



**П. А. Корнилов,**

*Ярославский государственный педагогический университет им. К. Д. Ушинского,*

**С. И. Носок,**

*Центр детско-юношеского технического творчества г. Ярославля*

## ОТКРЫТЫЙ КОМАНДНЫЙ ТУРНИР ЯРОСЛАВСКОЙ ОБЛАСТИ ПО ПРОГРАММИРОВАНИЮ СРЕДИ ОБУЧАЮЩИХСЯ\*

### *Аннотация*

Статья посвящена описанию Ярославского турнира по программированию: его специфики, опыта проведения, вариантов заданий очного и заочного туров, подготовки и проведения турнира программ, реализующих стратегии некоторых игр.

**Ключевые слова:** турнир программ, командные соревнования по программированию.

**DOI:** 10.32517/2221-1993-2019-18-9-36-41

### 1. Введение

Ярославский государственный педагогический университет совместно с центром детско-юношеского технического творчества г. Ярославля с 2003 года проводит открытый командный турнир Ярославской области по программированию среди обучающихся. Турнир задумывался таким образом, чтобы он заметно отличался по форме проведения от других соревнований по программированию и отстоял по срокам проведения подальше от остальных олимпиад и конкурсов (конец апреля).

Первоначально турнир был организован как соревнование учебных заведений дополнительного образования, но в дальнейшем в силу своей оригинальности и доступности он стал востребованным и преобразовался в открытый турнир, куда стали допускать команды не только из центров технического творчества школьников, но и из общеобразовательных школ и колледжей. При этом команда учебного заведения состоит из трех участников, обучающихся в IX—XI классах (или на любых курсах колледжа) и трех участников более младшего возраста. Допускается участие неполных команд (только из старших или только из младших участников).

### 2. Регламент турнира

Турнир состоит из двух туров — очного и заочного.

**В первом (очном) туре** участникам предлагается решить примерно 10 задач, отдельно составленных для младшей и старшей возрастных групп. Продолжительность первого тура — три часа. На команду предоставляется один компьютер.

Во время первого тура разрешается пользоваться любой литературой и личными записями. Запрещается использовать любую информацию в электронном виде, личные компьютеры или калькуляторы, мобильные средства связи. Во время тура участники могут общаться только с членами своей команды и представителями жюри.

**Для участия во втором (заочном) туре** участники команд заранее готовят программы (домашние задания), реализующие алгоритмы некоторых игр. Решения участников предоставляются для регистрации в турнире в день проведения перед началом очного тура в виде исходного кода (текста программы) на языках Pascal или C/C++. Решения участников, предоставляемые для регистрации, проходят предварительное тестирование, во время которого проверяются на соответствие техни-

\* Материалы к статье можно скачать на сайте ИНФО: [http://infojournal.ru/journals/school/school\\_09-2019/](http://infojournal.ru/journals/school/school_09-2019/)

#### **Контактная информация**

**Корнилов Петр Анатольевич**, канд. физ.-мат. наук, доцент, зав. кафедрой теории и методики обучения информатике, Ярославский государственный педагогический университет им. К. Д. Ушинского; *адрес:* 150000, г. Ярославль, ул. Республиканская, д. 108; *e-mail:* kornilovpa@yandex.ru

**Носок Сергей Иванович**, педагог дополнительного образования, Центр детско-юношеского технического творчества г. Ярославля; *адрес:* 150000, г. Ярославль, ул. Республиканская, д. 5; *e-mail:* rcto@list.ru

**P. A. Kornilov,**

Yaroslavl State Pedagogical University named after K. D. Ushinsky,

**S. I. Nosok,**

Centre of Children and Youth Technical Creativity of the City of Yaroslavl

#### **OPEN TEAM TOURNAMENT OF YAROSLAVL REGION ON PROGRAMMING AMONG STUDENTS**

##### **Abstract**

The article describes Yaroslavl programming tournament, its specifics, experience, variants of tasks of intramural and extramural rounds, preparation and realization of the tournament of programs that implement the strategies of some games.

**Keywords:** tournament of programs, team programming competitions.

ческим требованиям и ограничениям. Затем жюри проводит чемпионат среди предоставленных участниками (и прошедших регистрацию) программ. Схема чемпионата определяется жюри в зависимости от количества участников и технических условий турнира.

