

ИНФОРМАТИКА

17 мая —
Всемирный день
информационного
общества

электронная
версия журнала
в Личном кабинете
на сайте
www.1september.ru



НА ОБЛОЖКЕ

► День информационного общества был учрежден Генеральной Ассамблеей ООН в 2006 г. Выбранная дата — 17 мая — связана с тем, что в этот день в 1865 году был учрежден Международный телеграфный союз, впоследствии переименованный в Международный союз электросвязи.

В НОМЕРЕ

3 ПРОФИЛЬ

- Изучение интеллектуальных алгоритмов на примере реализации Arjori в углубленном курсе информатики
- Имитационные модели работы полиграфического комплекса и аэропорта

34 МЕТОДИКА

- От компьютерной грамотности и внедрения ИКТ к трансформации работы школы

44 ЕГЭ

- ЕГЭ: новый “маскарад в Сети”

46 ЗАНИМАТЕЛЬНЫЕ МАТЕРИАЛЫ ДЛЯ ПЫТЛИВЫХ УЧЕНИКОВ И ИХ ТАЛАНТЛИВЫХ УЧИТЕЛЕЙ

- “В мир информатики” № 218

В ЛИЧНОМ КАБИНЕТЕ

Облачные технологии от Издательского дома “Первое сентября”

Все подписчики журнала имеют возможность получать электронную версию, которая является полной копией бумажной. Для получения электронной версии:

1) Откройте Личный кабинет на портале “Первое сентября” (www.1september.ru).

2) В разделе “Газеты и журналы / Получение” выберите свой журнал и кликните на кнопку “Я — подписчик бумажной версии”.

3) Появится форма, посредством которой вы сможете отправить нам копию подписной квитанции.

После этого в течение одного рабочего дня будет активирована электронная подписка на весь период действия бумажной. Справки: podpiska@1september.ru или через службу поддержки на портале “Первое сентября”.

ИНФОРМАТИКА

ПОДПИСНЫЕ ИНДЕКСЫ по каталогу “Почта России”: 79066 — бумажная версия, 12684 — электронная версия

<http://inf.1september.ru>

Учебно-методический журнал для учителей информатики
Основан в 1995 г.
Выходит один раз в месяц

РЕДАКЦИЯ:

гл. редактор С.Л. Островский
редакторы

Е.В. Андреева,
Д.М. Златопольский
(редактор вкладки
“В мир информатики”)

Дизайн макета И.Е. Лукьянов
верстка Н.И. Пронская
корректор Е.Л. Володина
секретарь Н.П. Медведева
Фото: фотобанк Shutterstock
Журнал распространяется по подписке
Цена свободная
Тираж 18 000 экз.
Тел. редакции: (499) 249-48-96
E-mail: inf@1september.ru
<http://inf.1september.ru>

ИЗДАТЕЛЬСКИЙ ДОМ “ПЕРВОЕ СЕНТЯБРЯ”

Генеральный директор:
Наум Соловейчик

Главный редактор:
Артем Соловейчик

Коммерческая деятельность:
Константин Шмарковский
(финансовый директор)

Реклама, конференции и техническое обеспечение Издательского дома:
Павел Кузнецов

Производство:
Станислав Савельев

Административно-хозяйственное обеспечение:
Андрей Ушков

Педагогический университет:
Валерия Арсланьян (ректор)

ЖУРНАЛЫ ИЗДАТЕЛЬСКОГО ДОМА “ПЕРВОЕ СЕНТЯБРЯ”

Английский язык – Е.Богданова
Библиотека в школе – О.Громова
Биология – Н.Иванова
География – и.о. А.Митрофанов
Дошкольное образование – Д.Тюттерин
Здоровье детей – Н.Сёмина
Информатика – С.Островский
Искусство – О.Волкова
История – А.Савельев
Классное руководство и воспитание школьников – А.Полякова

Литература – С.Волков
Математика – Л.Рослова
Начальная школа – М.Соловейчик
Немецкий язык – М.Бузоева
ОБЖ – А.Митрофанов
Русский язык – Л.Гончар
Спорт в школе – О.Леонтьева
Технология – А.Митрофанов
Управление школой – Е.Рачевский
Физика – Н.Козлова
Французский язык – Г.Чесновицкая
Химия – О.Блохина
Школа для родителей – Л.Печатникова
Школьный психолог – М.Чибисова

УЧРЕДИТЕЛЬ:
ООО “ИЗДАТЕЛЬСКИЙ ДОМ «ПЕРВОЕ СЕНТЯБРЯ»”

Зарегистрировано ПИ № ФС77-58447 от 25.06.2014 в Роскомнадзоре
Подписано в печать: по графику 24.02.2016, фактически 24.02.2016
Заказ №
Отпечатано в ОАО “Первая Образцовая типография” Филиал “Чеховский Печатный Двор”

ул. Полиграфистов, д. 1, Московская область, г. Чехов, 142300
Сайт: www.chpd.ru
E-mail: sales@chpk.ru
Факс: 8 (495) 988-63-76
АДРЕС ИЗДАТЕЛЯ:
ул. Киевская, д. 24, Москва, 121165
Тел./факс: (499) 249-31-38

Отдел рекламы:
(499) 249-98-70
<http://1september.ru>
ИЗДАТЕЛЬСКАЯ ПОДПИСКА:
Телефон: (499) 249-47-58
E-mail: podpiska@1september.ru



Изучение интеллектуальных алгоритмов на примере реализации Apriori в углубленном курсе информатики

Салахова Алена,
МПУ,
aa.salakhova@yandex.ru

► В курсе информатики углубленного уровня изучается несколько самообучающихся алгоритмов, позволяющих составлять правила из большого набора данных, относящихся к такому направлению информационных технологий, как искусственный интеллект [2]. Одним из таких алгоритмов является Apriori.

