Olympiads in Informatics, 2011, Vol. 5, 113–119

Tasks of "Mission Impossible" and "Mission Impeded" Types

Pavel S. PANKOV[1], Kirill A. BARYSHNIKOV[2]
[1]*International University of Kyrgyzstan*
A. Sydykov str. 252, apt. 10, 720001 Bishkek, Kyrgyzstan
[2]*OJSC "Finance Credit Bank", Bishkek, Kyrgyzstan*
e-mail: pps50@rambler.ru, kiryakg@gmail.com

Abstract. We propose a way to derive new tasks by reversing the goals of existing tasks as follows. Some operation is possible now and we can perform it in some number of steps. We would like to make this operation impossible (or to increase the number of steps to complete it). What is the minimum number of steps we should do to achieve our goal? Some tasks of the proposed types and ways to create such tasks in various branches of informatics are presented in the paper.

Key words: olympiads in informatics, tasks, impossibility, impediment.

1. Survey of Types of Tasks and the Aim of Paper

There are number of publications on the topic of creating tasks for informatics olympiads, for example Kemkes *et al.* (2007), Diks *et al.* (2008), Pankov (2008), Pankov and Baryshnikov (2009), Pankov *et al.* (2010).

We cite Burton *et al.* (2008) "The question might involve finding a specific object (or set of objects) that has some property, or that maximizes or minimizes some property. It might involve aggregating some attributes over a set of objects, or setting the attributes of each object to reach a particular condition."

Thus, most of tasks given earlier may be classified as follows:

- TA) to detect whether the object exists (or the operation is possible; the aim is attainable) (call them **alternative**);
- TC) to find the number of cases (**combinatory**);
- TO) to find the extreme value, the minimum number of steps (**optimization**);
- TB) to build the object.

All these goals may be called positive. We propose to build tasks with negative goals using environments of known types of tasks. We shall call such positive tasks basic (for reversing).

We propose the following pendants to the types of TA), TC), TO):

- TA-A) to detect whether the possible operation can be made impossible (is it possible to make the existing solution impossible in exactly N steps? If yes, then provide these steps); and the optimization tasks:

TA-L) to find the minimum number of steps to make the possible operation impossible;

TC-L) to find the minimum number of steps to make the number of cases less than a given number;

TC-O) to make the number of cases the least possible by means of the given number of steps;

TO-L) to find the minimum number of steps to make the extreme value worse than a given number (boundary);

TO-O) to make the extreme value the worst possible by means of the given number of steps.

We are going to demonstrate that many well-known tasks can be transformed in such a way. We also consider general tasks (G-Tasks), i.e., such schemes that can generate concrete tasks. If the above-listed "negative" aim is already attained then the output is zero.

We classify tasks by the traditional subject fields: text processing, graphs, rectangular geometry and miscellaneous tasks. A general proposal on subject fields was presented in Verhoeff *et al.* (2006), surveys of actual subject fields were given in (2008), Verhoeff (2009).

We simplify the texts of tasks to the phrases beginning with: "Can . . . be made . . .", "Output . . .", "How many . . ." instead of the formal "Write a program finding/detecting . . .". Restrictions on data also are not specified. They are to be put when the simplest (brute force) algorithm and effective ones for each task are built. Comments are given in square brackets. All numbers given in tasks are supposed to be integer and greater than zero.

2. Tasks on Text Processing

Task 1 (TA-L). (The Kyrgyzstan quarterfinal of the International Collegiate Programming Contest administered by the ACM, October 2010). Given a word W of 4..100 capital letters. How many, at least, letters must me erased from W in such a way that the word 'SU' cannot be obtained from the rest of W by means of further erasing letters?

We shall call such relation as "scattered embedding" or "S-embedding" of 'SU' into W.

Example 1. Input: UKRS. Output: 0.

Example 2. Input: BSSSSKKRRSSUU. Output: 2.