### 3. Немного статистики

В первые годы в турнире принимало участие большое количество команд от организаций дополнительного образования Ярославской области. В последующем к участию в турнире стали допускать команды школ и колледжей, и число участников возросло. Так, за последние три года статистика количества команд и участников выглядит следующим образом:

Год	Количество младших команд	Участников младших команд	Количество старших команд	Участников старших команд
2017	7	21	12	36
2018	9	26	18	53
2019	10	30	17	50

В нашем турнире принимали участие команды из Московской области и Костромы.

### 4. Очный тур

В очном туре нашего турнира система оценивания представляет собой нечто среднее между системой соревнований АСМ и системой всероссийских олимпиад по информатике. Первым и главным показателем является количество полностью решенных задач, но в качестве дополнительного параметра мы выбрали процент успешно пройденных тестов в наилучших попытках по не полностью решенным задачам. Время решения задач при этом не учитывается.

Количество задач очного тура варьируется в районе десяти, при этом некоторые задачи берутся с сайтов по программированию, но большая часть задач — авторские. Поскольку уровень подготовки участников самый разный, задачи тоже сильно различаются по сложности. В варианты (в том числе для старших школьников) обязательно включаются задачи, требующие только знания команд присваивания и ветвления и некоторой смекалки. Также в вариантах представлены задачи традиционной олимпиадной тематики — динамическое программирование, вычислительная геометрия, переборные задачи, задачи на знание алгоритмов на графах, обработки текстов, моделирование.

**Приведем пример варианта задач для старших школьников одного из последних турниров.**

#### Задача 1 «Волшебное дерево».

Прапорщик Иванов подошел к волшебному дереву, на котором висело  $K_1$  апельсинов,  $K_2$  ананасов и  $K_3$  бананов. Прапорщик знал, что если он стукнет по дереву, то с него упадет один какой-нибудь вкусный плод. Также ему было известно, что на дереве всегда должно оставаться не менее  $M_1$  фруктов одного вида и не менее  $M_2$  фруктов другого вида. Прапорщик не мог

определить, какое наибольшее количество раз он может стучать по дереву, чтобы гарантированно не нарушить данное условие. Напишите программу, которая поможет ему это сделать.

*Комментарий.* Задача не требует развитой техники программирования, однако, оказалась достаточно сложна для понимания и реализации. На самом деле, если упорядочить  $K_1$ ,  $K_2$ ,  $K_3$  и  $M_1$ ,  $M_2$ , то останется проверить и сравнить всего два варианта — когда все три значения  $K$  станут меньше  $M_1$  и когда два из них станут меньше  $M_2$ .

#### Задача 2 «Новогодний бал».

Петя и Маша вышли из дома на новогодний бал в момент времени  $HH:MM$ . Пройдя  $1/K$  часть пути, Петя вспомнил, что забыл дома маску. Он повернул домой, взял маску и сразу же пошел снова в школу. В результате он опоздал к началу бала на  $N$  минут, а Маша пришла за  $M$  минут до начала. Все время Петя и Маша шли с одинаковой скоростью. Во сколько начался новогодний бал в школе?

*Комментарий.* Эта задача обобщает известную задачу, которую можно встретить в кружках по математике для начальных классов. Она требует некоторой смекалки и минимальных навыков работы со строками.

#### Задача 3 «Игра на деньги».

Петя и Вася играли в некоторую игру на деньги  $K$  раз подряд. Получилось так, что они по очереди проигрывали друг другу ровно половину имеющихся на тот момент денег. Первым проиграл Петя. В конце они посчитали имеющиеся у них деньги  $X$  и  $Y$ . Сколько денег в начале игры было у Пети?

*Комментарий.* Задача относится к утешительным. Для ее решения требуется смоделировать игру в обратном порядке и следить за возможным выходом за пределы неотрицательных чисел.

#### Задача 4 «Эпочмаки».

С приближением холодов любимая крыса Василия Федоровича решила готовить запасы на зиму. На ее радость Василий Федорович неожиданно решил продемонстрировать свое кулинарное мастерство и приготовил  $N$  эпочмаков, оставив их на некоторое время без присмотра.

