PROBLEMS OF NUMBER THEORY IN PROGRAMMING CONTESTS

M. Falileeva, L. Shakirova, M. Kinder

Kazan Federal University (RUSSIAN FEDERATION)

Abstract

There are many different competitions in the field of informatics with different objectives. In spite of these differences, they all share the same need for high quality tasks. This paper describes the algorithmic tasks of number theory of different levels of difficulty. Most tasks are used in the specific scope of teaching and learning informatics.

In this paper we consider the problems of enumerating the number of ordered tuples of positive integers with fixed greatest common divisor and least common multiple, and we analyze the some properties of the resulting arithmetic functions. The important feature of these tasks is that they are multilevel tasks. They assume to use solution algorithms of various complexity which depend on dimension of the task. All algorithms we present have low sample complexity that depends only on the input parameters.

Many of the examples in this paper are taken from the Open Cup named after E.V. Pankratiev (Grand-Prix of Tatarstan). Full texts for all of these problems are available on the Internet: http://codeforces.com/gym/100942?locale=en . We hope that some classes of such tasks would enlarge scope of tasks for use in programming contests at various levels.

Keywords: programming contests, olympiads of informatics, tasks of number theory, training.

1 INTRODUCTION

Each year a wealth of informatics olympiads are held worldwide at national, regional and international levels, all of which require fairly engaging and challenging tasks. Usually, programming competitions require students to submit programs which are then run through a variety of test scenarios and judged accordingly. The difficulty, however, lays not so much in the programming but rather the design of the underlying algorithms [2]. In most cases, these contests are based on automatic rating of the submitted solutions. This is accomplished by running them on batches of input data and testing correctness of the output.

The first programming contest hosted in Kazan, the capital of Tatarstan (Russia), in 2000. The 16th international programming contest will be hold in Kazan in 2016. The conditions of contests are as close as possible to the ACM ICPC conditions: the participating team, consisting of three people, has only one computer and five hours to solve 10 to 12 fairly difficult problems. The successful solution of problems requires actual knowledge of programming languages, mathematical training, knowledge of algorithms and data structures, and skill to work in team.

Tatarstan students are often become winners of the final stage of the All-Russian Olympiad for School Students in Informatics. In 2015, the students of Kazan Federal University are reached the final of the prestigious World Championship in Collegiate Programming Contest (Association for Computing Machinery International Collegiate Programming Contest). This competition consists of several stages. In order to reach the final, it is necessary to show the best results in the regional and subregional stages of the competition, which, this year, involved about 13 thousand teams from almost three thousand universities in 90 countries.

Every year, teams for sports programming attend summer and winter camp, organized specially for them in Petrozavodsk and Izhevsk, where, besides Russian programmers, teams from Poland, the Czech Republic, Japan, Ukraine and Belarus gather together. In addition, a dozen of open championships are held during the year, where it is possible to meet potential ACM ICPC finalists. Open All-Siberian Programming Contest, Open Championship of the Republic of Tatarstan – Tournament of ICL (International Computers Limited) [8] and Ural Championship are among these events.

ISBN: 978-84-608-8860-4

2 CONTEST PROBLEMS OF TOURNAMENT OF ICL

A vast majority of problems appearing in programming contests are mathematical or logical in nature. Typical such tasks belong to one of the following categories: combinatorics, number theory, graph theory, geometry, string analysis and data structures.

Ideas for tasks can come from many different disciplines; however some branches of mathematics are particularly useful for creating programming problems. In the article [5] we discussed the features of combinatorial problems on programming competitions. In this section we will focus on tasks derived from the mathematical field of number theory.

These problems can be used for teaching innovative courses in algorithmic number theory and programming, and in training for differential competition. Consider the following problem.

THE SMALLEST FRACTION [Open Cup named after E.V. Pankratiev, 2016, author – M. Kinder.]

Three and a half sheep

And eight hundredths of a shepherd

Once met by a river

With four-fifths of a rooster.

M. Weitzman 'Actions with fractions'.

Currently the world knows a great number of sensational archaeological discoveries. Data storage devices relating to the period of programming of the XX century were discovered during recent excavations. Decryption of files allowed scientists to prove the hypothesis that ancient programmers were able to make simple arithmetic operations with fractions. Many texts have been deciphered, many mysteries have been solved. However one problem has remained unsolved: a calculation of the smallest positive fraction that, if divided by each of n given fractions a_i / b_i, gives an integer number.