Solution

If $(\textit{not}(S \textit{ in } W))$ *or* $(\textit{not}(U \textit{ in } W))$ *then Output* 0 *else*

$\{M := \min\{\textit{number of } S \textit{ in } W;\ \textit{number of } U \textit{ in } W\};$

for all clearances C *in* W $\{M1 := (\textit{number of } S \textit{ in } W \textit{ left to } C)$

$+(\textit{number of } U \textit{ in } W \textit{ right to } C);$

$M := \min\{M, M1\}\};$ *Output* $M\}.$

Remark. Let $N := \mathrm{length}(W)$. If $M1$ is counted directly then the complexity is $O(N^2)$; if preceding values of numbers are used then the complexity is $O(N)$. This complexity can put restrictions of the length of the given word W.

A more general version can be defined as follows.

Task 2. (TA-L). Given words W (of length N) and $W1$ (of length M). How many, at least, letters must be erased from W in such a way that (*) $W1$ cannot be S-embedded into the rest of W?

Under the additional assumption, that all letters in $W1$ are distinct (Task 2A), U. Degenbaev proposed the following

Solution (dynamical programming)

Denote the solution for words $W[1..i]$ *and* $W1[1..j]$ *correspondingly as* $D[i, j]$.

for all i $D[i, 0] := i$;
for all $j > 0$ $D[0, j] = 0$;
for all $i > 0,\ j > 0$ { *if not (letters* $W[i] = W1[j]$) *then* $D[i, j] := D[i-1, j]$
else $D[i, j] := \min\{D[i-1, j] + 1,\ D[i-1, j-1]\}\}$;
Output $D[N, M]$.

Other types of tasks in the environment of Task 2 – Given words W (of length N) and $W1$ (of length M).

Task 3 (TA-A). Can the goal (*) be attained by erasing the given number K of letters from W?

G-Task 4 (TC-L). How many, at least, letters must be erased from W in such a way that the number of S-embeddings of $W1$ becomes less than the given number K?

G-Task 5 (TC-O). Find the minimum possible number of S-embeddings of $W1$ after a given number Q letters have been erased from W.

Certainly, these G-Tasks are too complicated for vast initial data. To derive interesting tasks from them, additional conditions are to be put.

Task 6 (classical, as basic). Given the set S of words and the word W. Can W be composed of a subset of S under the condition A) without overlapping or B) with possible overlapping?

Other types of tasks in the environment of Task 6 – Given the set S of words and the word W:

Task 7 (TA-A). Can K words be removed from S to make such composing impossible?

Task 8 (TA-L). How many, at least, words must be removed from S to make such composing impossible?

If all words are made of a same (one) letter then we obtain:

Task 9 (classical, as basic). Given a set S of numbers and a number N. Can N be presented as the sum of a subset of S?

Other types of tasks in the environment of Task 9 – Given the set S of numbers and the number N:

Task 10 (TA-A). Can K numbers be removed from S to make such presentation impossible?

Task 11 (TA-L). How many, at least, numbers be removed from S to make such presentation impossible?

3. Tasks on Graphs

We recall two classical tasks and give adjacent tasks due to the schemes of Section 1.

Task 12 (TA-L) (classical, negative in our terminology). Given a connected graph and two of its vertices (which are not subject for removal). How many, at least, A) arcs or B) vertices are to be removed to disconnect these two vertices?

Other types of tasks in the environment of Task 12:

Task 13 (TO-L) ... to make the distance between these vertices greater than a given number?

Task 14 (TO-O) ... to make the distance between these vertices as large as possible?

Task 15 (TC-L) ... to make the number of paths implementing the distance between these vertices less than a given number?

Task 16 (TA-L) (classical). How many, at least, A) arcs or B) vertices are to be removed to make a given connected graph non-connected?

Other types of tasks in the environment of Task 16:

Task 17 (TA-A). Can we remove the given number of A) arcs or B) vertices to make these vertices disconnected?

Task 18 (TO-L) ... to split a given graph into a given number of independent graphs?

G-Task 19 (TO-L) ... to make a) the diameter of the graph or b) the radius of the graph greater than a given number whilst preserving connectedness?

