

**Е. А. Мирончик,**

лицей № 111, г. Новокузнецк, Кемеровская область

## ГРАФЫ И СИСТЕМЫ ЛОГИЧЕСКИХ УРАВНЕНИЙ

### Аннотация

В статье рассматривается метод решения системы логических уравнений. Он позволяет разбить трудоемкое задание на несколько простых и составить схему решения задания ЕГЭ высокого уровня сложности. Использование графа для решения системы логических уравнений связывает внешне принципиально различные теоретические темы школьного курса информатики.

**Ключевые слова:** граф, логика, логические операции, система логических уравнений, ЕГЭ.

### Контактная информация

**Мирончик Елена Александровна,**  
учитель информатики лицея № 111, г. Новокузнецк, Кемеровская область; *адрес:* 654007, Кемеровская область, г. Новокузнецк, ул. Кирова, д. 35а; *телефон:* (3843) 46-82-08; *e-mail:* mironch-elena@yandex.ru

**Е. А. Mironchik,**  
Lyceum 111, Novokuznetsk, Kemerovo Region

### GRAPHS AND BOOLEAN EQUATIONS SYSTEMS

#### Abstract

The article considers the method for solving a system of boolean equations. It allows you to split time-consuming task into several simple tasks and to chart the solution of the USE task of a high level of difficulty. The usage of the graph to solve the system of boolean equations binds visually different theoretical topics of the school informatics course.

**Keywords:** graph, logic, boolean operations, system of boolean equations, Unified State Examination.



Одним из основных умений школьника, используемым им не только в учебе, но и на бытовом уровне, является умение формализовывать объекты и задавать отношения между ними с помощью схем. Часто к построенной схеме можно применить термин «граф». Графы и алгоритмы на графах проходят через весь курс школьной информатики, включая курс олимпиадного программирования.

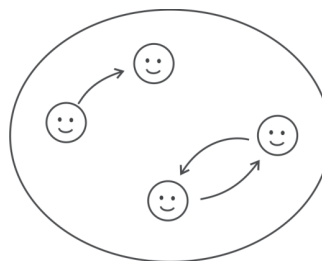
**В данной статье будет рассмотрено, как от графов и схем, изучаемых в начальной школе, перейти к графам и алгоритмам на графах для решения одного из заданий ЕГЭ высокого уровня сложности — задания на логику.**

Наглядность и схематичная краткость графа дают возможность познакомить школьника с термином «граф» до добавления его в лексикон ученика. В 1967 году в Бельгийском математическом педагогическом центре Фредерик и Жорж Паппи проводили эксперимент по изучению графов с детьми дошкольного возраста [8]. В результате проведенного исследования было доказано, что дети пяти-шести лет успешно справляются с формализацией, изображают однородные объекты точками и легко объединяют объекты-точки в множества. На следующем шаге с помощью стрелок дети описывают отношения между объектами, изображая граф отношений.

Примерами заданий, с которых можно начинать пропедевтику базовых понятий математики и информатики, могут служить приведенные ниже два задания.

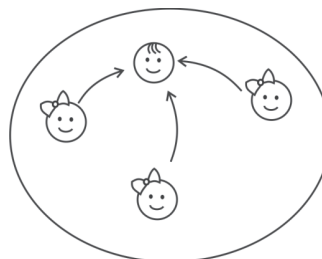
### Задание 1.

Во дворе гуляли четверо детей. Стрелка задает отношение «мой родной брат». Отметить на рисунке условными обозначениями мальчиков  и девочек .



### Задание 2.

В семье четверо детей. Стрелка задает отношение «мой родной брат». Нарисовать граф, задающий отношение «моя родная сестра».



Разбивая множество объектов на подмножества по различным свойствам и оформляя стрелками отношения между объектами «брат», «старше», «ребенок», дети приближаются к определению математической функции.

Комбинаторные задачи требуют перебора всех возможных вариантов или поиска лучшего из всех существующих. Для решения задач этого типа учащиеся строят дерево решений и анализируют ориентированный и взвешенный граф. Школьники начинают применять некоторые алгоритмы на графах без их формального описания. Заметим, что блок-схема некоторого алгоритма также является ориентированным графом, но если блок-схема описывает повторяющийся процесс, то термин «граф» можно уточнить — «циклический граф».