Алгоритм Apriori наиболее понятен и востребован обучающимися, так как они сталкиваются с его применением ежедневно, что позволяет связать достаточно сложный теоретический материал с жизнью. Данный алгоритм позволяет пополнять и формировать базы знаний. Он был разработан в 1993 году Ракешем Агравалом, сотрудником IBM Almaden Research Center (сейчас — работником Microsoft Research), для исследования потребительских корзин, поэтому Apriori иногда называют market basket analysis, т.е. анализ

потребительских корзин. Задача поиска шаблонов покупок, совершаемых клиентами супермаркетов, актуальна сегодня и, вероятно, не потеряет актуальность в будущем. Apriori находит ассоциативные правила (*association rule mining*) — правила отношения к чему-либо, используемые для группировки данных, которые проще всего записать конструкцией “if-then”. Простым и понятным примером, встречающимся ежедневно, является расположение полок в гипермаркете: соленые орешки лежат рядом с минеральной водой и т.д. Здесь работает следующее правило: если (if) человек купил и съел соленое, тогда (then) он захочет пить и купит воду. Нахождение определенных правил, которые встречаются достаточно редко, позволяют продать потребителю тот товар, на который он бы не обратил внимания, если бы не расположение на витрине. Именно

алгоритм Apriori применяется при составлении акций и распродаж и позволяет получить больший доход. Аналогично его можно использовать для анализа продуктивности взаимодействия людей и составления эффективных команд (в менеджменте). Однако сфера применения описанными категориями не ограничивается: Apriori используется как элемент системы управления подводными лодками, в вопросах логистики, игре на бирже и автоматических ставках и т.д. В американском проекте робота для армии ARMS (*Autonomous Robots for Military Systems*, часть широко известного проекта DARPA — *Defense Advanced Research Projects Agency*) при реализации реактивного движения используется похожий принцип выбора правил для включения определенных групп сенсоров и экономии энергии [3]. У алгоритма Apriori огромное множество реализаций, используемых в различных проектах, например: Orange Data Mining (<http://orange.biolab.si/>), ARTOOL (<http://www.artool.cl/>), Weka (*Waikato Environment for Knowledge Analysis*) и другие. Apriori относится к стандартным алгоритмам анализа данных и применяется в том числе в работе таких сложных экспертных систем, как IBM Watson, один из лидеров рынка. Алгоритм Apriori также входит в состав медицинских диагностических экспертных систем [4].

Использование алгоритма буквально во всех сферах нашей жизни говорит, несомненно, о его актуальности. Важным следствием является необходимость умения реализовать Apriori на практике хотя бы для частных случаев, что доступно для понимания и усвоения обучающимися старшей школы в рамках курса информатики углубленного уровня [5].

Apriori предназначен для выявления ассоциативных правил при исследовании больших объемов информации. Главная его особенность состоит в том, что он масштабируемый, т.е. экономный по отношению к памяти и, следовательно, времени выполнения, что крайне важно для объемов, исчисляемых гигабайтами или терабайтами информации. Масштабированным алгоритм становится из-за использования свойства антимонотонности, которое служит для снижения пространства поиска подходящих элементов [8].

Под транзакциями понимается каждая покупка, включающая в себя множество выбранных человеком товаров. Исследование большого количества транзакций позволяет составить определенные правила. Чем больше было покупок (чеков), тем точнее и надежнее будет полученное правило.

Надежность правила называется достоверностью (confidence), и чем выше достоверность, тем обычно лучше правило. Чаще всего достоверность представляют в виде процентов [6].

Если анализ действий (рефлексию и создание правил человеком) можно имитировать в определенном объеме, то основной проблемой для ИИ является отсутствие интуиции, которая есть у живых существ. Для ее имитации необходимы изучение и подсчет вероятностей при анализе результатов, ис-

пользование нечеткой логики и выбор результатов не с наиболее высоким значением, а рассмотрение определенных групп решений. Ниже приведен пример, характерный для исследования потребительской корзины продавцом:

Достоверность	%
Важная ниша для проведения акций. Означает, что большинство не интересуется рассматриваемыми товарами вместе, но есть постоянные покупатели, которые выбирают данную пару (или больше) товаров. Возможно привлечение новых постоянных покупателей	5–10
Достоверный факт, на него можно положиться	>60
Была маленькая выборка или правило — неоспоримая истина	100

Для работы алгоритма необходимо привести данные к бинарному виду и определенной структуре. Для этого общую таблицу приводят к нормализованному виду. Алгоритм является поуровневым и использующим стратегию поиска в ширину и снизу вверх, поэтому так важно упорядочить таблицу исходных данных.