May be you will manage to solve it...

Here a_i and b_i are numerator and denominator of irreducible fraction: $1 \le a_i \le 10^3$, $1 \le b_i \le 10^9$; $1 \le i \le n$, $1 \le n \le 6$. (Output two positive integers that are numerator and denominator of the smallest irreducible fraction satisfying the condition of the problem.)

This is a simple task, so almost all the participants of the competition have solved this problem. Let x / y is the desired irreducible fraction. (A fraction x / y is irreducible if and only if x and y are coprime, that is, if x and y have a greatest common divisor of 1.) It is not difficult to guess that

$$x = \text{lcm}(a_1, a_2, \ldots, a_n), \quad y = \text{gcd}(a_1, a_2, \ldots, a_n),$$

i.e., a number x is the lower common multiple of all numbers a_i, a number y is the greatest common divisor of all numbers b_i. In particular, if all the numbers a_i are coprime, then $x = a_1 \cdot a_2 \cdot \ldots \cdot a_n$. Since the numbers a_i are less than 10^3, we see that the number n less than 10^{18}.

The following task is also related to the greatest common divisor and the lower common multiple.

GCD AND LCM [Open Cup named after E.V. Pankratiev, 2016, author – M. Kinder.]

It is not difficult to find out the greatest common divisor and the least common multiple of several integers. There is no doubt that you have solved such problems. But have you ever tried to find the integers by their greatest common divisor (GCD) and least common multiple (LCM)? Or, at least, have you determined how many of such sets do GCD and LCM have? Probably, you haven't ...

Find out how many ordered sets of k positive integers which greatest common divisor and the least common multiple are equal to d and m respectively. For example, for $k = 2$, $d = 2$, $m = 12$, there are four described sets: (2,12)*,* (12,2)*,* (4,6) *and* (6,4)*.*

Here are $2 \le k \le 10^{18}$, $1 \le d \le m \le 10^9$.

(Output the number of ordered sets by modulo ($10^9 + 9$).)

Foremost, the least common multiple is always divisible by the greatest common divisor. So if m is not divisible by d then $k = 0$, i.e. the number of ordered sets is equal to zero. If m is divisible by d then all the numbers of each tuple are divisible by d. Then the problem reduces to the case $d = 1$ and $M = m / d$. If M has the factorization $M = p_1^{\alpha_1} p_2^{\alpha_2} \ldots p_r^{\alpha_r}$, the number of ordered k-tuples whose gcd is 1 and lcm is M, is given by the formula [1]

$$L(M,k) = \prod_{i=1}^{r} \left[(\alpha_i + 1)^k - 2\alpha_i^k + (\alpha_i - 1)^k \right].$$

For instance, *for* $k = 2$, $M = 12 = 2^2 \cdot 3^1$ there are $L(M,2) = [(2 + 1)^2 - 2 \cdot 2^2 + (2 - 1)^2] \cdot [(1 + 1)^2 - 2 \cdot 1^2 + (1 - 1)^2] = 4$ ordered 2-tuples. In order to calculate the *k*-th degree of the numbers we should to apply the binary exponentiation. This leads to a solution running in time $O(Fact(M) + \log k)$.

The online encyclopedia of integer sequences [9] is full of interesting theoretical and numerical sequences and recurrence relations; simply browsing through the encyclopedia can yield interesting results.

BEAUTIFUL SUMS [Open Cup named after E.V. Pankratiev, 2015, author – M. Kinder.]

Beautiful sums are the sums of several consequent positive integers. For example, the sums 7 + 8 *and* 4 + 5 + 6 *are beautiful, and the sum* 3 + 5 + 7 *is not beautiful even though the value in all cases equals* 15*. (The sum of single summand* 15 *is also considered beautiful.)*

Given this, the beauty index of integer is the number of its representations as a beautiful sum. For example, the beauty index of number 15 *equals* 4 *as* 15 *is represented by a beautiful sum in four ways:* 15 = 7 + 8 = 4 + 5 + 6 = 1 + 2 + 3 + 4 + 5*.*

One number is more beautiful than another if its beauty index is higher. If numbers have equal beauty indexes the smaller one is considered more beautiful. For example, 15 *is the smallest integer having beauty index* 4*.*

You have to find the smallest integer for given beauty index n $(1 \le n \le 10^5)$*.*

(Output the smallest integer for given beauty index *n* by modulo $(10^9 + 9)$.)

