CSCI 1300
Programming Assignment 3

Collaboration

You are allowed to work with a one other student on this assignment or you may work alone. While working you can submit problems to either (or both) students account. So we know which problems to grade, submit a readme file that indicates which version of each problem to grade.

Overview

This programming contest will involved solving 5 different challenge problems. All of these are modeled after ICPC programming contest problems. However, they are setup a little differently.

Feedback

Each problem will have a feedback system enabled. This system will give you feedback you solution to each problem. If you have done the problem correctly it will tell you so. In many cases you will given a hint or suggestion to improve your program.

Problem 1: Gnome sequencing

In the book All Creatures of Mythology, gnomes are kind, bearded creatures, while goblins tend to be bossy and simple-minded. The goblins like to harass the gnomes by making them line up in groups, ordered by the length of their beards. The gnomes, being of different physical heights, vary the size of the groups and their arrangements to confuse the goblins. Therefore, the goblins must actually measure the beards in centimeters to see if everyone is lined up in order.

Your task is to write a program to assist the goblins in determining whether or not the gnomes are lined up properly from shortest beard length to longest.

Input The beard lengths of a group of gnomes all appear in a single line separated by spaces.

Output Should be “Ordered” if they are standing in order of increasing beard length and “Unordered” otherwise. There should be no other output.

Example sessions

```
80 62 77
Unordered
```

```
50 75 100 200
Ordered
```

```
20 25 30 35 40 50 45
Unordered
```

Problem 2: Road Trip

Bill and Ted are taking a road trip. But the odometer in their car is broken, so they don't know how many miles they have driven. Fortunately, Bill has a working stopwatch, so they can record their speed and the total time they have driven. Unfortunately, their record keeping strategy is a little odd, so they need help computing the total distance driven. You are to write a program to do this computation.

For example, if their log shows

Speed in miles per hour	Total elapsed time in hours
20	2
30	6
10	7

this means they drove 2 hours at 20 miles per hour, then 6-2=4 hours at 30 miles per hour, then 7-6=1 hour at 10 miles per hour. The distance driven is then (2)(20) + (4)(30) + (1)(10) = 40 + 120 + 10 = 170 miles. Note that the total elapsed time is always since the beginning of the trip, not since the previous entry in their log.

Input Each line of the input will consist of two values: the speed (in miles per hour) and a timestamp (in hours). The input will end with a line that says "Done". Note all of the speeds and times will be positive integers.

Output The output should the the total distance driven with units, for example `180 miles`

Example sessions

```
20 2
30 6
10 7
Done
170 miles
```

```
45 2
70 6
60 7
55 10
Done
595 miles
```

Problem 3

A sequence of numbers (consisting of at least two numbers) will be entered. You need to find the two largest are print them out. Note that there could be a tie. There is one additional rule for this problem, you cannot use the sort method.

Input A single line will contain all of the numbers.

Output The output should be the two largest numbers printed in increasing order.

Example sessions

```
20 25 -10 15 70 25
25 70
```

```
45 10 -5 45
45 45
```

Problem 4

You will be given a single positive integer and perform repeated addition of its digits until the result has a single digit. The process works as follows:

Number	Digit sum
93874	$9+3+8+7+4=31$
31	$3+1=4$

So the repeated digit sum of 93874 is 4.

It turns out the the repeated digit sum of any number divisible by 9 is always 9. For example, $9^25 = 717,897,987,691,852,588,770,249$ and it's repeated digit sum can be calculated as:

Number	Digit sum
717897987691852588770249	$7+1+7+8+9+7+9+8+7+6+9+1+8$ $+5+2+5+8+8+8+7+7+0+2+4+9=144$
144	$1+4+4=9$

Input A single positive integer.

Output The repeated digit sum (which has a single digit.)

Example sessions

```
93874
4

717897987691852588770249
9
```

Problem 5: Rotating sentences

In Rotating Sentences, you are asked to rotate a series of input sentences 90 degrees clockwise. So instead of displaying the input sentences from left to right and top to bottom, your program will display them from top to bottom and left to right

Input As input to your program, you will be given a series of sentences each on a separate line. The output will end with a line saying “Done”. This line is not one of your input sentences

Output The output of the program should have the first sentence printed out vertically in the leftmost column; the last sentence of the input would subsequently end up at the rightmost column.

Example sessions

```
Rene Decartes once said,
"I think, therefore I am."
Done
R"
eI
n
et
 h
Di
en
ck
a,
r
tt
eh
se
 r
oe
nf
co
er
 e
s
aI
i
da
,m
 .
 "
```

```
Hello
Konnichiwa
Hola
Namaste
Bonjour
Privet
Gutten tag
Salaam
HKHNBPGS
eooaorua
lnlmnitl
lnaajvta
oi soeea
 c tutnm
 h er
 i    t
 w    a
 a    g
```

Note there should not be blank lines at the end.