```c
#include <stdio.h>
#include <stdlib.h>
#include "headers/i2p.h"

/*FUNCTION main
A.1 Declare variables: operatorExpr, num1, num2, num3, num4, numOfNumbers, i
A.2 Display the message "How many times: "
A.3 Get the number of iterations (numOfNumbers)
A.4 Loop for i < numOfNumbers with a step of 1
A.4.1 Call the function readExpression to get the expression parameters
A.4.2 Use the value of operatorExpr to perform the appropriate arithmetic operation
A.4.2.1 Call the functions add, sub, multi, or divide based on the operator
A.5 End the program

FUNCTION add
B.1 Declare variables: upper, base
B.2 Calculate the common denominator (base)
B.3 Calculate the products of the numerators (num1 * num4, num3 * num2)
B.4 Add the multiplied numerators
B.5 Call the simplify function with the parameters upper, base

FUNCTION sub
C.1 Declare variables: upper, base
C.2 Calculate the common denominator (base)
C.3 Calculate the products of the numerators (num1 * num4, num3 * num2)
C.4 Subtract the multiplied numerators
C.5 Call the simplify function with the parameters pper, base

FUNCTION multi
D.1 Declare variables: upper, base
D.2 Calculate the common denominator (base)
D.3 Calculate the product of the numerators (num1 * num3)
D.4 Call the simplify function with the parameters upper, base

FUNCTION divide
E.1 Declare variables: upper, base
E.2 Calculate the common denominator (base)
E.3 Calculate the product of the numerators (num1 * num4)
E.4 Call the simplify function with the upper, base

FUNCTION simplify
F.1 Declare variable: i
F.2 Loop for each i from 1 to base with a step of 1
F.2.1 Check if i divides both numerators
F.2.1.1 Divide the numerators by i
F.2.1.2 Set i to 2 and repeat the check
F.3 Display the simplified expression based on the case
*/

//St prototypes
void add(int num1, int num2, int num3, int num4);

void sub(int num1, int num2, int num3, int num4);

void multi(int num1, int num2, int num3, int num4);

void divide(int num1, int num2, int num3, int num4);

void simplify(int upper, int base);

int main() {
    //Variables
    char operatorExpr;
    int num1, num2, num3, num4;
    int numOfNumbers, i;

    //Get number of times u want to run the program
    printf("How many times : ");
    scanf("%d" , &numOfNumbers);

    for (i = 0; i < numOfNumbers; i++) {
        //Get the expression parameters
        readExpression(&operatorExpr, &num1, &num2, &num3, &num4);
```

```
74:
75:         //Use the value to do the correct operation
76:         switch(operatorExpr){
77:
78:             case '+':
79:                 //Add numbers if +
80:                 add(num1, num2, num3, num4);
81:                 break;
82:             case '-':
83:                 //Remove numbers if -
84:                 sub(num1, num2, num3, num4);
85:                 break;
86:             case '*':
87:                 //Multiply number if *
88:                 multi(num1, num2, num3, num4);
89:                 break;
90:             case '/':
91:                 //Divide number if /
92:                 divide(num1, num2, num3, num4);
93:                 break;
94:             default:
95:                 break;
96:         }
97:     }
98:     return 0;
99: }
100:
101: void add(int num1, int num2, int num3, int num4) {
102:     //Set variables
103:     int upper, base;
104:     //Get the same base
105:     base = num2 * num4;
106:     //Multiply the upper numbers
107:     num1 = num1 * num4;
108:     num3 = num3 * num2;
109:     //Add them
110:     upper = num1 + num3;
111:     simplify(upper, base);
112: };
113:
114: void sub(int num1, int num2, int num3, int num4) {
115:     //Set variables
116:     int upper, base;
117:     //Get the same base
118:     base = num2 * num4;
119:     //Multiply the upper numbers
120:     num1 = num1 * num4;
121:     num3 = num3 * num2;
122:     //Sub them
123:     upper = num1 - num3;
124:     //Simplify
125:     simplify(upper, base);
126: }
127:
128: void multi(int num1, int num2, int num3, int num4) {
129:     //Set variables
130:     int base, upper;
131:     //Get same base
132:     base = num2 * num4;
133:     //Get upper number
134:     upper = num1 * num3;
135:     //Simplify
136:     simplify(upper, base);
137: }
138:
139: void divide(int num1, int num2, int num3, int num4){
140:     //Set variables
141:     int base, upper;
142:     //Get same base
143:     base = num2 * num3;
144:     //Get upper number
145:     upper = num1 * num4;
146:     simplify(upper, base);
```

```c
}

void simplify(int upper, int base) {
    //Set i variable
    int i, num = 0;
    //Loop for the base
    for (i = 2; i <= base; i++ ) {
        //Check to see if i divides both numbers
        if (upper % i == 0 && base % i == 0) {
            //Divide the numbers
            upper = upper / i;
            base = base / i;
            //Set i to 2 and repeat
            i = 2;
        }
    }

    //Check case and print the matching case
    if (upper == base) {
        printf("\t1\n");
    }
    //Check if upper > base
    else if (upper > base) {
        do {
            //Remove from the upper the base and 1 to number
            num++;
            upper -= base;
        } while (upper >= base);
        //If there is no leftover
        if (upper == 0) {
            printf("\t%d\n", num);
        } else printf("\t%d %d/%d", num, upper, base);
    }
    else {
        //Print the numbers
        printf("\t%d/%d\n", upper, base);
    }
}
```