

PACIFIC NORTHWEST REGION  
PROGRAMMING CONTEST  
DIVISION 2



November 3rd, 2018



**icpc** International Collegiate  
Programming Contest



north america  
sponsor



programming  
tools sponsor

2018 ICPC Pacific Northwest Regional Contest

## 2018 ICPC Pacific Northwest Regional Contest

### Reminders

- For all problems, read the input data from standard input and write the results to standard output.
- In general, when there is more than one integer or word on an input line, they will be separated from each other by exactly one space. No input lines will have leading or trailing spaces, and tabs will never appear in any input.
- Platform is as follows:

```
Ubuntu 18.04.1 LTS Linux (64-bit)
GNOME
```

```
vi/vim
gvim
emacs
gedit
geany
kate
```

```
Java OpenJDK version 10.0.2
C gcc 7.3.0
C++ g++ 7.3.0
Python 2.7.13 (implemented using PyPy 5.10.0).
CPython 3.6.6.
C# via Mono 5.10.1.57
Kotlin 1.2.70
```

Eclipse 4.8 (Photon), configured with:

```
Java
C/C++
PyDev
```

IntelliJ (IDEA Community Edition 2018.2.4), configured with:

```
Java
Kotlin
```

CLion (version 2018.2.4), configured with

```
C/C++
```

Pycharm Community Edition Python IDE version 2018.2.4

Code::Blocks (version 13.12+dfsg-4), configured with

```
Java (OpenJDK version 1.8.0_131)
C/C++ (CDT 9.3.0 with Ubuntu 5.4.0-6ubuntu16.04.4 5.4.0 20160609)
```

MonoDevelop 7.5.0.1254

```
Kotlinc 1.2.70
```

- Compiler options are as follows:

## 2018 ICPC Pacific Northwest Regional Contest

```
gcc -g -O2 -std=gnu11 -static $* -lm
g++ -g -O2 -std=gnu++14 -static $*
javac -encoding UTF-8 -sourcepath . -d . $*
python2 -m py_compile $*
python3 -m py_compile $*
mcs $*
kotlinc -d . ${files}
```

- Execution options are as follows:

```
java -Dfile.encoding=UTF-8 -XX:+UseSerialGC -Xss64m -Xms1920m -Xmx1920m $*
pypy $*
mono $*
kotlin -Dfile.encoding=UTF-8 -J-XX:+UseSerialGC -J-Xss64m -J-Xms1920m -J-Xmx1920m
```

- Python may not have sufficient performance for many of the problems; use it at your discretion. This is especially true of Python 3.

## 2018 ICPC Pacific Northwest Regional Contest

PROBLEM N — LIMIT 1 SECOND

# Exam



Your friend and you took a true/false exam of  $n$  questions. You know your answers, your friend's answers, and that your friend got  $k$  questions correct.

Compute the maximum number of questions you could have gotten correctly.

## Input

The first line of input contains a single integer  $k$ .

The second line contains a string of  $n$  ( $1 \leq n \leq 1000$ ) characters, the answers you wrote down. Each letter is either a 'T' or an 'F'.

The third line contains a string of  $n$  characters, the answers your friend wrote down. Each letter is either a 'T' or an 'F'.

The input will satisfy  $0 \leq k \leq n$ .

## Output

Print, on one line, the maximum number of questions you could have gotten correctly.

