CCC JUNIOR

Array/List, 2D Array/List

CCC 2015 Junior 1

https://cemc.math.uwaterloo.ca/contests/computing/2015/stage%201/juniorEn.pdf

- **Description**

February 18 is a special date for the CCC this year. Write a program that asks the user for a numerical month and numerical day of the month and then determines whether that date occurs before, after, or on February 18. If the date occurs before February 18, output the word Before. If the date occurs after February 18, output the word After. If the date is February 18, output the word Special.

- **Input Specification**

The input consists of two integers each on a separate line. These integers represent a date in 2015. The first line will contain the month, which will be an integer in the range from 1 (indicating January) to 12 (indicating December). The second line will contain the day of the month, which will be an integer in the range from 1 to 31. You can assume that the day of the month will be valid for the given month.

Sample Input and Output

Output Specification

Exactly one of Before, After or Special will be printed on one line.

Sample Input 1
1
7
Output for Sample Input 1
Before
Sample Input 2
8
31
Output for Sample Input 2
After

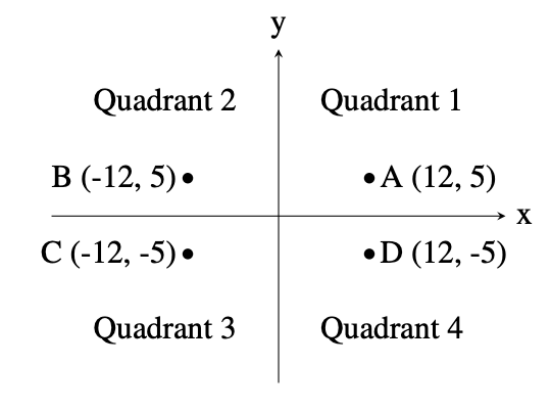
Sample Input 3
2
18
Output for Sample Input 3
Special

CCC 2017 J1

https://cemc.math.uwaterloo.ca/contests/computing/2017/stage%201/juniorEF.pdf

- Problem Description

A common problem in mathematics is to determine which quadrant a given point lies in. There are four quadrants, numbered from 1 to 4, as shown in the diagram below:



For example, the point A, which is at coordinates (12, 5) lies in quadrant 1 since both its x and y values are positive, and point B lies in quadrant 2 since its x value is negative and its y value is positive. Your job is to take a point and determine the quadrant it is in. You can assume that neither of the two coordinates will be 0.

Sample Input and Output

- **Input Specification**

The first line of input contains the integer x ($-1000 \leq x \leq 1000$; x 6= 0). The second line of input contains the integer y ($-1000 \leq y \leq 1000$; y 6= 0).

- **Output Specification**

Output the quadrant number (1, 2, 3 or 4) for the point (x, y).

Sample Input 1
12
5

Output for Sample Input 1
1

Sample Input2
9
-13

Output for Sample Input 2
4

Array

- Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.
- To declare an array, define the variable type with **square brackets**
- int[], boolean[], String[]
- Two ways to create, either by each elements in “{“ and ”}”, or pass in a size first
- Can set/get by index, can get length of an array, array.length
- Index starts with 0
- If we only care about each element rather than index, can use for loop in a simpler way
- IndexOutOfBounds exception

Array of Primitive/Non-Primitive types

- int[] a = new int[10];

  will be all 0 by default -> a[0] will be 0
- boolean[] b = new boolean[20];

  will be all false by default -> b[0] will be false
- String[] s = new String[20];

  will be null by default -> s[0] will throw NullpointerException

Loop through Array

- for (*type variable* : *arrayname*) {

  ...

  ...

  }
- for(initialization; condition;...){

  ...

  ...

  }

2D Array

- A multidimensional array is an array containing one or more arrays. To create a two-dimensional array, add each array within its own set of **curly braces.**
- int[][] myNumbers = {{1, 2, 3, 4}, {5, 6, 7}};
- myNumbers.length is 2
- myNumbers[0].length is 4
- myNumbers[1].length is 3

List

- import java.util.ArrayList;
- ArrayList<String> list = new ArrayList<String>();
- You can retrieve element by index – list.get(int index)
- You can remove element by index – list.remove(int index)
- You can get the size of a list – list.length()
- You can change the length of the list
- You can check whether the list contains an element
- You can query the list in for loop

Why list? Why Array?

- Array is a fixed size data structure while ArrayList is not. One need not to mention the size of Arraylist while creating its object. Even if we specify some initial capacity, we can add more elements.
- Array can contain both primitive data types as well as objects of a class depending on the definition of the array. However, ArrayList only supports object entries, not the primitive data types.

CCC 2016 J2 Magic Squares

https://cemc.math.uwaterloo.ca/contests/computing/2016/stage%201/juniorEn.pdf

- **Problem Description**

  Magic Squares are square arrays of numbers that have the interesting property that the numbers in each column, and in each row, all add up to the same total. Given a 4 × 4 square of numbers, determine if it is magic square.

- **Input Specification**

  The input consists of four lines, each line having 4 space-separated integers.

- **Output Specification**

  Output either magic if the input is a magic square, or not magic if the input is not a magic square.