Первоначальный вид таблицы:

Номер чека	Наименование товара	Количество товара
102	Лосось	3
102	Икра красная	1
103	Лосось	2
103	Хлеб	3
...

Видно, что в каждом чеке может содержаться более одного товара. Количество товаров для простой реализации не важно. После преобразования таблица будет иметь вид:

Номер чека	Лосось	Икра красная	Хлеб	...
102	1	1	0	
103	1	0	1	
...

Каждая строка соответствует отдельной транзакции, т.е. списку всех покупок данного потребителя. Наличие товара отмечается 1 в случае присутствия в корзине, 0 — если товар отсутствует.

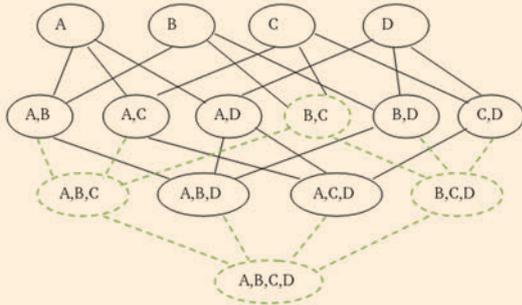
Количество столбцов в таблице равно количеству различных товаров. Нормализованная таблица имеет уникальные строки и уникальные столбцы.

Работу алгоритма можно разделить на два больших этапа:

- 1) поиск часто встречающихся наборов элементов (товаров);
- 2) извлечение из них ассоциативных правил.

Рассмотрим первый этап. Самый простой вариант решения данной задачи — это перебор всех возможных элементов, что займет $O(2^{Items})$ операций. Используется свойство антимонотон-

ности: “Любой k -элементный набор будет часто встречающимся, если все его $(k - 1)$ -элементные подмножества будут часто встречающимися”. То есть набор из четырех элементов будет часто встречающимся тогда и только тогда, когда любые (!) три элемента из этого набора будут часто встречающимися. Эти трехэлементные наборы будут часто встречающимися, если любые два элемента будут часто встречающимися, а те, в свою очередь, обязаны состоять из часто встречающихся одноэлементных подмножеств. Это и есть реализация стратегии поиска в ширину и по направлению снизу вверх.



Если приведенное условие не выполняется, множество и его составляющие отбрасываются, т.е. удаляются из памяти, что позволяет алгоритму занимать не слишком много ресурсов и эффективно справляться с большими объемами данных.

Процесс подбора кандидатов и поиска часто встречающихся выглядит следующим образом [7]:

```

Apriori (T, ε)
  L1 ← {large 1 – itemsets}
  k ← 2
  while Lk-1 ≠ 0
    Ck ← {a ∪ {b} | a ∈ Lk-1 ∧
    b ∉ a} - {s | s ⊆ c ∧ |s| = k - 1} ∉ Lk-1}
    for transactions t ∈ T
      Ct ← {c | c ∈ Ck ∧ c ⊆ t}
      for candidates c ∈ Ct
        count[c] ← count[c] + 1
    Lk ← {c | c ∈ Ck ∧ count[c] ≥ ε}
    k ← k + 1
  return ∪k Lk
  
```

На входе подается множество всех транзакций и минимальное допустимое значение поддержки ϵ . Поддержка (support) — это сумма элементов столбцов нормализованной матрицы для данной транзакции:

$$\text{sup}(ID) = \sum X \cup Y.$$

Для первого прохода поддержка товара считается как

$$\text{sup}(ID) = \sum X,$$

где X — номер столбца, т.е. ID товара, аналогично для Y .

Минимальная поддержка задается в условиях задачи и является пороговым значением, превы-

шение которого означает, что k -элементное множество является часто встречающимся.

Как было видно из схемы, кандидаты (k -элементные множества, которые, возможно, станут часто встречающимися, если их поддержка больше или равна минимальной) формируются как все возможные сочетания из $(k - 1)$ -элементных часто встречающихся подмножеств.

Когда невозможно составить кандидаты для шага k (отсутствуют часто встречающиеся $(k - 1)$ -элементные подмножества или невозможно создать сочетания), необходимо перейти ко второму этапу алгоритма — составлению ассоциативных правил. Для этого выписывают все часто встречающиеся множества.

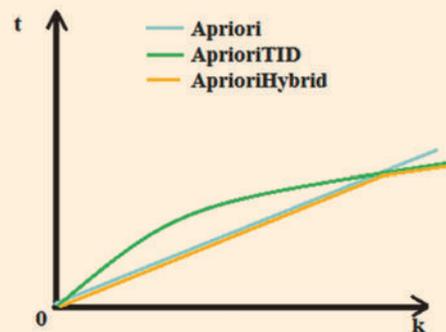
Из каждого множества $\{a, b\}$ можно получить два правила: $a \rightarrow b$ или же $b \rightarrow a$, причем полученные правила могут иметь разную достоверность, так как:

$$\text{conf}(A_i) = \frac{\text{sup}(X \rightarrow Y)}{\text{sup}(X)},$$

где A_i — это правило.