В рамках школьного курса информатики учащиеся к выпускному классу должны освоить основные термины теории графов: вершина, ребро, цикл (петля), ориентированный граф.

Реже в школьном курсе встречаются двудольный и многодольный графы [1]. **Двудольный граф** — это граф, вершины которого можно разбить на два множества таким образом, что ребра будут соединять вершины одного множества с вершинами другого множества. Термин «двудольный граф» тесно связан с понятием функции курса математики. Задавая ребрами графа отображение одного множества в другое, мы описываем функцию, а если граф является **многодольным**, т. е. состоящим из некоторого количества повторяющихся двудольных графов, мы можем переходить к функции, описывающей рекуррентное соотношение между переменными, которое очень часто используется в программировании.

Постепенно расширяя область применения графов, можно с их помощью решить **одно из самых сложных заданий ЕГЭ — нахождение количества решений системы логических уравнений**. Применение графов для анализа системы значительно упрощает само решение. Хочется отметить, что это задание ЕГЭ является единственным заданием, в котором от учащегося требуются не только анализ построенного графа, но выявление лучших объектов для его построения и запись отношений на ориентированном графе.

При выполнении данного задания по алгоритму решения методом отображения [3, 7] сначала строится двудольный граф решения первого уравнения. Подключение следующих уравнений переводит граф в многодольный ориентированный граф. При этом используются базовые понятия математики «множество» и «функция», известные учащимся из курса математики.

Можно рассматривать это задание и с другой стороны: для описания модели строятся рекуррентные отношения, а само задание является примером задания на динамическое программирование, которое предлагается на олимпиадах по программированию.

Проиллюстрируем решение на примере.

### Система 1.

$$\begin{cases} x_1 + x_2 = 1, \\ x_2 + x_3 = 1, \\ x_3 + x_4 = 1, \\ x_4 + x_5 = 1. \end{cases}$$

В первом уравнении две переменных, так же как и в последующих уравнениях.

Первый двудольный граф будет соответствовать первому уравнению системы и свяжет множество значений  $x_1$  и множество значений  $x_2$ , второй свяжет множество значений  $x_2$  и множество значений  $x_3$ , и т. д.

$$x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5$$

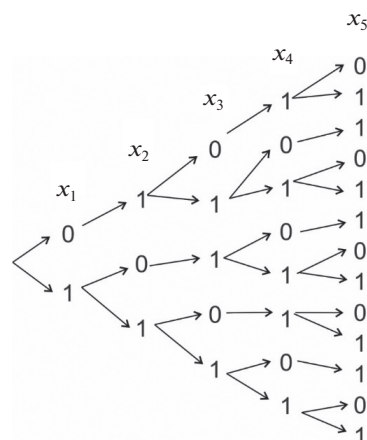
В наших рассуждениях будем переходить от первого уравнения ко второму, от него — к третьему, опираясь на значения общих для двух соседних уравнений переменных:

$$x_1 \Rightarrow x_2 \Rightarrow x_3 \Rightarrow x_4 \Rightarrow x_5.$$

Представим дерево решений первого уравнения в табличном виде:

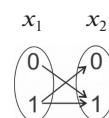
$x_1$	$x_2$
0	1
1	0
	1

Продолжая построение, получим бинарное дерево, состоящее из одинаковых поддеревьев. От каждой единицы дерево идет в два направления — к нулю и к единице, а от нуля — только к единице.

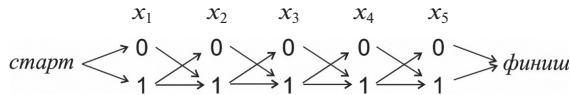


За счет объединения повторяющихся поддеревьев построенное дерево может быть сокращено.