Sample Input and Output

Sample Input 1
16 3 2 13
5 10 11 8
9 6 7 12
4 15 14 1
Output for Sample Input 1
magic

Sample Input 2
5 10 1 3
10 4 2 3
1 2 8 5
3 3 5 0
Output for Sample Input 2
not magic

CCC 2014 J4 Party Invitation

https://cemc.math.uwaterloo.ca/contests/computing/2014/stage%201/juniorEn.pdf

- **Problem Description**

You are hosting a party and do not have room to invite all of your friends. You use the following unemotional mathematical method to determine which friends to invite. Number your friends 1, 2, . . . , K and place them in a list in this order. Then perform m rounds. In each round, use a number to determine which friends to remove from the ordered list. The rounds will use numbers r1, r2, . . . , rm. In round i remove all the remaining people in positions that are multiples of ri (that is, ri , 2ri , 3ri , . . .) The beginning of the list is position 1. Output the numbers of the friends that remain after this removal process.

Input and Output Specification

- **Input Specification**
  The first line of input contains the integer K ($1 \leq K \leq 100$). The second line of input contains the integer m ($1 \leq m \leq 10$), which is the number of rounds of removal. The next m lines each contain one integer. The ith of these lines ($1 \leq i \leq m$) contains ri ($2 \leq ri \leq 100$) indicating that every person at a position which is multiple of ri should be removed.
- **Output Specification**

The output is the integers assigned to friends who were not removed. One integer is printed per line in increasing sorted order.

Sample Input and Output

Sample Input

10
2
2
3

Output for Sample Input

1
3
7
9

Explanation of Output for Sample Input

Initially, our list of invitees is 1, 2, 3, 4, 5, 6, 7, 8, 9, 10. There will be two rounds of removals. After the first round of removals, we remove the even positions (i.e., every second position), which causes our list of invitees to be 1, 3, 5, 7, 9. After the second round of removals, we remove every 3rd remaining invitee: thus, we keep 1 and 3, remove 5 and keep 7 and 9, which leaves us with an invitee list of 1, 3, 7, 9.

CCC 2018 J3 Are we there yet

https://cemc.math.uwaterloo.ca/contests/computing/2018/stage%201/juniorEF.pdf

- **Problem Description**

  You decide to go for a very long drive on a very straight road. Along this road are five cities. As you travel, you record the distance between each pair of consecutive cities. You would like to calculate a distance table that indicates the distance between any two of the cities you have encountered.

- **Input Specification**

  The first line contains 4 positive integers less than 1000, each representing the distances between consecutive pairs of consecutive cities: specifically, the ith integer represents the distance between city i and city i + 1.

- **Output Specification**

  The output should be 5 lines, with the ith line ($1 \leq i \leq 5$) containing the distance from city i to cities 1, 2, ... 5 in order, separated by one space.

Sample Input and Output

- Sample Input

  3 10 12 5

- Output for Sample Input

  0 3 13 25 30

  3 0 10 22 27

  13 10 0 12 17

  25 22 12 0 5

  30 27 17 5 0

Explanation of Output for Sample Input

The first line of output contains:

- 0, since the distance from city 1 to city 1 is 0;
- 3, since the distance between city 1 and city 2 is 3;
- 13, since the distance between city 1 and city 3 is 3 + 10 = 13;
- 25, since the distance between city 1 and city 4 is 3 + 10 + 12 = 25;
- 30, since the distance between city 1 and city 5 is 3 + 10 + 12 + 5 = 30.

CCC 2010 J4 Global Warming

https://www.cemc.uwaterloo.ca/contests/computing/2010/stage1/juniorEn.pdf

- Problem Description

Your task is to help scientists predict the trend of the global warming. One of the hypotheses they are considering is that over long periods of time, the average temperature follows certain cycles, but each time the cycle starts from a higher temperature level. The temperatures are measured over five-year averages, and expressed in tenths of a degree. For example, if the following five-year averages are observed:

3, 4, 6, 4, 5, 7, 5

then we can calculate that the temperature changes first 1 tenth up, then 2 up, then 2 down, 1 up, 2 up, and 2 down. There is a cycle of changes of length three which covers all of the temperature differences: (+1, +2, −2). In other words, if we look at the differences starting at the first position, there is a cycle of length three of the form (+1, +2, −2) followed by another cycle of length three of exactly the same form.

CCC 2010 J4 Global Warming

https://www.cemc.uwaterloo.ca/contests/computing/2010/stage1/juniorEn.pdf

- By way of another example, suppose the following average temperatures are observed:

3, 4, 6, 7.

In this case, there is a difference of one up, two up, then one up. We would consider the shortest cycle to be length two in this case: the cycle (+1, +2). Notice that this cycle occurs once, followed by one truncated occurrence of exactly the same cycle.

Your task is to find the shortest such cycle from a given sequence of temperatures.

Input and Output Specification

- Input Specification

The input consists of a number of test cases. Each test case starts with the number n ($1 \leq n \leq 20$), representing the number of temperatures in a sequence, followed by the sequence of n temperatures. You may assume that each temperature input is an integer in the range −1000...1000 inclusive. The numbers are separated by a single space. The last test case is indicated by a zero and should produce no output.

- Output Specification

For each test case, produce the length of the shortest temperature cycle. The cycle always exists, since the whole sequence could be treated as one long cycle.

Sample Input and Output

Sample Input	Output for this test case
7 3 4 6 4 5 7 5	3
3 1 3 5	1
3 1 4 5	2
4 3 4 6 7	2
0	