Затем происходит анализ полученных правил с точки зрения целей применения алгоритма.

Алгоритм Apriori, зарекомендовавший себя как отличный инструмент поиска ассоциативных правил, постоянно модифицируется учеными и энтузиастами. К разновидностям данного алгоритма относят AprioriTID и AprioriHybrid. В первом из них оптимизировано обращение к базе данных: подсчет поддержки происходит только на первом шаге, затем используется кодирование кандидатов, что может занимать намного меньше ресурсов, чем вся база данных. Однако оказалось, что в ранних проходах Apriori показывает лучшие результаты, чем AprioriTID, поэтому был создан третий алгоритм, который сочетает в себе эти два, — AprioriHybrid.



Увеличить быстродействие алгоритма Apriori возможно следующими методами [1]:

- составление хэшированных таблиц транзакций T , позволяющих сразу удалить точно не часто встречающиеся k -элементные кандидаты по подсчету значения хэша (в сравнении с пороговым значением);
- сразу удалять ту часть БД, где не может быть часто встречающихся элементов;
- способ разделения полноты: разбивать базу на несколько небольших разделов, проверять части, затем — полученные из разделов результаты;

• динамический подсчет элементов: добавлять новый кандидат только в случае, если он является чьим-то подмножеством дальше (во всех подмножествах).

Для реализации алгоритма были выбраны в качестве промежуточных (подготовительных) шагов реализации без технических средств и с помощью электронных таблиц, а конечный результат — это реализация алгоритма для учебных задач на языке программирования C++. Данный язык был выбран как широко используемый вне школы, входящий в тройку лидеров в рейтинге языков программирования среди специалистов (на апрель 2015 года), также разрешенный для использования при решении заданий ЕГЭ и олимпиад.

Для понимания и легкого запоминания первая реализация алгоритма Apriori была получена без использования современных технических средств. Приведенный ниже порядок действий (практикум) используется для демонстрации работы алгоритма и записывается в тетрадях и/или на доске. Следовательно, автоматизация процесса не производится.

1. Выписать множество I (itemset):

ID	PrName
...	...

2. Выписать множество T (множество всех транзакций):

TID	Items
...	ID1, ID2, ...

ID и TID упорядочены по возрастанию.

3. Составить нормализованную таблицу, где строка соответствует TID, столбец — ID. Наличие товара в конкретной транзакции t отмечается в бинарном виде: 0 — товар в чеке отсутствует, 1 — товар в чеке присутствует.

$$\delta(t) = \begin{cases} 0, & \text{если товар отсутствует} \\ 1, & \text{если товар в чеке присутствует} \end{cases}$$

Получить матрицу вида:

$$\begin{pmatrix} 0 & \dots & 1 \\ \dots & \dots & \dots \\ 1 & \dots & 1 \end{pmatrix}$$

Перейти к первому этапу выполнения алгоритма Apriori (выявление часто встречающихся элементов).

Следует руководствоваться правилом:

“Любой k -элементный набор будет часто встречающимся, если все его $(k - 1)$ -элементные подмножества будут часто встречающимися”.

То есть пятиэлементный набор будет часто встречающимся, только если все четырехэлементные наборы в нем часто встречались.

4. Для каждого столбца подсчитать значение поддержки товара как сумму элементов столбцов матрицы, так как для первого шага $Y = \emptyset$.

$$\text{sup}(ID) = \sum X \cup Y$$

На первом этапе алгоритма (сейчас) кандидатами C_k являются все одноэлементные множества, т.е. все товары.

5. Сравнить полученные значения поддержки с объявленным в условии минимальным значением поддержки minsup (например, $\text{minsup} = 3$).

Элементы, у которых поддержка больше или равна минимальной, заносятся во множество часто встречающихся F_k .

Во множество часто встречающихся заносятся сами элементы, а не значения их поддержки.

6. Кандидатами C_k считать множество всех сочетаний разных элементов F_k из предыдущего шага.

7. Подсчитать поддержку каждого такого множества по таблице.

8. Повторить пункты 5–7 до тех пор, пока $F_k \neq \emptyset$.

9. Выписать все полученные k -элементные часто встречающиеся наборы, где множество

$$F = \sum_{i=1}^k F_i.$$

Перейти ко второму этапу алгоритма Apriori (составление ассоциативных правил).

Ассоциативные правила — это правила отношений элементов друг к другу, иначе — правила, по которым группируют данные.

10. Достоверность (confidence) правила показывает, с какой вероятностью из первого условия следует второе, т.е. правило $X \rightarrow Y$ имеет достоверность conf , если t транзакций из T, содержащих X, также содержат Y.

Обычно достоверность переводят в процентное значение

$$\text{conf}(A_i) = \frac{\text{sup}(X \rightarrow Y)}{\text{sup}(X)},$$

где A_i — это правило.

Очевидные правила имеют достоверность выше 60%. Внимание заслуживают правила с поддержкой в 5–10%, охватывающие перспективные решения.