В системе жизненных ценностей крысы эпочмак определяется как треугольник, заданный длинами своих сторон (ввиду отсутствия кулинарного мастерства у Василия Федоровича эпочмаки получались в виде произвольных невырожденных треугольников). Основное качество эпочмака по логике крысы — это его площадь (чем больше площадь, тем более ценен эпочмак).

Крыса была бы рада утащить себе в гнездо, свитое в полости дивана, самый ценный эпочмак целиком, но прогрызенная ею щель имеет ограниченные размеры и туда могут пролезть не все эпочмаки. Щель задается своей шириной  $D$  (вещественное число) в сантиметрах. Эпочмак может быть ориентирован произвольно в плоскости отверстия, и если хотя бы в одном положении он проходит в щель, то он считается подходящим.

Необходимо найти номер наиболее ценного эпочмака, который проходит в отверстие.

*Комментарий.* Задача взята с одного из сайтов по программированию, для ее решения достаточно знаний школьной программы по математике и информатике.

#### Задача 5 «Шахматный путь».

Клетчатое поле размером  $N \times M$  раскрашено в два цвета — белый и черный.

Назовем путь (т. е. последовательность клеток, в которой каждая пара соседних клеток имеет общую сторону) шахматным, если цвета клеток в нем чередуются. Требуется по известной раскраске поля определить, есть ли шахматный путь из клетки  $(1, 1)$  в клетку  $(N, M)$ . Если такого пути нет, то надо вывести сообщение «No solution», а если шахматные пути есть, то надо вывести длину кратчайшего из них.

*Комментарий.* Это модификация стандартной задачи на нахождение кратчайшего пути в лабиринте. Для ее решения требуется знание метода динамического программирования либо «волнового» алгоритма, или обхода в ширину, а также техника программирования, не выходящая за рамки школьного курса.

#### Задача 6 «Кусок книги».

Из книги выпал кусок. Номер первой страницы из выпавшего куска —  $A$ . Номер последней страницы выпавшего куска записан теми же цифрами, что и в числе  $A$ , но в другом порядке. Какова наименьшая возможная длина выпавшего куска?

*Комментарий.* Решение задачи требует техники работы с цифрами числа, но имеет и «подвох», связанный с нумерацией страниц в книге, а именно с их четностью.

#### Задача 7 «Число квадратов».

На клетчатой бумаге нарисован квадрат  $N \times N$  клеток, стороны которого проходят по линиям сетки. В нем по некоторым из внутренних горизонтальных и вертикальных линий сетки проведены линии, проходящие через весь квадрат. Каково число разных квадратов, ограниченных нарисованными линиями?

*Комментарий.* Задача не очень проста идейно, особенно учитывая, что размеры поля могут быть достаточно велики (до 1000), чтобы исключить перебор всех квадратов на поле или перебор всех нарисованных линий. В ней удобно сначала отсортировать данные по вертикали и горизонтали, а затем аккуратно посчитать отдельно по вертикали и по горизонтали количество вариантов получения между нарисованными линиями фиксированного расстояния (для каждого значения от 1 до  $N$ ). После этого останется только просуммировать произведения счетчиков с одинаковыми номерами.

#### Задача 8 «Дележ печенья».

На весенних каникулах группа одноклассников решила сходить в однодневный поход. Во время привала ребята стали делить поровну печенье из одинаковых пачек. После того как они разделили  $K_1$  пачек печенья, оставалось  $M_1$  штук, а после того как они разделили  $K_2$

пачек, оставалось  $M_2$  штук. Какое наименьшее количество одноклассников могло отправиться в поход?

*Комментарий.* Из техники программирования для решения задачи требуется только умение работать с делителями чисел и НОД, однако с математической стороны задачи никто из участников полностью не справился. Для решения задачи требуется двумя способами распределить  $K = \text{НОК}(K_1, K_2)$  пачек печенья среди участников похода. Если числа  $K_1 \cdot M_2$  и  $K_2 \cdot M_1$  различны, то разность  $(K_1 \cdot M_2 - K_2 \cdot M_1) \div \text{НОД}(K_1, K_2)$  делится на количество участников. Тогда среди ее делителей требуется найти наименьший с ограничением, чтобы он был строго больше  $M_1$  и  $M_2$ . Например, если при делении двух пачек осталось 24 печенюшки, а при делении шести пачек — четыре печенюшки, то число участников похода является делителем числа  $24 \cdot 3 - 4 = 68$  и должно быть при этом больше 24. Минимальный такой делитель равен 34, что и является ответом. В случае, когда числа  $K_1 \cdot M_2$  и  $K_2 \cdot M_1$  одинаковые, требуется среди чисел, которые строго больше  $M_1$  и  $M_2$ , найти наименьшее  $N$ , для которого  $\text{НОД}(N, K_1)$  является делителем  $M_1$ , а  $\text{НОД}(N, K_2)$  является делителем  $M_2$ . Например, если при делении четырех пачек осталось шесть печенюшек, а при делении шести пачек осталось девять печенюшек, то число 10 не может быть ответом, так как при делении шести пачек печенья на 10 человек обязательно должно остаться четное число печенюшек, поэтому ответом будет 11.