G-Task 20 (TO-O) ... to make a) the diameter of the graph or b) the radius of the graph as large as possible whilst preserving connectedness?

G-Task 21 (TC-L) ... to make the number of paths implementing a) the diameter of the graph or b) the radius of the graph less than the given number whilst preserving connectedness?

Demonstrate pendants to a certain task:

Task 22 (as basic), Pankov, 2008. Given a graph (its arcs are of length 1) and two of its neighbor vertices. The train has the length 1, its head is at the 1st vertex, and its tail is at the 2nd vertex. The train can move ahead only. Output the length of the shortest way to be passed by the train such that its head and tail would swap.

Types of pendant tasks in the environment of Task 22:

Task 23 (TA-L). How many, at least, A) arcs or B) vertices are to be removed to make such swap impossible?

Task 24 (TO-L) . . . to make the length of such a way greater than a given number?

Task 25 (TO-O) . . . to make the length of such a way as large as possible?

4. Tasks on Rectangular Geometry

We shall consider squared paper and call points with integer coordinates "integer points".

Demonstrate pendants to two tasks.

Task 26 (as basic). Given a number $N > 1$ and a list of integer points (obstacles) which are not within the square Q: $0 <= X <= N,\ 0 <= Y <= N$. Can Q go away (far from all obstacles) by shifts parallel to coordinate axes?

Example. Input: $N = 5$, four integer points: (0, 4); (5, 1); (5, 3); (6, 9). Output: It can.

Types of pendant tasks in the environment of Task 26:

Task 27 (TA-L). . . . How many, at least, integer points (additional obstacles) are to be added to make the going away of Q impossible?

The same Example. Output: 2 points.

Task 28 (TA-L) . . . How many, at least, integer points are to be added to make the number of shifts for the going away of Q greater than a given natural number K?

The same Example; Also input: $K = 2$. Output: 3 points.

Task 29 (as basic), Pankov, 2008. The train stands on the segment $(0, N)(\textit{Head}) - (0, 0)(\textit{Tail})$. The train can move (forward only) along edges of the rectangular grid not self-touching and cannot pass given integer points (obstacles) $(X1, Y1),\ (X2, Y2), \ldots,$ (XK, YK). Find the length of the shortest ways to be passed by the train in order to reach the state $(XH, YH)(\text{Head}) - (XT, YT)(\text{Tail})$. Main restriction for these given points: $|XH - XT| + |YH - YT| \leqslant N$.

Types of pendant tasks in the environment of Task 29.

Task 30 (TA-L). How many, at least, obstacles (not on the initial segment) are to be added to make such motion impossible?

Task 31 (TO-L) . . . to make the length of such a way greater than a given number?

5. Miscellaneous Tasks

G-Task 32 (TA-L). Given a set U (a rectangle, a segment etc.) and a covering C of it. How many, at least, elements, must be removed from C to make it not a covering of U?

Task 33 (as basic, Pankov, 2010). There are M known chemicals; the first N of them are present.

Some of the chemicals can be obtained from other ones (we will only consider reactions that cause two chemicals to become one). All reactions are given as four numbers $B1,\ B2,\ A$ (all different natural numbers in $1..M$) and T (integer number denoting releasing heat, if $T > 0$, or required heat, if $T < 0$) indicating the Ath chemical is obtained of $B1$th and $B2$th ones. Find the most profitable way to obtain the given $A0$th chemical of the list $M-N+1..M$ (if it is possible), i.e., such sequence of reactions $B1,\ B2,\ A,\ T$ that:

- all As are in $M - N + 1..M$ and different; the last A is $A0$;
- each $B1,\ B2$ are either in $1..N$ or of preceding As;
- all As except $A0$ are used in following reactions;
- the sum of all Ts has the greatest possible value.

A pendant task in the environment of Task 33.

Task 34 (TA-A). How many, at least, A) present chemicals of $1..N$ or B) permitted reactions must be excluded to make obtaining the $A0$th chemical impossible?

