

The input data for each problem should to be read from the text file and the answer should to be written to the fixed file *OUTPUT.TXT*. The program has to ask the user for the name of the input file. All the programs will be tested on IBM PC 486 DX2, 65 MHz computers.

RECTANGLES (Latvia)

20 points, run time limit – 10 sec.

Let *ABCD* be a rectangle with the following vertices: *A*(0; 0), *B*(0; 1), *C*(*k*; 1), *D*(*k*; 0). Let us say that this rectangle has index 1. The proportion of the lengths of the edges is 1 : *k*, see Fig. 1.

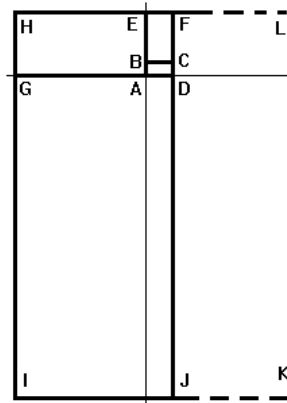


Fig. 1

The second rectangle (with index 2) is built on the "North" edge of the first rectangle (*BC*) with the same proportion of the lengths of the edges (1 : *k*). This rectangle is *BEFC*. $|BC| : |BE| = 1 : k$.

The "West" edge of the common rectangle *AEFD* is *AE*.

The third rectangle is built on this edge *AE* with the same proportion of the lengths of the edges (1 : *k*). This rectangle is *AGHE*. $|AE| : |AG| = 1 : k$.

In the same way next rectangles are built on the edges of the previous common rectangles with the same proportion of the edges and sequence of directions of the base edge ("South", "East", "North", "West", "South", ...).

Write a program that finds out for the given whole numbers *k*, *x*, *y* ($1 < k < 100, -10^9 < x, y < 10^9$) the index of the rectangle which covers the point *P*(*x*; *y*).

If the point *P* is on the edge or at the vertex, assume that it is covered by the rectangle of a minimal index.

Input Data

The input data will come from the ASCII file containing only one line with the values of *k*, *x*, *y*, separated by a single space character.

Output

Your answer must be written to the ASCII file *OUTPUT.TXT* containing only one integer.

Examples

(files *REC1.DAT*, *REC2.DAT*, *REC3.DAT*)

INPUT	OUTPUT
3 0 1	1
2 7 -2	5
4 1 17	2

The input data for each problem should to be read from the text file and the answer should to be written to the fixed file OUTPUT.TXT. The program has to ask the user for the name of the input file. All the programs will be tested on IBM PC 486 DX2, 65 MHz computers.

AN EXPRESS TAXI (Lithuania)

30 points, run time limit – 30 sec.

In a long street the traffic of express taxis is organised in the following way. There is a taxi stop every one kilometer. An express taxi drives along the street from each stop 1, 2, 3, ... or 10 kilometers without a stop.

For each of the ten distances a separate price is fixed and marked in the table, e.g.

1	12
2	21
3	31
4	40
5	49
6	58
7	69
8	79
9	90
10	101

A passenger wants to go n ($1 \leq n \leq 100$) kilometers. Which driving distances does the passenger have to choose so that the journey would be the cheapest, and what is the total price of the journey?

Write a program to solve the problem.

Input Data

The input file has two lines. The first line contains ten integer numbers which are prices for driving 1, 2, 3, ... , 10 kilometers. The second line contains the number n .

Output

Write the results to the file OUTPUT.TXT. Each line but the last one should contain two numbers which are the length of the route and the price of the ticket. The total price of the journey is written in the last line of the output file.

Example

```
INPUT
12 21 31 40 49 58 69 79 90 101
15

OUTPUT
3 31
6 58
6 58
147
```

If several solutions are possible, choose one of them.

The input data for each problem should to be read from the text file and the answer should to be written to the fixed file OUTPUT.TXT. The program has to ask the user for the name of the input file. All the programs will be tested on IBM PC 486 DX2, 65 MHz computers.

IFTHENELSE (Estonia)

50 points, run time limit – 3 min.

Any line of the program in the programming language IFTHENELSE begins with the line number and contains only one command. The variables are of an integer type, the lower-case letters are used as their identifiers.

The input file presents a subprogram (*number of lines* ≤ 20) that computes one integer value and contains only the lines of the following syntax:

```
<line number> IF <ident>=<ident> THEN <line number> ELSE <line number>  
<line number> RETURN(<integer>)
```

where the command RETURN finishes the execution of the subprogram and returns the integer as a value.

Find all the possible results of the execution of the subprogram. Write each of them only once together with such values of all the variables of the subprogram that bring to this result.

Example

The file ITE.DAT contains the subprogram

```
11 IF a=b THEN 12 ELSE 15  
12 IF b=c THEN 13 ELSE 15  
13 IF a=c THEN 14 ELSE 16  
14 RETURN(14)  
15 RETURN(15)  
16 RETURN(77)
```

The answer:

```
14:  
a=1,b=1,c=1  
15:  
a=1,b=2,c=3
```