#### Задача 9 «Палиндром».

В данной строке определить длину самого длинного палиндрома, который можно получить вычеркиванием из строки некоторых символов.

*Комментарий.* Это известная задача на динамическое программирование (так называемая «треугольная динамика»). В нем задача решается сначала для всех подстрок длины 1, потом длины 2, ..., длины  $N$ . Для заполнения очередной клеточки рабочей таблицы используется либо максимальное из двух значений с предыдущей диагонали (если в начале и конце подстроки стоят разные символы), либо одно значение с предыдущей диагонали (если в начале и конце подстроки стоят одинаковые символы). Элемент рабочей таблицы с номерами  $[1, N]$  и будет ответом.

#### Задача 10 «Тест».

Трое учеников — Петя, Вася и Трофим — на уроке информатики писали тест, состоящий из  $N$  вопросов. На каждый вопрос возможны были только ответы «Да» или «Нет». Петя и Вася рано сдали работы, и учитель, проверив их, сообщил каждому ученику количество верных ответов. Работу Трофима учитель проверить не успел.

Нужно по имеющейся информации (матрицы ответов на тест, состоящей из трех строк из знаков «+» и «-», и количеству правильных ответов у Пети и Васи) определить, на какое максимальное количество верных ответов может рассчитывать Трофим.

*Комментарий.* Ограничение на количество вопросов таково (до 18), что задача решается полным перебором подмножеств (вопросов с ответами «Да») из  $N$  элементов.

### Задача 11 «Таблица с цифрами».

На клетчатой доске размером  $N \times M$  в каждой клетке написали одну из цифр. Затем посчитали суммы записанных цифр по строкам и по столбцам и записали их. После этой операции все цифры были заменены на буквы, причем одинаковые цифры — на одинаковые буквы, а разные цифры — на разные буквы.

Требуется составить программу, которая по имеющейся таблице с буквами и посчитанным суммам по строкам и столбцам будет определять, какие цифры какими буквами закодированы, или выводить сообщение «Many solutions».

#### Комментарий.

Эта задача изначально задумывалась как сложная и в техническом, и в математическом плане. Задачи такого уровня сложности мы регулярно включаем в турнир, потому что в течение многих последних лет в нем участвуют призеры, а иногда и победители Всероссийской олимпиады школьников по информатике и Всероссийской командной олимпиады школьников по программированию. (Заметим, что в 2019 году команда, в составе которой выступали Валерий Згурский, Михаил Кузнецов, Дмитрий Судаков из Рыбинска, смогла впервые в истории турнира справиться со всеми задачами очного тура за отведенное время.)

Частичное решение задачи можно получить перебором различных значений букв. Однако, когда в таблице представлен полный набор из 10 букв, перебор не проходит по времени. В авторском решении по данным задачи составляется система линейных уравнений и решается методом Гаусса. При этом значения некоторых переменных могут определиться, что существенно ускоряет перебор. После решения системы аккуратно реализуется перебор значений свободных переменных, который, как правило, очень быстро заканчивается.

## 5. Заочный тур — описание игр и процедуры их проверки

Примерно за месяц-полтора до дня соревнований будущим участникам рассылаются правила каких-либо логических игр (одна игра — для старшей возрастной категории, другая — для младшей). В качестве домашнего задания участникам предлагается разработать программу, которая являлась бы реализацией стратегии одного из игроков, т. е. могла бы в соответствии с правилами и более или менее разумно выполнять ходы. Непосредственно в день проведения соревнований среди программ участников проводится турнир, в ходе которого и выявляется сильнейшее решение, лучший алгоритм. При этом участники имеют возможность наблюдать за ходом турнирной борьбы своих программ на экранах компьютеров.

