1081. Rational Sum (20)

时间限制

400 ms

内存限制

65536 kB

代码长度限制

16000 B

判题程序

Standard

作者

CHEN, Yue

Given N rational numbers in the form "numerator/denominator", you are supposed to calculate their sum.

**Input Specification:**

Each input file contains one test case. Each case starts with a positive integer N (<=100), followed in the next line N rational numbers "a1/b1 a2/b2 ..." where all the numerators and denominators are in the range of "long int". If there is a negative number, then the sign must appear in front of the numerator.

**Output Specification:**

For each test case, output the sum in the simplest form "integer numerator/denominator" where "integer" is the integer part of the sum, "numerator" < "denominator", and the numerator and the denominator have no common factor. You must output only the fractional part if the integer part is 0.

**Sample Input 1:**

5

2/5 4/15 1/30 -2/60 8/3

**Sample Output 1:**

3 1/3

**Sample Input 2:**

2

4/3 2/3

**Sample Output 2:**

2

**Sample Input 3:**

3

1/3 -1/6 1/8

**Sample Output 3:**

7/24

[提交代码](https://www.patest.cn/contests/pat-a-practise/1081)

#include<iostream>

#include<string>

#include<algorithm>

#include<queue>

#include<vector>

#include<sstream>

#include<stack>

#include<map>

#include<cstring>

#include<climits>

#include<cmath>

#define MAX 1001

#define LL long long

using namespace std;

struct item

{

LL integ = 0;

LL num = 0;

LL domi = 0;

};

LL gcd(LL a, LL b)

{

LL r;

while (r = a%b)

{

a = b;

b = r;

}

return b;

}

item addition(item a, item b)

{

item result;

result.integ = a.integ + b.integ;

if (a.num != 0 && b.num != 0)

{

LL LCD = a.domi\*b.domi / gcd(a.domi, b.domi);

result.num = a.num\*LCD / a.domi + b.num\*LCD / b.domi;

result.domi = LCD;

if (result.num != 0)

{

LL GCD = gcd(result.domi, result.num);

result.domi /= GCD;

result.num /= GCD;

if (result.num / result.domi)

{

result.integ += result.num / result.domi;

result.num = result.num%result.domi;

}

}

}

else if (a.num == 0)

{

result.num = b.num;

result.domi = b.domi;

}

else

{

result.num = a.num;

result.domi = a.domi;

}

if (result.integ > 0 && result.num < 0 && result.domi != 0)

{

result.num = result.integ\*result.domi + result.num;

result.integ = result.num / result.domi;

result.num = result.num%result.domi;

}

else if (result.num > 0 && result.integ < 0 && result.domi != 0)

{

result.num = result.integ\*result.domi + result.num;

result.integ = result.num / result.domi;

result.num = result.num%result.domi;

}

if (result.num == 0)

result.domi = 0;

return result;

}

void output(item total)

{

int zeroflag = 0;

int middleflag = 0;

if (total.integ&&total.num)

middleflag = 1;

if (total.integ != 0)

{

cout << total.integ;

zeroflag = 1;

}

if (middleflag)

cout << " ";

if (total.num != 0)

{

if(total.domi>0)

printf("%lld/%lld", total.num, total.domi);

else

printf("%lld/%lld", -total.num, -total.domi);

zeroflag = 1;

}

if (!zeroflag)

cout << "0";

}

int main()

{

int num;

cin >> num;

item total;

while (num--)

{

item addi;

scanf("%lld/%lld", &addi.num, &addi.domi);

if (addi.domi != 0)

{

addi.integ = addi.num / addi.domi;

addi.num %= addi.domi;

}

if (addi.domi < 0)

{

addi.num -= 2 \* addi.num;

addi.domi -= 2 \* addi.domi;

}

total = addition(total, addi);

}

output(total);

}