## Sample Input and Output

3 FTFFF TFTTT	2
---------------------	---

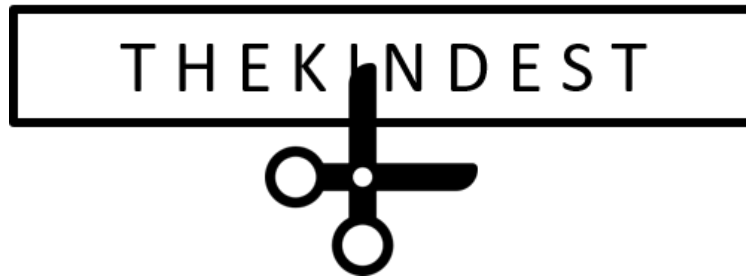


2018 ICPC Pacific Northwest Regional Contest

6 TTF1FF1FTF TTTTFFTTTT	9
-------------------------------	---

PROBLEM O — LIMIT 1 SECOND

## Paper Cuts



Tito has a paper strip with some letters written on it. He would like to rearrange the letters to form some other word. He does this by making some number of vertical cuts and then rearranging the remaining strips of paper. For example, a strip with letters

```
abbaaddccee
```

can be cut into four pieces,

```
abb | aa | ddcc | ee
```

and be put together in a different order:

```
aaabbeeddcc
```

Given Tito's paper strip and the word he wants to form, determine the minimum number of cuts that Tito needs to make in order to construct the desired word.

### Input

The first line of input contains a single string of lowercase letters, representing Tito's paper strip.

The second line of input contains a single string of lowercase letters, representing the word Tito wants to form by rearranging the letters.

It is guaranteed that the two lines consist of the same number of letters between 1 and 18 (inclusive), and that the letters consisting the two lines are exactly the same, *i.e.*, it is always possible to reach Tito's desired word by rearranging the letters in the paper strip.

### Output

Print, on a single line, the minimum number of cuts that Tito has to make.

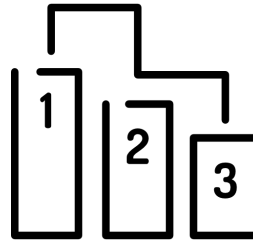
2018 ICPC Pacific Northwest Regional Contest

**Sample Input and Output**

abbaaddccee aaabbeeddcc	3
abba abba	0



PROBLEM P — LIMIT 1 SECOND  
**Contest Setting**



A group of contest writers have written  $n$  problems and want to use  $k$  of them in an upcoming contest. Each problem has a *difficulty level*. A contest is valid if all of its  $k$  problems have different difficulty levels.

Compute how many distinct valid contests the contest writers can produce. Two contests are distinct if and only if there exists some problem present in one contest but not present in the other.

Print the result modulo 998,244,353.

**Input**

The first line of input contains two space-separated integers  $n$  and  $k$  ( $1 \leq k \leq n \leq 1000$ ).

The next line contains  $n$  space-separated integers representing the difficulty levels. The difficulty levels are between 1 and  $10^9$  (inclusive).

**Output**

Print the number of distinct contests possible, modulo 998,244,353.

**Sample Input and Output**

<pre>5 2 1 2 3 4 5</pre>	<p>10</p>
<pre>5 2 1 1 1 2 2</pre>	<p>6</p>



2018 ICPC Pacific Northwest Regional Contest

12 5 3 1 4 1 5 9 2 6 5 3 5 8	316
---------------------------------	-----

## 2018 ICPC Pacific Northwest Regional Contest

PROBLEM Q — LIMIT 1 SECOND

# Poker Hand



You're given a five-card hand drawn from a standard 52-card deck. Each card has a rank (one of A, 2, 3, ..., 9, T, J, Q, K), and a suit (one of C, D, H, S).

The *strength* of your hand is defined as the maximum value  $k$  such that there are  $k$  cards in your hand that have the same rank.

Find the strength of your hand.

## Input

The input consists of a single line, with five two-character strings separated by spaces.

The first character in each string will be the rank of the card, and will be one of A23456789TJQK. The second character in the string will be the suit of the card, and will be one of CDHS.

It is guaranteed that all five strings are distinct.

## Output

Output, on a single line, the strength of your hand.