### 5.1. Выбор игры

Игра, предлагаемая участникам для программирования, может быть модификацией какой-либо известной настольной, или логической, или компьютерной игры (многочисленные примеры можно найти, например, в [1, 2]), адаптированной для данного турнира, либо быть новой, оригинальной, авторской. Выбор необходимо делать с учетом следующих моментов.

Во-первых, правила выполнения ходов и формат входных/выходных данных не должны быть сложными и запутанными.

Во-вторых, характер выбираемой для турнира игры должен допускать различные варианты стратегии — жадный алгоритм, перебор позиций, различного рода эвристики и прочие варианты и их комбинации. Для игр с полной информацией выбирается такой размер задачи, который не позволяет полностью вычислить выигрышную стратегию за отведенное на ход время. Другими словами, у участников должно быть достаточно широкое поле для реализации алгоритма стратегии игры, для проявления своих творческих амбиций, соответствующих их пониманию данной проблемы, уровню теоретической подготовки и практических навыков в программировании.

В-третьих, важным моментом является зрелищность игры. При просмотре матча участники должны иметь возможность следить за ходом игры, по возможности успевать оценивать сложившуюся игровую ситуацию, анализировать стратегию программ, предугадывая возможные ходы программ-соперниц, переживая за свои программы. Просмотр матчей должен проходить в атмосфере спортивного состязания. Именно переживания за своего игрока, спортивный азарт, возникающие в пылу турнирной борьбы, разжигают интерес участников и желание разработать более сильные программы, найти лучшие стратегии и реализации этих стратегий (может быть, для других игр, для следующего турнира). Именно в этот момент может возникнуть желание глубже и серьезнее изучать теорию и практику программирования. Такой турнир с интересом могут наблюдать не только участники, но и приглашенные гости — школьники, только начинающие заниматься программированием, учителя информатики, представители спонсоров и пр. Именно наличие такого совместного просмотра выгодно отличает данный турнир от других подобных соревнований.

Просмотр проходит с использованием специально разработанного организаторами программного обеспечения, позволяющего участникам наблюдать за процессом игры на экране монитора или на большом экране с использованием проектора. Параллельно на отдельном экране выводится турнирная таблица, которая в динамике отражает ход турнира.

В качестве примера далее описана игра «Биокляксы», которую авторы придумали для турнира, прошедшего в апреле 2019 года. Согласно правилам игры, программы участников должны управлять многоклеточными «организмами», которые нейтрализуют вредные вещества, расположенные в ячейках игрового поля. Игра оказалась достаточно интересной как для программирования, так и для просмотра. На экране очень хорошо было видно поведение «микроорганизмов», сильные и слабые стратегии игроков, их особенности.

### 5.2. Тестирование

Для того чтобы помочь участникам до начала турнира проверить работоспособность своих программ, организуется предварительное тестирование. Минимум за неделю до дня соревнований участники получают возможность по сети послать свои программы в систему, которая проводит проверку их корректности.

На этапе предварительного тестирования программы участники проводят несколько матчей с тестовыми игроками. Тестовый игрок — это программа-игрок, подготовленная организаторами, которая работает корректно и полностью соблюдает правила и сценарий игры, но не обладает при этом какой-либо сильной стратегией и чаще всего выполняет случайно выбранные ходы.

Такое же тестирование проводится в день проведения соревнования для регистрации программ участников в турнире.

Турнир программ при относительно небольшом числе участников проводится по круговой схеме. Каждая программа должна провести матч с каждым из соперников. Матч может состоять из нескольких игр, в зависимости от начальной конфигурации игрового поля, права первого хода и т. п. При большом количестве участников возможны более сложные схемы проведения турнира, включающие предварительные туры.

### 5.3. Игра «Биокляксы»

В биохимическом кластере промышленно-инновационного технопарка проводятся научные исследования по изучению возможностей переработки отходов промышленного производства при помощи примитивных многоклеточных микроорганизмов (ученые назвали их биокляксами). Смесь химических веществ, которую необходимо нейтрализовать, очень токсична и опасна. Она состоит из углерода, хлора, мышьяка, свинца, ртути и урана. Причем урана в смеси может быть около 5 %, а остальные вещества находятся примерно в равной пропорции. Рассматривается также возможность управления микроорганизмами при помощи искусственного интеллекта. Вам поручено написать программу, которая позволит организму действовать наиболее эффективно.

