3 Floating-Point Numbers in Binary

3.1 Using Floating-Point Numbers

C provides three floating point types: `float`, `double`, and `long double`. Though the sizes of these types are in that order (the size of the latter is no less than that of the former), the latter is not always bigger than the former.

The function `printf` supports two format strings, `%f` and `%Lf`, for floating-point numbers. The former (`%f`) is for `float` and `double` and the latter (`%Lf`) is for `long double`. Though `printf` uses `%f` for `double`, note that `scanf` uses `%lf` for `double *`.

The following shows a sample program to check if the amount of pizza shared by several persons is no less than 1/2:

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int pizza = 0, person = 1;
6      double share = 0.0;
7
8      fscanf(stdin, "%d%d", &pizza, &person);
9      share = pizza;
10     share /= person;     // share = share / person
11     fprintf(stdout, "%f pizza/person\n", share);
12     if (share >= 0.5)
13         fputs("No less than 0.1 (in binary) pizzas can be shared!\n", stdout);
14
15     return 0;
16 }
```

Beware that the division operator `/` performs an integer division if both operands are integers. It performs a floating-point division if at least one of them is a floating-point number. The division in Line 10 is a floating-point division since `share` is of `double` (double-precision floating-point type).

3.2 Programming Lab 3: fbits.c

Given a positive integer n and a positive floating-point number x ($0 < x < 1$), print the binary representation of x in n binary digits after the decimal point. For example, if $n = 3$ and $f = 0.29$, your program should print `0.010` as output.

The input consists of two lines in standard input. The first line contains the number of binary digits n and the second line contains the floating-point number f. Your program should print the binary representation of f in n binary digits after the decimal point.

Additional requirements for bonus points

- Check the quality of your code to confirm that there are no style issues.

Input	Output
5 0.29	0.01001
10 0.9	0.1110011001