



Team Packet

Java Problems



Java Problems

Distance 3D ~ 10 points

Divisors ~ 20 points

Repeating Decimals ~ 30 points



Distance 3D (10 points)

Write a Java console program that calculates the distance between a pair of points on a three dimensional Cartesian coordinate system.

The distance d between two points, (x_1, y_1, z_1) and (x_2, y_2, z_2) , on a three dimensional Cartesian coordinate system is

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

Input Specification

Enter each of the six integer coordinates one perline. The first three integers represent the coordinates of one point on the coordinate system. The second three integers represent another point on the same system.

Output Specification

After entering the coordinates, output a line that says “The distance from (x,y,z) to (a,b,c) is d.ddd” where $x,y,z,a,b,$ and c stand for the actual integers input by the user and d.ddd stands for the distance between those two points with three decimal places displayed (0’s if needed) and rounded to the nearest thousandth.

Sample runs

```
Enter the first coordinate of the first point: 0
Enter the second coordinate of the first point: 0
Enter the third coordinate of the first point: 0
Enter the first coordinate of the second point: 0
Enter the second coordinate of the second point: 0
Enter the third coordinate of the second point: 0
The distance from (0,0,0) to (0,0,0) is 0.000
```

```
Enter the first coordinate of the first point: 1
Enter the second coordinate of the first point: -2
Enter the third coordinate of the first point: 3
Enter the first coordinate of the second point: 1
Enter the second coordinate of the second point: -2
Enter the third coordinate of the second point: 4
The distance from (1,-2,3) to (1,-2,4) is 1.000
```

Java ~ Problem for Newport News Computer Challenge 2004
Distance 3D ~ Page 2 of 2

Enter the first coordinate of the first point: 1
Enter the second coordinate of the first point: 2
Enter the third coordinate of the first point: 3
Enter the first coordinate of the second point: 4
Enter the second coordinate of the second point: 5
Enter the third coordinate of the second point: 6
The distance from (1,2,3) to (4,5,6) is 5.196

Enter the first coordinate of the first point: 18
Enter the second coordinate of the first point: 22
Enter the third coordinate of the first point: 72
Enter the first coordinate of the second point: -122
Enter the second coordinate of the second point: -144
Enter the third coordinate of the second point: 99
The distance from (18,22,72) to (-122,-144,99) is 218.826



Divisors (20 points)

Mathematicians love all sorts of odd properties of numbers. For instance, they consider 945 to be an interesting number, since it is the first odd number for which the sum of its divisors is larger than the number itself.

To help them search for interesting numbers, you are to write a console program that scans a range of numbers and determines the number that has the largest number of divisors in the range. Unfortunately, the size of the numbers, and the size of the range is such that a too simple-minded approach may take too much time to run. So make sure that your algorithm is clever enough to cope with the largest possible range in just a few seconds.

Input Specification

The first line of input specifies the number N of ranges, and each of the N following pairs of lines contains a range, consisting of a lower bound L and an upper bound U (L and U are each on separate lines), where L and U are included in the range. L and U are chosen such that $1 \leq L \leq U \leq 1,000,000,000$ and $0 \leq U - L \leq 10,000$.

Output Specification

For each range, find the number P which has the largest number of divisors (if several numbers tie for first place, select the lowest), and the number of positive divisors D of P (where P is included as a divisor). Print the text 'Between L and H , P has a maximum of D divisors.', where L , H , P , and D are the numbers as defined above.

Example input

```
3
1
10
1000
1000
999999900
1000000000
```

Example output

```
Between 1 and 10, 6 has a maximum of 4 divisors.
Between 1000 and 1000, 1000 has a maximum of 16 divisors.
Between 999999900 and 1000000000, 999999924 has a maximum of 192 divisors.
```



Repeating Decimals (30 points)

The decimal expansion of the fraction $1/33$ is $0.\overline{03}$, where the $\overline{03}$ is used to indicate that the cycle 03 repeats indefinitely with no intervening digits. In fact, the decimal expansion of every rational number (fraction) has a repeating cycle as opposed to decimal expansions of irrational numbers, which have no such repeating cycles.

Examples of decimal expansions of rational numbers and their repeating cycles are shown below. Here, we use parentheses to enclose the repeating cycle rather than place a bar over the cycle.

<u>fraction</u>	<u>decimal expansion</u>	<u>repeating cycle</u>	<u>cycle length</u>
1/6	0.1(6)	6	1
5/7	0.(714285)	714285	6
1/250	0.004(0)	0	1
300/31	9.(677419354838709)	766419354838709	15
655/990	0.6(61)	61	2

Write a Java GUI program that reads numerators and denominators of fractions and determines their repeating cycles.

For the purposes of this problem, define a repeating cycle of a fraction to be the first minimal length string of digits to the right of the decimal that repeats indefinitely with no intervening digits. Thus for example, the repeating cycle of the fraction $1/250$ is 0, which begins at position 4 (as opposed to 0 which begins at positions 1 or 2 and as opposed to 00 which begins at positions 1 or 4).

Input

Input consists of an integer numerator, which is nonnegative, and an integer denominator, which is positive. None of the input integers exceeds 3000.

Output

In a text field, display the decimal expansion through the first occurrence of the cycle to the right of the decimal or 50 decimal places (whichever comes first), and the length of the entire repeating cycle.

In writing the decimal expansion, enclose the repeating cycle in parentheses when possible. If the entire repeating cycle does not occur within the first 50 places, place a left parenthesis where the cycle begins - it will begin within the first 50 places - and place ``...)' ' after the 50th digit.

Sample Runs

Sample #1

The screenshot shows a Java application window titled "Repeating Decimals ~ by Charles F. Monroe". It contains the following fields and controls:

- Numerator:** A text box containing the value "76".
- Denominator:** A text box containing the value "25".
- Convert:** A button located below the denominator field.
- Decimal:** A text box displaying the result "3.04(0)".
- Num Reps:** A text box containing the value "1".

Sample #2

The screenshot shows the same application window with the following values:

- Numerator:** "5"
- Denominator:** "43"
- Convert:** Button
- Decimal:** "0.(116279069767441860465)"
- Num Reps:** "21"

Sample #3

The screenshot shows the same application window with the following values:

- Numerator:** "1"
- Denominator:** "397"
- Convert:** Button
- Decimal:** "3168765743073047858942065491183879093198992...)"
- Num Reps:** "99"

The complete value in the field labeled Decimal above is:
0.(00251889168765743073047858942065491183879093198992...)