#### *Правила игры.*

В игре участвуют два игрока. Поле для игры имеет размер 26 × 26 ячеек. Каждая ячейка поля содержит одно из указанных выше веществ. Каждый игрок управляет одним многоклеточным микроорганизмом. В начале партии своим первым ходом игроки выставляют управляемые ими микроорганизмы в начальную позицию. Микроорганизм в этот момент имеет форму квадрата 2 × 2 клетки, каждая клетка микроорганизма занимает одну ячейку поля. Далее в процессе игры участники выполняют ходы по очереди.

Выполняя ход, игрок должен передвинуть несколько клеток организма (не менее одной клетки) в новую позицию. При этом должны соблюдаться следующие правила:

- часть клеток организма (не менее одной) при выполнении хода должны оставаться неподвижными, образуя при этом связную область;
- каждая из клеток, которая будет перемещена в новую позицию, должна иметь общую сторону минимум с одной из клеток из неподвижной части организма;
- новая позиция каждой перемещаемой клетки также должна иметь общую сторону хотя бы с одной клеткой из неподвижной части организма и должна быть свободна, т. е. в ней перед выполнением хода не должно быть другой клетки (данного микроорганизма или микроорганизма соперника).

Связная область — совокупность клеток, в которой каждую пару клеток можно соединить ломаной линией, проходящей по клеткам связной области и пересекающей только стороны клеток и не проходящей через вершины клеток.

Если при выполнении хода одна или более клеток микроорганизма перемещаются в ячейку на поле, содержащую углерод, то тело организма прирастает на 1 за каждую такую клетку. Прирост происходит за счет того, что в момент перемещения клетки в ячейку с углеродом происходит деление этой клетки, при котором одна дочерняя клетка остается в прежней позиции (при этом она не считается принадлежащей неподвижной части микроорганизма), а вторая занимает ячейку с углеродом. Это правило не учитывается во время выполнения первого хода каждого игрока, т. е. в момент, когда игрок выставляет микроорганизм на поле.

Если при выполнении хода хотя бы одна клетка микроорганизма перемещается в ячейку поля с ртутью, то организм временно парализуется и игрок пропускает следующий ход. Это правило работает и во время выполнения первого хода.

Каждая ячейка поля, в которую впервые попадает один из микроорганизмов в процессе игры, считается нейтральной и свободной от химического вещества. За нейтрализацию каждой ячейки с веществом игроку начисляются баллы: за углерод — 1, за хлор — 2, за мышьяк — 3, за свинец — 5, за ртуть — 8 и за уран — 13. Далее в процессе игры перемещаться по нейтральным клеткам можно, но баллов игроку они уже не добавляют.

Цель игры — набрать больше баллов за нейтрализацию ячеек поля.

Игра завершается, если:

- все ячейки поля нейтрализованы;
- один из игроков оказался заперт другим игроком и не может выполнить ход;
- исчерпан лимит ходов, который устанавливается в размере 80 ходов каждого игрока.

В первом и третьем случаях побеждает тот участник, который набрал больше баллов за нейтрализацию ячеек поля. Во втором случае игрок, который не может выполнить ход, проигрывает и его сумма баллов переходит сопернику.

Далее в условии подробно описаны сценарий работы программы, формат ввода-вывода, технические условия и ограничения, пример хода игры и возникающей игровой позиции.

В таблице дан пример ввода-вывода первого игрока в течение нескольких первых ходов игры. Ввод обозначен знаком >>, а вывод — знаком <<.