С другой стороны, имеем множество значений  $x_1$ , состоящее из двух элементов  $\{0, 1\}$ , и множество различных значений  $x_2$ , также состоящее из двух элементов  $\{0, 1\}$ . Отображение первого множества во второе оформим в виде двудольного графа, построенного по дереву решений первого уравнения:



Так как второе уравнение отличается от первого только индексами, то и граф второго перехода — от  $x_2$  к  $x_3$  — будет отличаться только индексами. От множества  $x_1$  будем переходить к множеству  $x_2$  и от  $x_2$  — к  $x_3$ . Каждая из переменных имеет два значения — 0 и 1. В многодольный граф решения этой системы необходимо включить для каждой переменной свою вершину 0 и свою вершину 1. Для того чтобы подчеркнуть сходство этой задачи с задачей о количестве маршрутов, в граф решения добавлены два узла — узел *старт* и узел *финиш*.



Если перечислять все маршруты, последовательно записывая посещенные узлы 0 и 1, то будут выписаны все битовые цепочки [11], являющиеся решением этой системы.

Каждый двудольный граф может быть описан функциями, задающими рекуррентные соотношения: пусть  $X_i(0)$  — это функция, показывающая, сколько маршрутов ведет в узел 0 множества  $x_i$ , а  $X_i(1)$  — в узел 1 этого же множества. Тогда решение описывается объединением двух функций:

$$X_i(0) = \begin{cases} 1, & \text{при } i = 1 \\ X_{i-1}(1), & \text{при } i > 1 \end{cases}$$

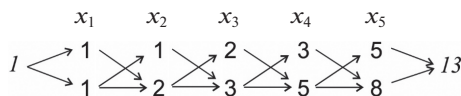
$$X_i(1) = \begin{cases} 1, & \text{при } i = 1 \\ X_{i-1}(0) + X_{i-1}(1), & \text{при } i > 1. \end{cases}$$

Вычисление количества решений по описанным функциям можно оформить в виде таблицы:

Функции	Количество маршрутов				
	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$
$X_i(0)$	1	1	2	3	5
$X_i(1)$	1	2	3	5	8

На последнем шаге остается сложить количество маршрутов, полученное в столбце  $X_5$ :  $5 + 8 = 13$ . То есть **количество решений системы уравнений 1 равно 13**.

На практике удобно значения функции вписывать в граф, не записывая общий вид функции:



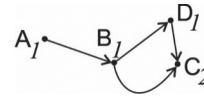
На *старте* всегда стоит 1 (единственный способ оказаться на *старте* — это в нем быть). Все промежуточные вычисления ведутся по стрелкам. А значения в узле *финиш* складываются из двух значений:  $5 + 8 = 13$ .

Все приведенные рассуждения применимы для решения логических уравнений, так как систему можно записать в виде уравнения. Например, система 1 равносильна уравнению:

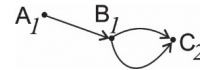
$$(x_1 + x_2) \cdot (x_2 + x_3) \cdot (x_3 + x_4) \cdot (x_4 + x_5) = 1.$$

Если система не будет однотипной, то многодольный граф решения будет состоять из разных двудольных графов. Опираясь в рассуждениях на сходство графического решения задачи о количестве маршрутов и решения системы логических уравнений после построения графа, рассмотрим, к чему приведет частичное изменение графа.

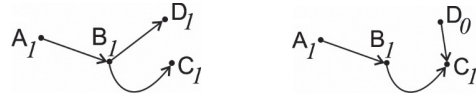
Во всех случаях будем подсчитывать количество маршрутов от А до С.



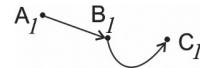
От вершины А до вершины С — два маршрута. В узел D можно попасть одним способом, и от D ведет всего один маршрут. Граф можно заменить на такой:



Удаление одного из ребер вершины D:



равносильно удалению вершины D вместе с ребрами:



Если вершина А не первая в маршруте и в нее «добрался»  $n$  способами, то количество маршрутов от А увеличивается в  $n$  раз.

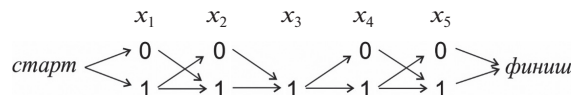
Чтобы решить систему логических уравнений, необходимо выделить простые зависимости. Метод отображений позволяет оформить зависимости в виде схем. С точки зрения методики преподавания, системы логических уравнений следует разбирать после задания о количестве путей в ориентированном графе. Графический способ решения удобен при решении системы однотипных уравнений с добавленным ограничением.

