///+////
```

→ 3 + 4 = 7

```
//+////////
```

→ 2 + 6 = 8

次の例は上の図を入力とした場合を示しています。

```
<////////+<<</+////
```

→ 17 + 31 + 4 = 52

🚩 評価ポイント

10回のテストケースで、正答率、実行速度、メモリ消費量をはかり得点が決まります。より早い回答時間で提出したほうが得点が高くなります。

1. 複数のテストケースで正しい出力がされるか評価（+50点）
2. 解答までの速さ評価（+50点）

📄 入力される値

入力は以下のフォーマットで与えられます。

```
E
```

- ・1 行目にはパイザ記法で書かれた数式を表す文字列 E が与えられます。
- ・入力は 1 行で、末尾に改行が 1 つ入ります。

それぞれの値は文字列で標準入力から渡されます。[標準入力からの値取得方法はこちらをご確認ください](#) 📖

📄 期待する出力

数式 E の答えを整数で出力してください。

最後は改行し、余計な文字、空行を含んではいけません。

✅ 条件

すべてのテストケースにおいて、以下の条件をみます。

- ・E は半角記号 "<", "/", "+" で構成される
- ・3 ≤ (E の長さ) ≤ 100

入力例1

```
///+////
```

出力例1

7

入力例2

```
<////////+<<</+////
```

出力例2

52

入力例3

```
<<<<<<<<<////////+<<<<<<<<<////////
```

出力例3

198