## Sample Input and Output

AC AD AH AS KD	4
2C 4D 4H 2D 2H	3



2018 ICPC Pacific Northwest Regional Contest

AH 2H 3H 4H 5H	1
----------------	---

PROBLEM R — LIMIT 1 SECOND  
**House Numbers**



Peter was walking down the street, and noticed that the street had houses numbered sequentially from  $m$  to  $n$ . While standing at a particular house  $x$ , he also noticed that the sum of the house numbers behind him (numbered from  $m$  to  $x - 1$ ) equaled the sum of the house numbers in front of him (numbered from  $x + 1$  to  $n$ ).

Given  $m$ , and assuming there are at least three houses total, find the lowest  $n$  such that this is possible.

**Input**

Input consists of a single line containing the integer  $m$  ( $1 \leq m \leq 1,000,000$ ).

**Output**

On a single line, print  $m$ ,  $x$ , and  $n$ , in order, separated by spaces.

It is guaranteed that there will be a solution with  $n$  less than or equal to 10,000,000.

**Sample Input and Output**

1	1 6 8
11	11 49 68
999999	999999 1317141 1571535



2018 ICPC Pacific Northwest Regional Contest

999000	999000 1000000 1000999
--------	------------------------

2018 ICPC Pacific Northwest Regional Contest

PROBLEM S — LIMIT 1 SECOND

# Greedy Scheduler



A store has  $n$  cashiers numbered sequentially from 1 to  $n$ , with  $c$  customers in a queue. A customer at the front of the queue is constantly assigned to the first unoccupied cashier, *i.e.*, cashier with the smallest number. The  $i$ th customer's shopping cart takes  $t_i$  seconds to process.

Find which cashier will process each customer's shopping cart.

## Input

The first line of input contains two space-separated integers  $n$  and  $c$  ( $1 \leq n \leq c \leq 1000$ ). The second line of input contains  $c$  space-separated integers  $t_1, \dots, t_c$ , representing the length of time required to handle that customer.

## Output

Output a single line containing  $c$  space-separated integers, each with the cashier number that handles that customer.

## Sample Input and Output

<pre>3 10 406 424 87 888 871 915 516 81 275 578</pre>	<pre>1 2 3 3 1 2 3 1 2 1</pre>
-------------------------------------------------------	--------------------------------



**icpc** International Collegiate  
Programming Contest



north america  
sponsor



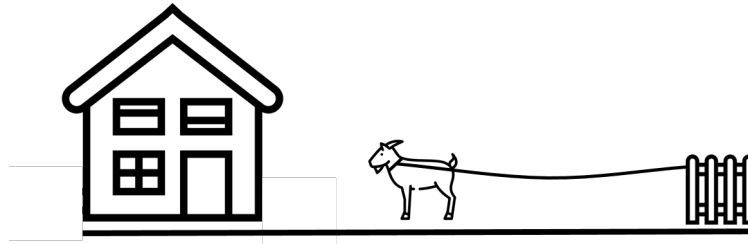
programming  
tools sponsor

2018 ICPC Pacific Northwest Regional Contest



2018 ICPC Pacific Northwest Regional Contest

PROBLEM T — LIMIT 1 SECOND  
Goat on a Rope



You have a house, which you model as an axis-aligned rectangle with corners at  $(x_1, y_1)$  and  $(x_2, y_2)$ .

You also have a goat, which you want to tie to a fence post located at  $(x, y)$ , with a rope of length  $l$ . The goat can reach anywhere within a distance  $l$  from the fence post.

Find the largest value of  $l$  so that the goat cannot reach your house.

**Input**

Input consists of a single line with six space-separated integers  $x, y, x_1, y_1, x_2,$  and  $y_2$ . All the values are guaranteed to be between  $-1000$  and  $1000$  (inclusive).

It is guaranteed that  $x_1 < y_1$  and  $x_2 < y_2$ , and that  $(x, y)$  lies strictly outside the axis-aligned rectangle with corners at  $(x_1, y_1)$  and  $(x_2, y_2)$ .

**Output**

Print, on one line, the maximum value of  $l$ , rounded and displayed to exactly three decimal places.