Добавим к системе 1 условие  $x_3 = 1$ , то есть рассмотрим такую систему:

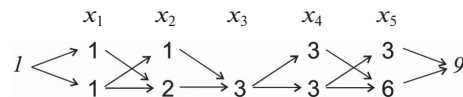
**Система 2.**

$$\begin{cases} x_1 + x_2 = 1, \\ x_2 + x_3 = 1, \\ x_3 + x_4 = 1, \\ x_4 + x_5 = 1, \\ x_3 = 1. \end{cases}$$

Добавленное ограничение приведет к удалению из многодольного графа решения системы 1 узла 0 переменной  $x_3$  и инцидентных этой вершине ребер (тех ребер, которые ведут к удаленной вершине).



Вычисления в этом случае будут такими:



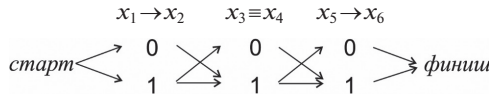
То есть в данном случае **система 2 будет иметь девять решений**.

Теперь рассмотрим систему, которую обычно решают, используя подстановку [9].

**Система 3.**

$$\begin{cases} (x_1 \rightarrow x_2) + (x_3 \equiv x_4) = 1, \\ (x_3 \equiv x_4) + (x_5 \rightarrow x_6) = 1. \end{cases}$$

В графе решения отметим узлы, соответствующие 0 и 1 значениям выражений:

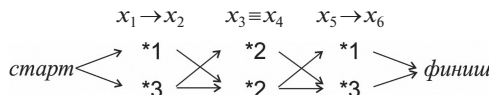


Выражение  $x_1 \rightarrow x_2$  равно 0 только в одном случае, а в трех случаях оно равно 1. Это приведет к увеличению в три раза числа, которое мы будем ставить при выполнении вычислений в этом узле.

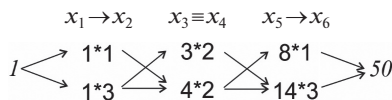
То же самое касается узла 1 выражения  $x_5 \rightarrow x_6$ .

Выражение  $x_3 \equiv x_4$  равно 1 в двух случаях, и в двух случаях оно равно 0.

Опираясь на построенный граф, составим схему вычисления, в которую внесем множители, увеличивающие количество соответствующих узлов:



Остается провести вычисления по составленной схеме:



То есть **количество решений системы 3 — 50.**

Каждое из уравнений системы имеет простое решение. Перенос простых решений на общую схему сродни процедурному программированию.

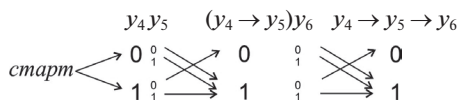
Рассмотрим еще одну систему, особенность которой заключается в том, что она состоит из нескольких элементарных выражений, соединенных в простые уравнения. Кроме того, два соседних уравнения имеют общее выражение.

**Система 4.**

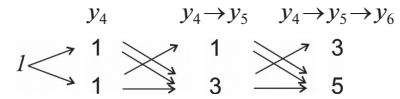
$$\begin{cases} (x_1 + y_1) \oplus (x_2 \cdot y_2) = 1, \\ (x_2 \cdot y_2) \cdot (x_3 \rightarrow y_3) = 0, \\ (x_3 \rightarrow y_3) + x_4 \cdot x_5 \cdot x_6 = 1, \\ x_4 \cdot x_5 \cdot x_6 + (y_4 \rightarrow y_5 \rightarrow y_6) = 1. \end{cases}$$

Самое сложное выражение этой системы:  $y_4 \rightarrow y_5 \rightarrow y_6$ . Для решения нам важно знать, сколько раз из восьми возможных вариантов это выражение равно нулю, а сколько — единице.

