



Fall 2017 Programming Team Kickoff

Introduction to Competitive Programming, Contest Environments, and I/O



What is Competitive Programming?

In competitive programming, a group of contestants race to produce **solutions** to a set of **problems**.

Produce as many **correct** solutions as possible, as **quickly** as possible, with as **few mistakes** as possible.

The correctness of a solution is verified by an automated **judge** against a suite of secret test cases.



What is a problem?

A problem typically consists of four parts:

- Description
- Input
- Output
- Sample Cases

A problem can be thought of as a formal specification for your solution: it describes exactly what your solution should do.

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 of three, ordered by the length of their beards. The gnomes, being of different physical heights, vary 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, either from shortest to longest beard or from longest to shortest.

Input

The input starts with line containing a single integer N , $0 < N < 30$, which is the number of groups to process. Following this are N lines, each containing three distinct positive integers less than 100.

Output

There is a title line, then one line per set of beard lengths. See the sample output for capitalization and punctuation.

Sample Case 1

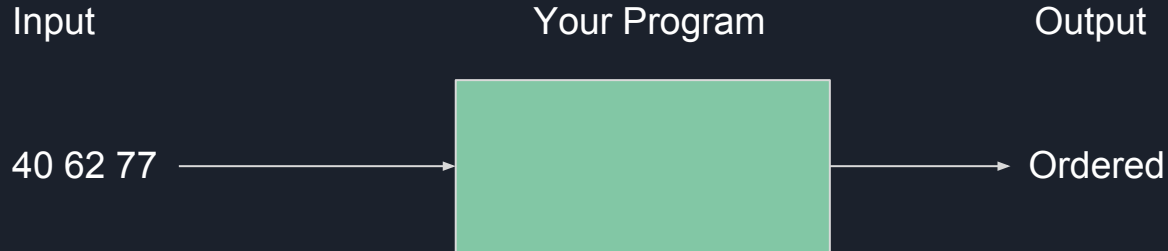
```
3
40 62 77
88 62 77
91 33 18
```

```
Gnomes :
Ordered
Unordered
Ordered
```

What is a solution?

A solution is a program that attempts to correctly meet the specifications laid out in the problem statement.

Your program must take the input from the automated tester, process it to determine the answer, and then output the correct answer to the automated tester.

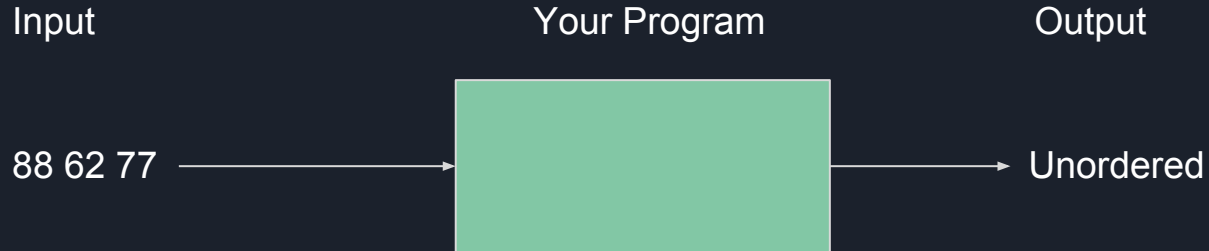




What is a solution?

A solution is a program that attempts to correctly meet the specifications laid out in the problem statement.

Your program must take the input from the automated tester, process it to determine the answer, and then output the correct answer to the automated tester.

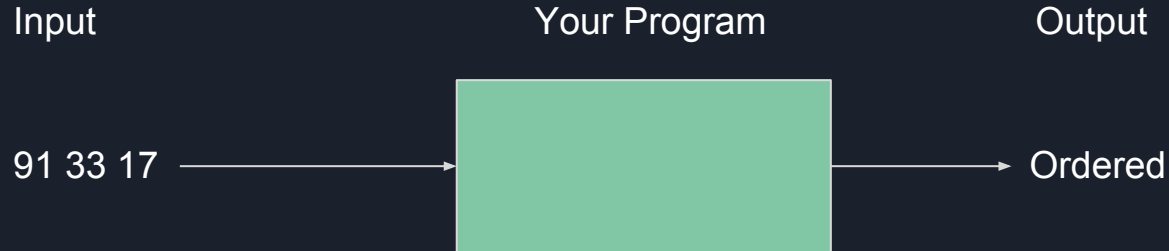




What is a solution?