## 6. Заключение

Опыт показал, что предложенный формат проведения турнира заинтересовал учащихся средних школ, колледжей, кружков и объединений по программированию. Прохождение обоих этапов турнира способствует развитию умений работать в команде, разбивать задачу на подзадачи, проходить все этапы работы над коллективным проектом. Подготовка к заочному туру у некоторых участников занимала целый

№ хода	Ввод-вывод	Ход	Комментарий
	>>	0	Игрок будет выполнять первый ход
1	<<	jd je kd ke	Игрок выставил свой микроорганизм на поле, получив за этот ход $5 + 3 + 5 + 2 = 15$ баллов
2	>>	dj dk ej ek	Игрок получил позицию соперника
3	<<	2 djf kd kf	Игрок выполняет ход двумя клетками. Обе перемещаются в ячейки с углеродом. Игрок набирает два балла, но увеличивает размеры микроорганизма до шести клеток
4	>>	2 djf dkel	Игрок читает ответный ход соперника. Соперник также увеличил размер своего микроорганизма до шести клеток
5	<<	3 jdle jelf jfkg	Из трех клеток хода две перемещаются в клетку с углеродом, и тело микроорганизма увеличивается еще на две клетки. Третья клетка перемещается в ячейку поля с ртутью. Игрок вынужден пропустить следующий ход. За ход игрок получает $1 + 1 + 8 = 10$ баллов
6	>>	3 djfi dkgj elfk	Соперник переместил три клетки и набрал за ход $3 + 3 + 5 = 11$ баллов
7	<<	0	Игрок пропускает ход, так как на предыдущем ходе выполнил перемещение в ячейку с ртутью
8	>>	3 figi ejhj ekgk	Соперник переместил три клетки. Одну из них — в ячейку с углеродом, увеличив число клеток организма до семи
9	<<	4 jdke jejj lelg lfkh	Игрок переместил четыре клетки, набрав за ход $5 + 2 + 2 + 2 = 11$ баллов
		...	... и т. д. до конца игры

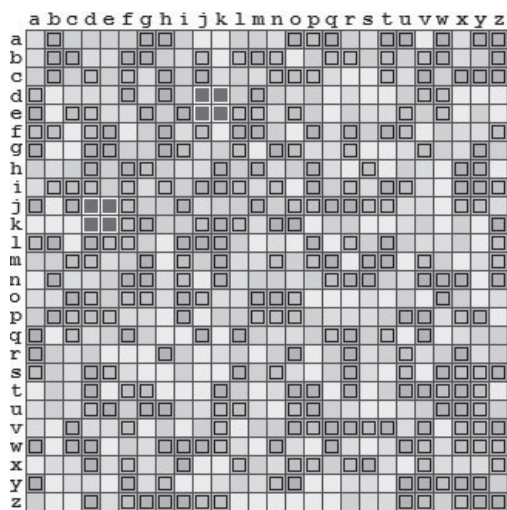


Рис. 1. Игроки выставили микроорганизмы на поле. Ярко-красный цвет — у первого игрока, ярко-синий — у второго

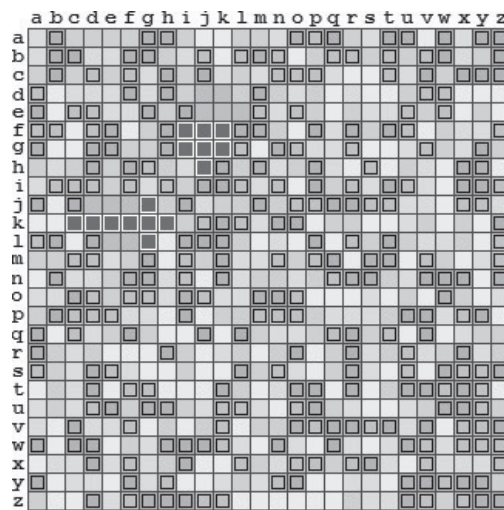


Рис. 2. Положение в игре после девятого хода из примера. Серым цветом обозначены нейтрализованные ячейки поля

месяц, что в большей степени моделирует профессиональную деятельность программиста, чем решение задач очного тура.

По положению командный турнир Ярославской области по программированию является открытым, поэтому организаторы могут рассмотреть вопрос об участии в турнире команд из других областей при наличии свободных мест. Возможно участие в заочном туре вне конкурса (также по согласованию с организа-

торами). Если читателей заинтересовали подробности проведения нашего турнира или появилось желание поучаствовать в нем, напишите авторам.

#### Список использованных источников

1. Арсак Ж. Программирование игр и головоломок: пер. с франц. М.: Наука, гл. ред. физ.-мат. лит., 1990.
2. Гук Е. Я., Сухарев А. В. Интеллектуальные игры и развлечения. М.: ФАИР-ПРЕСС, 1999.