Пример:

Товары I:

ID	PrName
1	мясо
2	хлеб
3	соль
4	сахар
5	мука
6	спички

Транзакции T:

TID	Items
101	мясо, хлеб
102	хлеб, соль, сахар
103	мука
104	спички, хлеб, мясо
105	хлеб, спички
106	мясо, мука
107	соль, сахар
108	мясо, хлеб, соль, спички

$$\delta(t) = \begin{cases} 0, & \text{если товар отсутствует} \\ 1, & \text{если товар в чеке присутствует} \end{cases}$$

		ID					
		1	2	3	4	5	6
TID	101	1	1	0	0	0	0
	102	0	1	1	1	0	0
	103	0	0	0	0	1	0
	104	1	1	0	0	0	1
	105	0	1	0	0	0	1
	106	1	0	0	0	1	0
	107	0	0	1	1	0	0
	108	1	1	1	0	0	1

minsup = 3

1-й шаг:

C_k	1	2	3	4	5	6
support	4	5	3	2	2	3
F_k	1	2	3	-	-	6

2-й шаг:

C_k	1, 2	1, 3	1, 6	2, 3	2, 6	3, 6
support	3	1	2	2	3	1
F_k	1, 2	-	-	-	2, 6	-

3-й шаг:

C_k	1, 2, 6
support	2
F_k	-

$F = \{1\}, \{2\}, \{3\}, \{6\}, \{1,2\}, \{2,6\}$

Ассоциативные правила:

№	Правило	Достоверность, %
A1	Если покупатель взял мясо, то он купит хлеб	75
A2	Если покупатель взял хлеб, то он купит спички	60
A3	Если покупатель взял хлеб, то он купит мясо	60
A4	Если покупатель взял спички, то он купит хлеб	100

$$conf(A1) = \frac{sup(\{1,2\})}{sup(\{1\})} = \frac{3}{4} = 75\%$$

$$conf(A2) = \frac{sup(\{2,6\})}{sup(\{2\})} = \frac{3}{5} = 60\%$$

$$conf(A3) = \frac{sup(\{2,1\})}{sup(\{2\})} = \frac{3}{5} = 60\%$$

$$conf(A4) = \frac{sup(\{6,2\})}{sup(\{6\})} = \frac{3}{3} = 100\%$$

Примечание. Из данной выборки следовало, что покупатель всегда берет хлеб, если он купил спички, что не будет действовать на выборках с большим количеством входящих данных. Чем больше транзакций и товаров рассматривается, тем результат надежнее.

Подготовительным этапом к реализации алгоритма Apriori на языке программирования явля-

ется реализация частного случая средствами программы Microsoft Excel, так как табличный процессор позволяет частично автоматизировать процесс.

Для этого предлагается следующая последовательность действий, оформленная в виде практикума для обучающихся.

1. Создать новую книгу в Excel.

2. В ячейке A1 ввести название будущей таблицы — **Itemset**, заполнить A2 как **ID** (идентификационный номер) и B2 как **PrName** (название товара). Начиная с B3 заполнить таблицу видами товаров, которые продаются в магазине. Для ячеек в столбце A использовать числовой формат с числом десятичных знаков 0, для ячеек столбца B — текстовый. Заполнить таблицу и оформить ее.

Книга1 * x			
	A	B	C
1	Itemset		
2	ID	PrName	
3		1 мясо	
4		2 хлеб	
5		3 соль	
6		4 сахар	
7		5 мука	
8		6 спички	

3. В D1 указать название таблицы **Transactions**. Назвать будущие столбцы: **TID** (номер транзакции) в D2, **Items** (входящие товары) в E2. Номера транзакций последовательны, значит, их можно задать автоматически. Заполнить таблицу и оформить ее.

Книга1 * x					
	A	B	C	D	E
1	Itemset		Transaction		
2	ID	PrName	TID	Items	
3		1 мясо	101	мясо, хлеб	
4		2 хлеб	102	хлеб, соль, сахар	
5		3 соль	103	мука	
6		4 сахар	104	спички, хлеб, мясо	
7		5 мука	105	хлеб, спички	
8		6 спички	106	мясо, мука	
9			107	соль, сахар	
10			108	мясо, хлеб, соль, спички	
11					

4. В ячейке G2 ввести **minsup=**, в ячейку H2 ввести значение минимальной поддержки.

F	G	H	I
	minsup=	3	

ПРОФИЛЬ

5. Составить нормализованную таблицу, где строка соответствует TID, столбец — ID. Наличие товара в конкретной транзакции t отмечается в бинарном виде: 0 — товар в чеке отсутствует, 1 — товар в чеке присутствует (формула 1):

$$\delta(t) = \begin{cases} 0, & \text{если товар отсутствует} \\ 1, & \text{если товар в чеке присутствует} \end{cases} \quad \text{формула 1}$$

Задать названия столбцов и полей.

5а. Объединить ячейки I3:I10, вертикальным текстом заполнить **TID**. В ячейки J3:J10 ввести номера транзакций (скопируйте и вставьте). Объединить ячейки K1:P1 и записать в них **ID**. K2:P2 заполнить номерами товаров. Оформить таблицу.