**Sample Input and Output**

7 4 0 0 5 4	2.000
6 0 0 2 7 6	2.000
4 8 7 8 9 9	3.000



**icpc** International Collegiate  
Programming Contest



north america  
sponsor

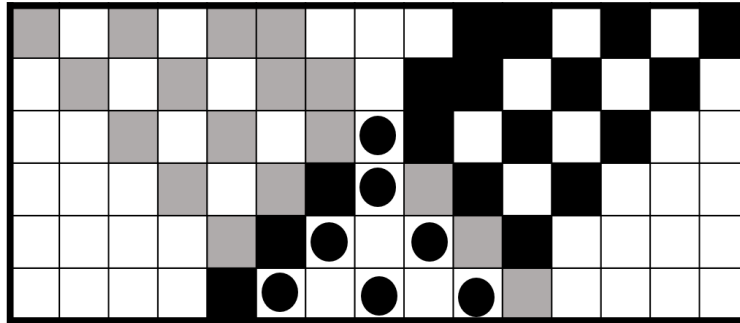


programming  
tools sponsor

2018 ICPC Pacific Northwest Regional Contest

PROBLEM U — LIMIT 1 SECOND

# Repeating Goldbachs



The Goldbach Conjecture states that any even number  $x \geq 4$  can be expressed as the sum of two primes. It can be verified that the conjecture is true for all  $x \leq 10^6$ .

Define a *Goldbach step* as taking  $x$  ( $4 \leq x \leq 10^6$ ), finding primes  $p$  and  $q$  (with  $p \leq q$ ) that sum to  $x$ , and replacing  $x$  with  $q - p$ . If there are multiple pairs of primes which sum to  $x$ , we take the pair with the largest difference. That difference must be even and less than  $x$ . Therefore, we can repeat more Goldbach steps, until we can reach a number less than 4.

Given  $x$ , find how many Goldbach steps it takes until reaching a number less than 4.

## Input

The input will consist of a single integer  $x$  ( $4 \leq x \leq 10^6$ ).

## Output

Print, on a single line, the number of Goldbach steps it takes to reach a number less than 4.

## Sample Input and Output

20	3
30	4
40	5

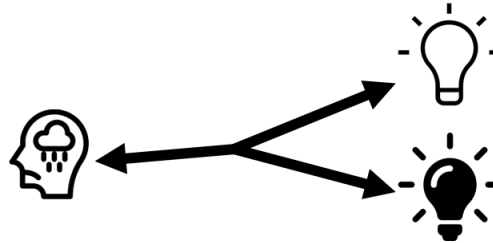


2018 ICPC Pacific Northwest Regional Contest

50	6
60	7
70	8

PROBLEM V — LIMIT 1 SECOND

# Random Index Vectors



A *Random Index Vector* (RIV) is a data structure that can represent very large arrays where most elements are zero. Here, we consider a version of RIVs which can contain only  $-1$ ,  $0$ , and  $+1$  as elements. There are three basic operations on RIVs:

- Addition of two RIVs  $a$  and  $b$ ; the resulting RIV  $c$  is defined by  $c_i = a_i + b_i$ . The values are clamped at  $\pm 1$ , *i.e.*, we define  $1 + 1 = 1$  and  $(-1) + (-1) = -1$ .
- Multiplication of two RIVs  $a$  and  $b$ ; the resulting RIV  $d$  is defined by  $d_i = a_i b_i$ .
- Rotation of an RIV  $a$  by some integer  $k$ ; this shifts all of the values in  $a$  to the left by  $k$  indices. The first  $k$  values at the start of  $a$  go to the end of the array and become the rightmost values.

An RIV is written in its *condensed form*. This representation is a list that starts with the number of nonzero values, followed by a sorted list of indices (1-indexed) that have nonzero values, where the indices are negated if the values there are  $-1$ .