Воспользуемся методом отображений [2, 5]. Построим граф выражения  $y_4 \rightarrow y_5 \rightarrow y_6$ :

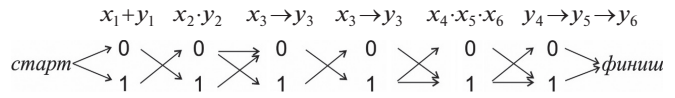


Выполним вычисления:

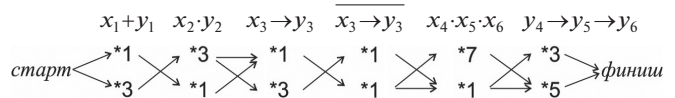


Число 3 соответствует количеству нулей, а число 5 — количеству единиц.

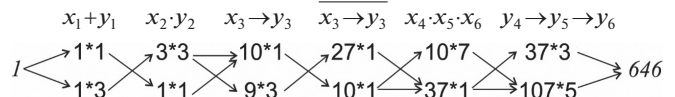
Построим граф системы:



Внесем в граф множители, соответствующие количеству нулей и количеству единиц, элементарных выражений, входящих в систему. Выражение  $(x_3 \rightarrow y_3)$  не добавляет количество переменных по сравнению с тем, что было на предыдущем шаге ( $x_3 \rightarrow y_3$ ), поэтому множители равны 1. Схема вычислений принимает вид:



Выполним вычисления:



Итак, **количество решений системы 4 — 646.**

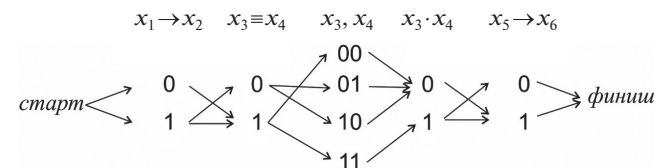
Рассмотрим случай, когда подстановка может быть проведена частично.

**Система 5.**

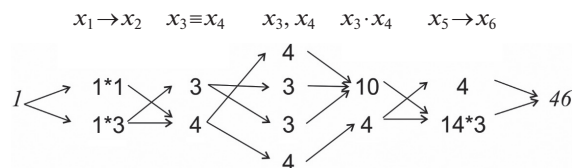
$$\begin{cases} (x_1 \rightarrow x_2) + (x_3 \equiv x_4) = 1, \\ ((x_3 \cdot x_4) + (x_5 \rightarrow x_6)) = 1. \end{cases}$$

Так как  $x_3$  и  $x_4$  встречаются в обоих выражениях системы, необходимо знать пару  $x_3, x_4$ . Все возможные пары: 00, 01, 10, 11.

Построим граф этой системы:



Выполним вычисления:



**Количество решений системы 5 равно 46.**

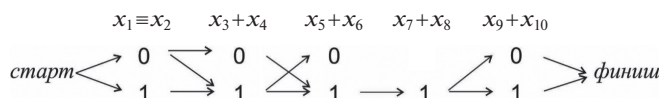
Для случая, когда в системе выражение встречается дважды, необходимо построить граф, а затем провести вычисления для двух случаев.



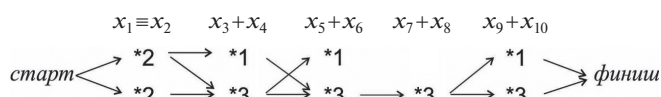
**Система 6.**

$$\begin{cases} (x_1 \equiv x_2) \rightarrow (x_3 + x_4) = 1, \\ (x_3 + x_4) + (x_5 + x_6) = 1, \\ (x_5 + x_6) \cdot (x_7 + x_8) = 1, \\ (x_7 + x_8) + (x_9 + x_{10}) = 1, \\ (x_9 + x_{10}) \rightarrow (x_1 \equiv x_2) = 1. \end{cases}$$

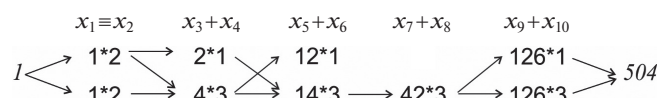
Построим граф решения этой системы без последнего уравнения. Из третьего уравнения системы следует, что  $x_7 + x_8 = 1$ , это означает, что в графе можно удалить узел 0 выражения  $x_7 + x_8$  и инцидентные этой вершине ребра.