A solution is a program that attempts to correctly meet the specifications laid out in the problem statement.

Your program must take the input from the automated tester, process it to determine the answer, and then output the correct answer to the automated tester.











How do we know if our solution is correct?

Once you think you have a working solution, you can submit it to the judge!

The judge will run a series of tests, where it will pass an input into your program and determine if your program produced the correct output.

Your program is only correct if it produces the correct output for every single input the judge passes in.

What do you mean by as quickly as possible with as few mistakes as possible?

	SCORE	ALPHABET 	BASESUMS 	BUGGY 	ENCLOSURE 	ILLUMINATE 	INTENTS 	ISLANDS 	PAINT 	PERIODIC 	WATER 	ZIGZAG 
UCF Badlands University of Central Florida	9 764	1/17	1	2/38	13	1/137	1/84	1/27	4/75	1/31	1/77	3/158
UCF Glacier University of Central Florida	8 817	1/7	0	4/137	1	4/246	0	1/26	1/37	1/16	1/160	1/68
UCF Pinnacle University of Central Florida	8 1004	1/14	0	3/237	0	1/293	0	1/38	1/135	1/26	1/171	1/50
UCF Acadia University of Central Florida	8 1048	1/33	0	2/194	0	1/289	0	1/25	1/60	1/12	3/256	1/119
Engineers Georgia Institute of Technology	8 1072	1/32	0	2/216	0	1/208	1	2/87	2/122	1/26	1/248	1/73
Yellow Jackets Georgia Institute of Technology	7 494	1/11	0	1/121	0	0	0	2/33	1/82	1/31	1/150	1/46
You have been eaten by a Grue University of Florida	7 605	1/8	0	1/169	0	0	0	1/29	2/87	1/24	2/126	1/122
FITtonia gigantea Florida Institute of Technology	7 713	1/13	2	3/211	0	1	0	1/50	1/73	1/40	1/247	1/39
UCF Zion University of Central Florida	7 717	1/21	0	0	0	1/232	0	2/78	2/166	1/62	1/109	1/9
FIU Gold Florida International University	6 661	1/30	2	0	0	0	0	2/46	1/128	1/21	3/281	1/95
("Bourne": "again") Auburn University	5 424	1/13	0	0	0	2	2	1/44	1/164	1/64	0	1/139
UCF Dev-2 University of Central Florida	5 611	1/20	0	0	0	0	0	1/29	5/257	2/62	1	1/143
Buzz Georgia Institute of Technology	5 699	1/85	0	0	0	0	0	1/109	4/254	1/60	2	3/91
Clemson 1 Clemson University	5 974	4/186	0	0	0	0	0	1/83	7/254	1/91	0	2/160
UCF Dev-1 University of Central Florida	4 217	1/49	0	1	0	0	0	1/45	6	1/27	0	1/96



What does this mean for strategy?

- Two identical twins participate in a programming contest.
- They both can solve problems A, B, and C, and because they are twins, they can solve every problem in the same amount of time.
 - Problem A: 10 minutes
 - Problem B: 20 minutes
 - Problem C: 30 minutes
- Since having completely identical submission times would be boring, they decide to mix up their strategy.
 - Twin 1 will solve the problems in the order A, B, and then C
 - Twin 2 will solve the problems in the order C, B, and then A.
- What will the final scoreboard look like?



Which twin will win?

