First Round Official Contest

FPTU Hackathon 2022

July 17, 2022

Problems

Do not open before the contest has started.

Advice, hints, and general information

- The problems are **not** sorted by difficulty.
- Your solution programs must read input from *standard input* (e.g. System.in in Java or cin in C++) and write output to *standard output* (e.g. System.out in Java or cout in C++). For further details and examples, please refer to your administrator guide and Domjudge documentation.
- For information about which compiler flags and versions are used, please refer to your administrator guide. (Python 2.7.17, Oracle Java 1.8.0_144, gcc 7.5.0 (C, C++ std14)).
- Your submissions will be run multiple times, on several different inputs. If your submission is incorrect, the error message you get will be the error exhibited on the first input on which you failed.
  - E.g., if your instance is prone to crash but also incorrect, your submission may be judged as either "Wrong Answer" or "Run Time Error", depending on which is discovered first. The inputs for a problem will always be tested in the same order.
- If you think some problem is ambiguous or underspecified, you may ask the judges for a clarification request through the Domjudge system. The most likely response is "No comment, read problem statement", indicating that the answer can be deduced by carefully reading the problem statement or by checking the sample test cases given in the problem, or that the answer to the question is simply irrelevant to solving the problem.
- In general we are lenient with small formatting errors in the output, in particular whitespace errors within reason, and upper/lower case errors are often (but not always) ignored. But not printing any spaces at all (e.g. missing the space in the string "1 2" so that it becomes "12") is typically not accepted. The safest way to get accepted is to follow the output format exactly.
- For problems with floating point output, we only require that your output is correct up to some error tolerance. For example, if the problem requires the output to be within either absolute or relative error of 10^{-4}, this means that
  - If the correct answer is 0.05, any answer between 0.0499 and .0501 will be accepted.
  - If the correct answer is 500, any answer between 499.95 and 500.05 will be accepted.

  Any reasonable format for floating point numbers is acceptable. For instance, "17.000000", "0.17e2", and "17" are all acceptable ways of formatting the number 17. For the definition of reasonable, please use your common sense.

A. Multiply of All Prime Number in a Range

Time Limit: 3 seconds

Problem description

Users are required to enter two integer variables 'M' and 'N' (note: M<=N or M>N) using standard input stream (stdin). The system displays the multiply of all prime numbers in range [M-N] (or may be [N-M]). If there are not any prime numbers in the range, system displays 1.

Input:

M, N are 2 integer numbers which is separated by space where M, N <= 50

Output:

An integer number which is the multiplication result of all prime numbers in range of M and N. in case there are not any prime numbers in the range, 1 is output.

Example 1:

Input	Output
2 4	6

Example 2:

Input	Output
34 20	20667

Example 3:

Input	Output
-4 10	210

Example 4:

Input	Output
-3 -9	1