I	J	K	L	M	N	O	P	Q
		ID						
		1	2	3	4	5	6	
T I D	101							
	102							
	103							
	104							
	105							
	106							
	107							
	108							

5б. Заполнить получившуюся таблицу согласно формуле 1. Для этого в ячейку K3 ввести формулу:

=ЕСЛИОШИБКА(ЕСЛИ(ПОИСК(\$B\$3;\$E3;1)>0;1;0);0)

5в. Внести изменения в формулу для других столбцов и заполнить таблицу полностью.

Transaction		ID					
TID	Items	1	2	3	4	5	6
101	мясо, хлеб	1	1	0	0	0	0
102	хлеб, соль, сахар	0	1	1	1	0	0
103	мука	0	0	0	0	1	0
104	спички, хлеб, мясо	1	1	0	0	0	1
105	хлеб, спички	0	1	0	0	0	1
106	мясо, мука	1	0	0	0	1	0
107	соль, сахар	0	0	1	1	0	0
108	мясо, хлеб, соль, спички	1	1	1	0	0	1

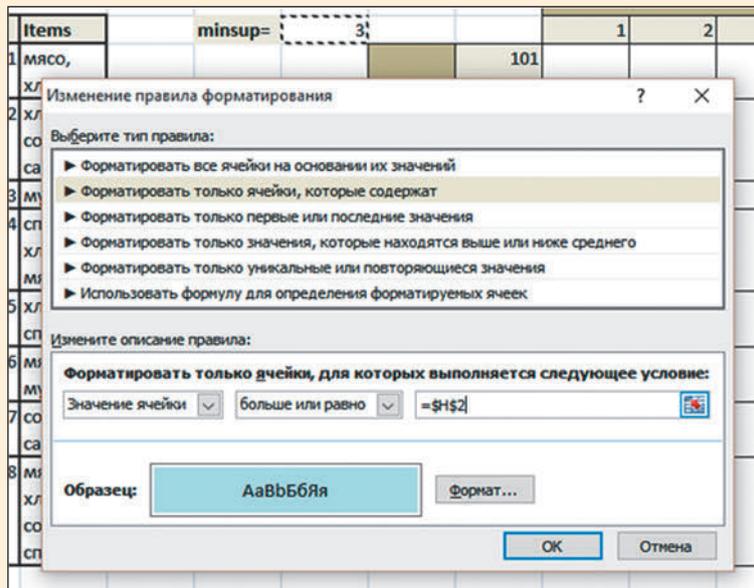
6. Выполнить первый шаг алгоритма.

6а. Ячейку A12 заполнить как C1, в B12:G12 ввести кандидаты (на данном шаге — все ID).

6б. Ячейку A13 заполнить как support. Подсчитать поддержку первого кандидата по формуле

=СУММ(K3:K10)

Подсчитать поддержку для остальных кандидатов. Выполнить условное форматирование:



Пусть ячейка будет выделена, если ее значение больше или равно минимальной поддержке.

6в. Ячейку A14 назвать **F1**. Заполнить значениями кандидатов с удовлетворяющей поддержкой. Для этого в B14 записать формулу:

=ЕСЛИ(B\$13>=\$H\$2; B\$12; "нет")

Заполнить значениями C14:G14.

6г. Оформить таблицу.

11							
12	C1	1	2	3	4	5	6
13	support	4	5	3	2	2	3
14	F1	1	2	3	=ЕСЛИ(Е\$13>=\$H\$2; Е\$12; "нет")		
15					=ЕСЛИ(лог_выражение; [значение_если_исти		

7. Выполнить второй шаг алгоритма.

7а. В A16 ввести обозначение как **C2** и заполнить строку кандидатами — двухэлементными множествами, составленными как сочетания элементов из F1.

7б. В ячейку A17 занести **support**. Для расчета понадобятся промежуточные вычисления. Для этого в ячейку R2 записать следующую формулу:

=ЕСЛИ((\$K3=1)*И(\$L3=1);1;0)

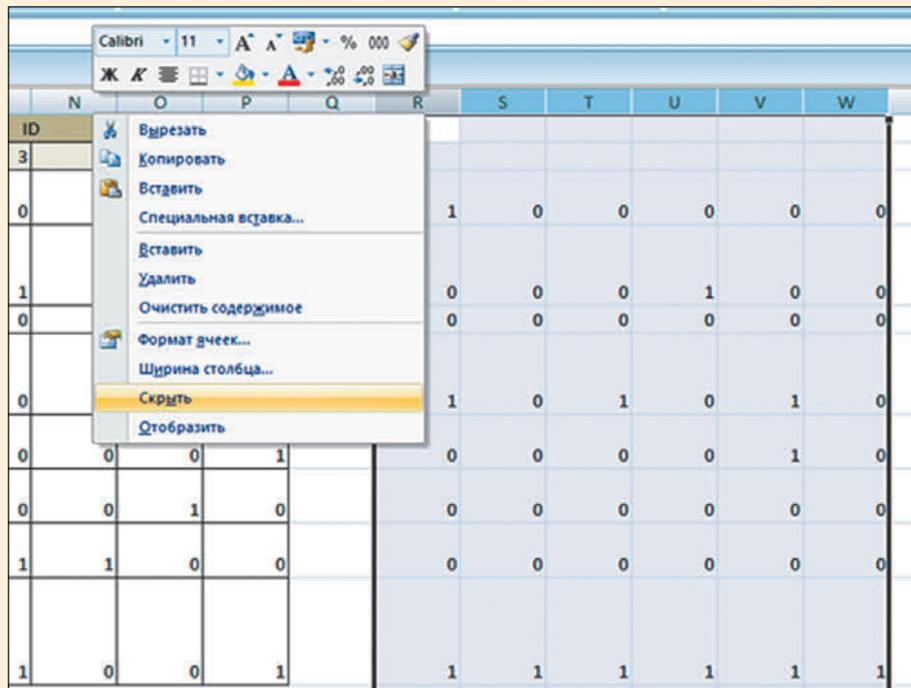
Заполнить R3:R10, используя данную формулу. Изменить ее и произвести проверку для остальных пар в ячейках S3:W10.