- Twin 1 (A, B and then C)
 - Submission 1 - 10 minutes into the contest ($0 + 10$ minutes)
 - Submission 2 - 30 minutes into the contest ($10 + 20$ minutes)
 - Submission 3 - 60 minutes into the contest ($30 + 30$ minutes)
 - Total penalty time - $10 + 30 + 60 = 100$ minutes
- Twin 2 (C, B, and then A)
 - Submission 1 - 30 minutes into the contest ($0 + 30$ minutes)
 - Submission 2 - 50 minutes into the contest ($30 + 20$ minutes)
 - Submission 3 - 60 minutes into the contest ($50 + 10$ minutes)
 - Total penalty time - $30 + 50 + 60 = 140$ minutes
- Twin 1 wins, even though they are equally skilled programmers.



Where are the automated judges?

- auacm.com
- open.kattis.com
- codeforces.com
- topcoder.com
- code.google.com/codejam
- projecteuler.net
- hackerrank.com



How do I go about writing a solution?

A solution has 3 parts to it:

- Read the input
- Determine the answer based on that input
- Output the answer

We should also be able to write and run our solution before submitting it to make sure that it works properly.



Writing and running a solution (in Java)

- Writing a solution - write to a file. Use an IDE or text editor.
- Running a java solution - two routes
 - Use an IDE (Eclipse, IntelliJ, Netbeans, JGRASP) - The IDE has the ability to compile and run your solution.
 - Use a Text Editor - You will need to compile and run from the command line.
 - Install the Java Development Kit
 - Have a preferred shell (console, terminal, command prompt, cygwin, cmdr)



Reading Input (in Java)

First, you must import Scanner in java.util.

```
import java.util.*; // or import java.util.Scanner;
```

You can declare a scanner with the line:

```
Scanner in = new Scanner(System.in);
```

You can read input using this scanner:

```
String inputString = in.next();
```

```
int inputNumber = in.nextInt();
```



Printing Output (in Java)

You can print to standard output easily.

```
System.out.println("This is my string"); // Prints a newline at the end.
```

```
System.out.print(1); // This will print "1" without a newline at the end.
```



Your Program's Skeleton (in Java)

```
// Put imports here.
```

```
public class MyProgram { // Must be in a file called MyProgram.java
    public static void main(String[] args) {
        // This code will be executed.
    }
}
```




An example program.

```
// Save this in Echo.java
import java.util.*;

public class Echo {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        String input = in.next();
        System.out.println(input);
    }
}
```

What type of program should I write?

```
1 import java.util.*;
2
3 public class Gnomes1 {
4     public static void main(String[] args) {
5         Scanner in = new Scanner(System.in);
6         int N = in.nextInt();
7         System.out.println("Gnomes:");
8         while(N --> 0) {
9             int a = in.nextInt(), b = in.nextInt(), c = in.nextInt();
10            System.out.println((a < b) == (b < c) ? "Ordered" : "Unordered");
11        }
12    }
13 }
```

```
1 import java.util.ArrayList;
2 import java.util.List;
3 import java.util.Scanner;
4
5 public class Gnomes2 {
6
7     static class TestCase {
8         int a;
9         int b;
10        int c;
11
12        TestCase(int a, int b, int c) {
13            this.a = a;
14            this.b = b;
15            this.c = c;
16        }
17
18        boolean ordered() {
19            if (a < b && b < c) {
20                return true;
21            } else if (a > b && b > c) {
22                return true;
23            } else {
24                return false;
25            }
26        }
27    }
28
29    public static void main(String[] args) {
30        Scanner in = new Scanner(System.in);
31
32        List<TestCase> input = new ArrayList<>();
33
34        int nCases = in.nextInt();
35        for (int i = 0; i < nCases; i++) {
36            input.add(new TestCase(in.nextInt(), in.nextInt(), in.nextInt()));
37        }
38
39        System.out.println("Gnomes:");
40        for (TestCase c : input) {
41            if (c.ordered()) {
42                System.out.println("Ordered");
43            } else {
44                System.out.println("Unordered");
45            }
46        }
47    }
48 }
49 }
```



Your Turn!

Go to auacm.com

If you don't have an account, we will set you up with one!

Register for the Fall 2017 Kickoff competition

Once it starts - You can solve the only problem "Add the Numbers 1"

After that, feel free to explore the problems on auacm.com!