Task 35 (TA-L). Given a set of segments with natural lengths. How many, at least, segments must be excluded to make construction of a (non-degenerate) triangle impossible?

An example of “negative“ task given earlier.

Idea of Task 36 ”Training” (IOI’2007). Mirko and Slavko are training for the tandem cycling marathon in Croatia. They need to choose a route to train on. . . . There are N cities and M roads in their country. Exactly $N - 1$ of those roads are paved, while the rest of the roads are unpaved trails . . . Riding in the back seat is easier. Because of this, Mirko and Slavko change seats in every city. To ensure that they get the same amount of training, they must choose a route with an even number of roads.

Mirko and Slavko’s competitors decided to block some of the unpaved roads, making it impossible for them to find a training route satisfying the above requirements. For each unpaved road there is a cost (a positive integer) associated with blocking the road. It is impossible to block paved roads.

Find the smallest total cost needed to block the roads so that no training route exists . . .

Probably, among the vast scope of tasks given at numerous competitions in informatics there were ones which could be considered as “negative“ in our terminology. We propose to develop such tasks systematically.

6. Conclusion

We hope that successful application of methods proposed above would yield new tasks with ”short and elegant formulation” (Dagienė and Skūpienė, 2007), and being interesting to solve. This would enlarge the scope of tasks involved into national and international informatics olympiads.

References

Burton, B.A., Hiron, M. (2008). Creating informatics olympiad tasks: exploring the black art. *Olympiads in Informatics: Tasks and Training*, 2, 16–36.

Dagienė, V., Skūpienė, J. (2007). Contests in programming: quarter century of Lithuanian experience. *Olympiads in Informatics: Country Experiences and Developments*, 1, 37–49.

Diks, K., Kubica, M., Radoszewski, J., Stencel, K. (2008). A proposal for a task preparation process. *Olympiads in Informatics: Tasks and Training*, 2, 64–74.

Kelevedjiev E., Dzhenkova Z. (2008). Tasks and training the youngest beginners for informatics competitions. *Olympiads in Informatics: Tasks and Training*, 2, 75–89.

Kemkes, G., Cormack, G., Munro, I., Vasiga, T. (2007). New task types at the Canadian computing competition. *Olympiads in Informatics: Country Experiences and Developments*, 1, 79–89.

Pankov, P.S. (2008). Naturalness in tasks for olympiads in informatics. *Olympiads in Informatics: Tasks and Training*, 2, 115–121.

Pankov, P.S., Baryshnikov, K.A. (2009). Representational means for tasks in informatics. *Olympiads in Informatics*, Selected Papers of the International Conference Joint with the XXI Olympiad in Informatics, Plovdiv, 3, 101–111.

Pankov P.S. (2010). Real processes as sources for tasks in informatics. *Olympiads in Informatics*, Selected Papers of the International Conference Joint with the XXII Olympiad in Informatics, Waterloo, 4, 95–103.

Verhoeff, T. (2009). 20 years of IOI competition tasks. *Olympiads in Informatics*, 3, 149–166.

Verhoeff, T., Horvath, G., Diks, K., Cormack, G. (2006). A proposal for an IOI Syllabus. *Teaching Mathematics and Computer Science*, 4(1), 193–216.

P.S. Pankov (1950), doctor of physical-math. sciences, prof., corr. member of Kyrgyzstani National Academy of Sciences (KR NAS), is the chairman of jury of Bishkek City Olympiads since 1985, of Republican Olympiads since 1987, the leader of Kyrgyzstani teams at IOIs since 2002. Graduated from the Kyrgyz State University in 1969, is a main research worker of Institute of Theoretical and Applied Mathematics of KR NAS, a manager of chair of the International University of Kyrgyzstan.

K.A. Baryshnikov (1985), OJSC "Finance Credit Bank", Bishkek, Kyrgyzstan. Participated in IOI'2002, in training the Kyrgyzstani teams for IOI'2003 and IOI'2004. Graduated from the Kyrgyz–Russian Slavic University in 2007.