=ЕСЛИ((\$M10=1)*И(\$P10=1);1;0)																					
					ID																
	minsup=	3			1	2	3	4	5	6											
	T I D	101			1	1	0	0	0	0		1	0	0	0	0	0	0			
		102			0	1	1	1	0	0		0	0	0	1	0	0				
		103			0	0	0	0	0	1	0		0	0	0	0	0	0			
		104			1	1	0	0	0	0	1		1	0	1	0	1	0			
		105			0	1	0	0	0	0	1		0	0	0	0	1	0			
		106			1	0	0	0	1	0			0	0	0	0	0	0			
		107			0	0	1	1	0	0			0	0	0	0	0	0			
		108																			
					1	1	1	0	0	1		1	1	1	1	1	1				

Тогда значение в B17 равно:

=СУММ(R\$3:R\$10)

Подсчитать поддержку для остальных пар. Скрыть промежуточные вычисления.



Использовать условное форматирование, как в пункте бб.

7в. Ячейку A18 назвать F2 и заполнить строку по аналогии с пунктом бв.

7г. Оформить таблицу.

8. Выполнить третий шаг алгоритма. Так как номер шага равен трем, необходимо найти трехэлементное множество. Выполнить пункт самостоятельно, для дополнительных расчетов использовать X3:X10.

9. Перейти к следующему этапу алгоритма.

Назвать ячейку A24 как F и выписать все полученные значения Fk.

23							
24	F	1	2	3	6,2	2,6	
25							

10. Составить таблицу ассоциативных правил. Для этого записать № в ячейку I12, объединить J12:N12 и записать в полученную ячейку **Правило**, в объединенные O12:P12 внести **Достоверность, %**.

Правила, полученные из одноэлементных множеств, не представляют интереса. Из каждого двухэлементного множества можно составить два правила. Выписать их.

		1	1	1	0	0	1
№	Правило	Достоверность, %					
1	Если покупатель взял мясо, то он купит хлеб.						
2	Если покупатель взял хлеб, то он купит мясо.						
3	Если покупатель взял хлеб, то он купит спички.						
4	Если покупатель взял спички, то он купит хлеб.						

11. Заполнить столбец достоверности, руководствуясь формулой:

$$\text{conf}(A_i) = \frac{\sup(X \rightarrow Y)}{\sup(X)} * 100\%$$

где A_i — это правило.

11а. Рассчитать достоверность первого правила в ячейке O13 по формуле:

$$=B17/B13*100, \text{ где}$$

$$\text{conf}(A1) = \frac{\sup(\{1,2\})}{\sup(\{1\})} * 100\%$$

116. Аналогично подсчитать достоверность остальных полученных правил. Оформить таблицу.

fx =F17/G13*100												
D	E	F	G	H	I	J	K	L	M	N	O	P
108	мясо, хлеб, соль, спички					108						
							1	1	1	0	0	1
3	4	5	6									
3	2	2	3									
3	нет	нет	6									
1, 6	2, 3	2, 6	3, 6									
2	2	3	1									
№	Правило											Достоверность, %
1	Если покупатель взял мясо, то он купит хлеб.											75
2	Если покупатель взял хлеб, то он купит мясо.											60
3	Если покупатель взял хлеб, то он купит спички.											60
4	Если покупатель взял спички, то он купит хлеб.											100

12. В ячейку G3 записать **minconf**, а в H3 внести значение минимальной достоверности. Пусть в данной задаче она равна 70%.

Используя условное форматирование, выделить те ассоциативные правила, которые удовлетворяют условию.

№	Правило	Достоверность, %
1	Если покупатель взял мясо, то он купит хлеб.	75
2	Если покупатель взял хлеб, то он купит мясо.	60
3	Если покупатель взял хлеб, то он купит спички.	60
4	Если покупатель взял спички, то он купит хлеб.	100

13. Сохранить файл под названием **Apriori.xlsx**.

Существует несколько решений для реализации алгоритма на языке C++. Однако подобные решения не являются достаточно демонстративными, так как привязаны к особенностям языка программирования. Например, с использованием векторов (аналог динамического массива). В этом случае идет частое обращение к функциям и конструкторам, описанным в подключаемых библиотеках `<vector>` и `<algorithm>`, и программы не могут быть перенесены на другой язык с сохранением аналогичного количества строк кода и сложности.

