Your task is to write a program that computes the result of adding a sequence of fractions. Using the definition

```c
struct Fraction {
    int num, den; // always strictly positive
};
```

your program must include and use the function

```c
Fraction addition(const Fraction &x, const Fraction &y);
```

that returns the addition of \(x\) and \(y\), without common factors in the numerator and denominator.

**Input**

The input is a sequence of one or more simplified fractions separated by plus signs, ended with an equal sign. Each fraction consists of its numerator, a bar, and its denominator. Numerators and denominators are natural strictly positive.

**Output**

Your program must print the simplified fraction corresponding to the sum of all the given fractions.

**Observations**

- In order to avoid overflows, use the function `addition()` to accumulate the partial calculations.
- Inefficient calculation of the greatest common divisor will be negatively valued.
- Using vectors is not allowed to solve this problem.

**Sample input 1**

```
1/2 + 1/2 =
```

**Sample output 1**

```
1/1
```

**Sample input 2**

```
1/2 + 2/3 + 3/4 + 4/5 + 5/6 =
```

**Sample output 2**

```
71/20
```

**Sample input 3**

```
1/10125 + 1/8000 + 1000/999 =
```

**Sample output 3**

```
4801073/4795200
```

**Sample input 4**

```
9/4 =
```

**Sample output 4**

```
9/4
```