For example, consider an RIV representing an array

$$(1, 0, -1, 0, 0, 0, -1, 0, 0, 1, 0).$$

There are 4 nonzero elements at indices 1, 3, 7 and 10, and the values at 3 and 7 are  $-1$ , so the condensed form of this RIV is

$$(4, 1, -3, -7, 10).$$

Given two RIVs in condensed form, add them, multiply them, and rotate them both. Output the results in condensed form.

## Input

The first line of input contains two space-separated integers  $n$  and  $k$  ( $1 \leq k \leq n \leq 10^{18}$ ), where  $n$  is the full (uncondensed) length of the RIVs and  $k$  is the number of indices to rotate by.

Each of the next two lines contains a condensed form of an RIV, starting with an integer  $m$  ( $0 \leq m \leq 1,000$ ), followed by  $m$  space-separated indices  $i_1, \dots, i_m$ . Each index  $i_j$  is a nonzero integer between  $-n$  and  $n$  (inclusive).

2018 ICPC Pacific Northwest Regional Contest

## Output

Output four vectors, one per line, in condensed form:

- Sum of the two input RIVs.
- Product of the two input RIVs.
- First RIV rotated by  $k$ .
- Second RIV rotated by  $k$ .

## Sample Input and Output

<pre>30 13 6 6 -9 -13 18 22 26 8 -1 3 7 11 13 19 20 -27</pre>	<pre>12 -1 3 6 7 -9 11 18 19 20 22 26 -27 1 -13 6 5 9 13 23 -26 -30 8 6 7 -14 -18 20 24 28 30</pre>
<pre>20 4 9 -2 -4 -8 -11 -12 15 18 19 20 7 4 5 -10 11 15 18 -20</pre>	<pre>8 -2 5 -8 -10 -12 15 18 19 5 -4 -11 15 18 -20 9 -4 -7 -8 11 14 15 16 -18 -20 7 1 -6 7 11 14 -16 20</pre>

2018 ICPC Pacific Northwest Regional Contest

PROBLEM W — LIMIT 1 SECOND

# Time Limits



A contest setter wants to determine the time limits for a given problem. There are  $n$  model solutions, and solution  $k$  takes  $t_k$  milliseconds to run on the test data. The contest setter wants the time limit to be an integer number of seconds, and wants the time limit to be at least  $s$  times larger than the slowest model solution. Compute the minimum time limit the contest setter can set.

## Input

The first line of input contains two space-separated integers  $n$  and  $s$  ( $1 \leq n \leq 100$  and  $1 \leq s \leq 20$ ).

The second line of input contains  $n$  space-separated integers  $t_1, \dots, t_n$  ( $1 \leq t_k \leq 2000$  for all  $k = 1, \dots, n$ ).

## Output

Print, on one line, the minimum time limit (in seconds) as a single integer.

## Sample Input and Output

2 5 200 250	2
3 4 47 1032 1107	5



**icpc** International Collegiate  
Programming Contest



north america  
sponsor



programming  
tools sponsor

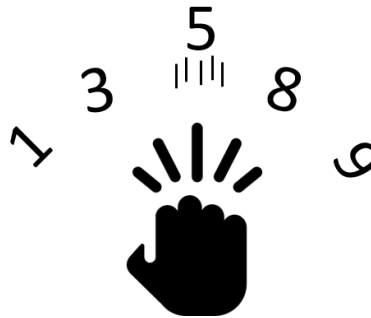
2018 ICPC Pacific Northwest Regional Contest



## 2018 ICPC Pacific Northwest Regional Contest

PROBLEM X — LIMIT 2 SECONDS

# Knockout



The solitaire game Knockout is played as follows. The digits from 1 to 9 are written down on a board, in order. In each turn, you throw a pair of six-sided dice. You sum the dice and cross out some set of digits that sum to the same total. If you cannot, the game ends and your score is the concatenation of the remaining digits. Otherwise, you throw the dice again and continue.