The problem of representing numbers as sums of consecutive integers and of counting the number of representations of this type has been studied by many authors ([3], [4], [6], [7]). The origin of this question is unknown, but one can easily believe that it is part of the mathematical folklore. Suppose *N* is the sum of consecutive positive integers, then it is possible to write

$$N = a + (a + 1) + \ldots + (a + k) = \frac{1}{2}(k + 1)(2a + k),\ k \ge 0.$$

The sum of two numbers $k + 1$ and $2a + k$ is an odd number $2(a + k) + 1$. Hence, the numbers $k + 1$ and $2a + k$ is opposite parity, i.e., one is odd and one is even. That is, the number of partitions of *N* into (one or more) consecutive parts is equal to the number of odd divisors of *N*. For instance, 15 has four partitions into consecutive parts, and 15 has four odd divisors, 1, 3, 5 and 15. An easy way of calculating the beauty index of a positive number is that of decomposing the number into its prime factors, taking the powers of all prime factors greater than 2, adding 1 to all of them, multiplying the numbers thus obtained with each other. For instance, 90 has the beauty index 6 because $90 = 2 \cdot 3^2 \cdot 5^1$; the powers of 3 and 5 are respectively 2 and 1, and applying this method $(2 + 1) \cdot (1 + 1) = 6$. Notice that if $N = 2^{\alpha_1} 3^{\alpha_2} 5^{\alpha_3} \ldots$ is the smallest integer such that the beauty index is equal of *n* then

$$(\alpha_2 + 1)(\alpha_3 + 1) \ldots = n.$$

This unique set of numbers $\alpha_i + 1$ multiplying to n is called the factorization of *n.* Thus, among all possible factorizations of number *n*, we find an expansion of n such that an appropriate number of *N* was the lowest possible. (See also [10].)

The full analysis of the solution is also available on web site of Tournament ICL [8].

3 CONCLUSION

Creating high quality tasks is a difficult and time-consuming process. We endeavour to make tasks interesting, understandable and accessible.

In this paper we introduced our approach to the development of algorithmic thinking within the subject number theoretic Algorithms and three tasks of different level of difficulty contributing to the development of students algorithmic thinking and their imagination. The paper is intended as an inspiration for all educators developing students programming skills.

In this paper we have considered the problems of enumerating the number of ordered tuples of positive integers with fixed greatest common divisor and least common multiple, and we analyze the some properties of the resulting arithmetic functions. The important feature of these tasks is that they are multilevel tasks. They assume to use solution algorithms of various complexity which depend on dimension of the task. All algorithms we present have low sample complexity that depends only on the input parameters.

Many of the examples in this paper are taken from the Open Cup named after E.V. Pankratiev (Grand-Prix of Tatarstan). Full texts for all of these problems are available on the Internet [8]. We hope that some classes of such tasks would enlarge scope of tasks for use in programming contests at various levels.

REFERENCES

[1] Bagdasar, O. (2014). On Some Functions Involving the lcm and gcd of Integer Tuples. Scientific Publications of the State University of Novi Pazar, Appl. Maths. Inform. and Mech., vol. 6, 2, pp. 91-100.

[2] Burton, B. (2008). Informatics olympiads: challenges in programming and algorithm design. In: Dobbie, G. and Mans, B. (Eds.) Proceedings of Thirty-First Australasian Computer Science Conference (ACSC 2008), Wollongong, NSW, Australia. CRPIT, 74, 9–13.

[3] Graham, R.; Knuth, D.; Patashnik, O. (1994). Concrete Mathematics, Addison-Wesley. (Problem 2.30.)

[4] Hirschhorn, M.; Hirschhorn, P. (2005). Partitions into consecutive parts. Mathematics Magazine, 78, pp. 396–397.

[5] Kinder, M.; Falileeva, M.; Shakirova, L. (2016). Combinatorial problems of enumeration in programming contests. INTED2016 Proceedings. 10th International Technology, Education and Development Conference, Valencia, Spain, pp. 0510-0513.

[6] Nyblom, M. (2001). On the representation of the integers as a difference of nonconsecutive triangular numbers. Fibonacci Quarterly 39:3, pp. 256-263.

[7] Verhoeff, T. (1999). Rectangular and Trapezoidal Arrangements. Journal of Integer Sequences, vol. 2, 99.1.6.

[8] www.icl.ru/turnir

[9] http://oeis.org

[10] http://oeis.org/A038547