Составим схему решения:



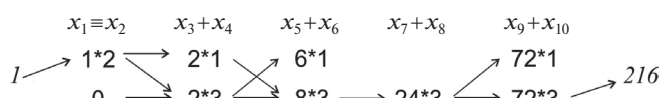
Выполним вычисления по схеме, не учитывая последнее уравнение:



Подключение последнего уравнения приведет к исключению из общего решения тех решений, при которых:

$$(x_9 + x_{10}) \rightarrow (x_1 \equiv x_2) = 0.$$

Количество лишних решений будем вычислять по той же схеме, но начать надо с  $x_1 \equiv x_2 = 0$  и завершить вычисления в  $x_9 + x_{10} = 1$ . Это означает, что из всех возможных необходимо убрать те, которые начинаются с 1 и приводят к 0 на последнем этапе. На графе это проявится удалением ребер, ведущих от старта к 1 и от 0 к финишу:



$$504 - 216 = 288.$$

**То есть количество решений системы 6 равно 288.**

Использование описанного метода решения системы логических уравнений напоминает использование процедур в программировании. Так же как в программировании, большая и сложная задача разбита на несколько очень простых. На практике разбирают системы однотипных логических уравнений, и приводит это к тому, что количество подзадач, которые необходимо решить, уменьшается. Кроме аналогии с программированием можно сказать и о том, что графы, легко осваиваемые учащимися, помогают решать задачи повышенного уровня сложности и связывают внешне разные теоретические темы школьного курса.

**Литературные и интернет-источники**

1. Авдошин С. М., Ахметсафина Р. З., Максименкова О. В. Информатика: Логика и алгоритмы: Эффективные методы решения задач: Пособие для самостоятельной подготовки (Серия «Сложные темы ЕГЭ»). М., СПб.: Просвещение, 2013.
2. Мирончик Е. А. Люблю ЕГЭ за В15, или Еще раз про метод отображения // Информатика. 2014. № 7–8.
3. Мирончик Е. А. Метод отображения // Информатика. 2013. № 10.
4. Мирончик Е. А. Презентация «Количество маршрутов в ориентированном графе». <http://www.liceum111.ru/wp-content/uploads/2014/03/Путешествие-Чеширского-кота.ppsx>
5. Мирончик Е. А. Решение одного уравнения методом отображения // Материалы XXIV международной конференции «Применение инновационных технологий в образовании». М., 2013.
6. Мирончик Е. А. Чеширский Кот путешествует по графу // Информатика. 2015. № 10.
7. Мирончик Е. А., Мирончик Е. А. Системы логических уравнений. Метод отображений // Преподавание информационных технологий в Российской Федерации: Материалы Десятой открытой Всероссийской конференции (16–18 мая 2012 года). М.: МГУ им. Ломоносова, 2012.
8. Панни Ф., Панни Ж. Дети и графы: Обучение детей шестилетнего возраста математическим понятиям / пер. с франц. М.: Педагогика, 1974.
9. Поляков К. Ю. Логические уравнения // Информатика. 2011. № 14.
10. Поляков К. Ю. Подготовка к ЕГЭ по информатике. <http://kpolyakov.spb.ru/school/ege.htm>
11. Поляков К. Ю., Ройтберг М. А. Системы логических уравнений: решение с помощью битовых цепочек // Информатика. 2014. № 12.

НОВОСТИ

**Лучшей школой — 2016 стал Президентский физико-математический лицей Санкт-Петербурга № 239**

Первое место в рейтинге школ топ-500 в 2016 году занял Президентский физико-математический лицей Санкт-Петербурга № 239. Второе место занял Республиканский лицей для одаренных детей, г. Саранск, Республика Мордовия. Третьей стала московская школа № 1329.

Рейтинг был представлен Московским центром непрерывного математического образования.

«Я думаю, этот рейтинг будет интересен для специалистов, для родителей, для всех, кто неравнодушен

к системе образования», — подчеркнула вице-премьер правительства РФ Ольга Голодец.

Она также добавила, что в топ-25 школ представлены 11 регионов РФ, а в топ-500 вошли школы 72 регионов.

В свою очередь министр образования и науки РФ Ольга Васильева отметила, что 33,5 % школьников Москвы, 10 % — Липецка и около 7 % — Санкт-Петербурга учатся в школах, вошедших в рейтинг.

(По материалам ТАСС)