This game can be played to either minimize or maximize your score. Given a position of the game (what digits remain) and a roll of the dice, compute which digits you should remove and what your expected total score is for both minimizing and maximizing versions of the game.

## Input

The input contains a single line with three space-separated integers. The first is the board state, containing some nonempty subset of the digits 1 through 9, in order. The next two integers are numbers between 1 to 6 (inclusive), representing the roll of the two dice.

## Output

On the first line of output, print the digit(s) you should remove to minimize the expected score, followed by the expected score. On the second line of output, do the same for the maximizing version of Knockout.

If multiple digits are removed, list the digits in order with no space separating them. If you cannot remove digits to match the sum of the dice, print '-1' for your move instead.

The expected scores should be printed with exactly five digits after the decimal point.

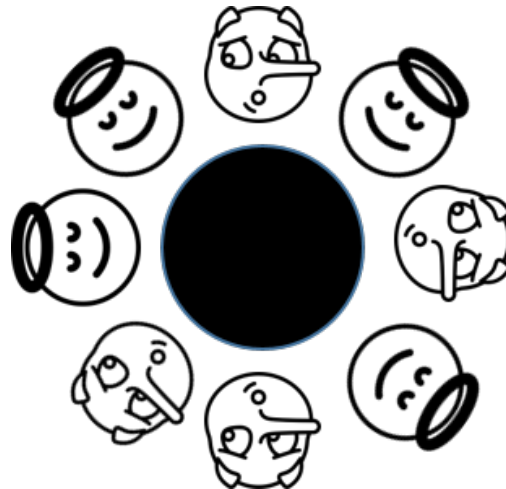
## 2018 ICPC Pacific Northwest Regional Contest

**Sample Input and Output**

1345 1 1	-1 1345.00000 -1 1345.00000
12349 3 1	13 151.70370 4 401.24546

PROBLEM Y — LIMIT 1 SECOND

# Liars



There are  $n$  people in a circle, numbered from 1 to  $n$ , each of whom always tells the truth or always lies.

Each person  $i$  makes a claim of the form: “the number of truth-tellers in this circle is between  $a_i$  and  $b_i$ , inclusive.”

Compute the maximum number of people who could be telling the truth.

## Input

The first line contains a single integer  $n$  ( $1 \leq n \leq 10^3$ ). Each of the next  $n$  lines contains two space-separated integers  $a_i$  and  $b_i$  ( $0 \leq a_i \leq b_i \leq n$ ).

## Output

Print, on a single line, the maximum number of people who could be telling the truth. If the given set of statements is inconsistent, print  $-1$  instead.

2018 ICPC Pacific Northwest Regional Contest

Sample Input and Output

3 1 1 2 3 2 2	2
------------------------	---

8 0 1 1 7 4 8 3 7 1 2 4 5 3 7 1 8	-1
-----------------------------------------------------------	----

## 2018 ICPC Pacific Northwest Regional Contest

PROBLEM Z — LIMIT 1 SECOND

## Pizza Deal



There's a pizza store which serves pizza in two sizes: either a pizza slice, with area  $A_1$  and price  $P_1$ , or a circular pizza, with radius  $R_1$  and price  $P_2$ .

You want to maximize the amount of pizza you get per dollar. Should you pick the pizza slice or the whole pizza?

**Input**

The first line contains two space-separated integers  $A_1$  and  $P_1$ .

Similarly, the second line contains two space-separated integers  $R_1$  and  $P_2$ .

It is guaranteed that all values are positive integers at most  $10^3$ . We furthermore guarantee that the two will not be worth the same amount of pizza per dollar.

**Output**

If the better deal is the whole pizza, print 'Whole pizza' on a single line.

If it is a slice of pizza, print 'Slice of pizza' on a single line.

**Sample Input and Output**

8 4 7 9	Whole pizza
9 2 4 7	Whole pizza



2018 ICPC Pacific Northwest Regional Contest

841 108 8 606	Slice of pizza
------------------	----------------