Были созданы и проанализированы различные решения. Наиболее наглядным и доступным для обучающихся старшей школы оказалось решение, использующее одномерные массивы и работу со счетчиками. Здесь не используются дополнительные библиотеки, кроме `<cstdlib>` для удобной остановки работы консольного приложения и `<fstream>` для вывода данных в созданный текстовый файл для дальнейших операций над ним.

```
#include <iostream>
#include <stdio.h>
#include <cstdlib>
#include <fstream>
```

Будут часто использоваться стандартные имена, поэтому целесообразно подключить все пространство этих имен:

```
using namespace std;
```

Подключим для удобства работы с будущим приложением русскую раскладку, изменим фон консоли на белый, цвет шрифта на черный и название консоли на название проекта:

```
int main() {
    setlocale(LC_ALL, "Russian");
    system("color 70");
    system("title Apriori");
```

Необходимо создать файл и настроить в него вывод. Аналогичная функция есть и в языке Pascal:

```
ofstream fout("AprioriResult.txt");
```

Используем сразу упорядоченную матрицу, где 0 — *i*-го товара нет, 1 — есть в чеке. Строки эквивалентны транзакциям. При необходимости можно реализовать через структуру отдельную таблицу с соответствиями номеров продуктов и названий, а сюда передавать только ID продуктов.

```
cout << "***APRIORI***" << endl << "Примечание: Для учебных задач обычно
хватает трех шагов." << endl << "Введите количество транзакций:" << endl;
int TransCount; //Кол-во транзакций
cin >> TransCount;
cout << "Введите количество товаров:" << endl;
int PrCount; //Кол-во товаров
cin >> PrCount;
```

```
fout << "Было " << TransCount << " транзакций с " << PrCount
    << " разными товарами." << endl; //Вывели в файл.
int T[TransCount][PrCount]; //Матрица наличия
cout << "Внесите данные о транзакциях, используя 0 для отсутствия товара,
    1 для наличия." << endl;
for (int i = 0; i < TransCount; i++){
    cout << "Введите позиции для чека №" << i + 1 << " :" << endl;
    fout << "Чек №" << i + 1 << ":" << endl;
    for (int j = 0; j < PrCount; j++){ cin >> T[i][j];
        fout << T[i][j] << " "; //Заполняются "почеково".
    }
    fout << endl;
};
int temp = 0; //Временная переменная
```

Временная переменная пригодится позднее. Теперь необходимо ввести значение минимальной поддержки и записать в текстовый файл, соответствующе оформив:

```
cout << "Введите значение минимальной поддержки:" << endl;
int minsup; //мин.поддержка.
cin >> minsup; //Ввели значение минимальной поддержки.
fout << "Минимальная поддержка равна " << minsup << "." << endl;
//Выписываем в файл значение минимальной поддержки.
cout << endl;
```

Так как матрица уже нормализована, можно перейти к выполнению первого шага алгоритма и поиску часто встречающихся одноэлементных множеств, $k = 1$.

```
/*Первый шаг.*/
int C1[PrCount]; /*Кандидаты — все одноэлементные.*/
for (int i = 0; i < PrCount; i++) { C1[i] = i + 1; } //Заполнили одноэлементными.
```

На первом шаге кандидатами автоматически становятся все одноэлементные множества, т.е. все товары, которых было PrCount штук.

```
int sup1[PrCount]; //Для каждого выделим место под поддержку.
for (int i = 0; i < PrCount; i++){ sup1[i] = 0; } //Очистили от мусора.
```

Поддержка конкретного множества будет увеличиваться, если 1 встречается в столбце при проверке конкретного чека (строки).

```
for (int i = 0; i < PrCount; i++){ //Проверяем по товарам
    for (int j = 0; j < TransCount; j++){
        if (T[j][i] == 1) sup1[i]++; }
    };
```

Apriori — масштабируемый алгоритм, и это значит, что кандидаты, которые не подошли под условие минимальной поддержки, будут удаляться, а нумерация — сокращаться. Следовательно, необходимо сохранить номера товаров, которые становятся часто встречающимися (их ID):

```
int num[PrCount]; //сохраним номера столбцов для будущего расчета.
for (int i = 0; i < PrCount; i++){ num[i] = 0; }; //Сохраним потом номера всех кандидатов.
```

Выведем полученные значения поддержки:

```
cout << "Поддержка на первом шаге у элементов:" << endl;
fout << "Поддержка на первом шаге у элементов:" << endl;
for (int i = 0; i < PrCount; i++){ cout << "{" << C1[i] << "
    sup=" << sup1[i] << endl; fout << "{" << C1[i] << "
    sup=" << sup1[i] << endl; }; //Выведем мн-ва с k = 1 и их поддержку.
```

Подсчет количества будущих часто встречающихся множеств нужен для выделения достаточной, но не слишком избыточной памяти для статического массива:

```
for (int i = 0; i < PrCount; i++){ if (sup1[i] >= minsup) temp++; }; //Подсчитываем
количество будущих часто встречающихся мн-в.
int F1[temp]; //Выделяем память под часто встречающиеся мн-ва размерности k = 1.
cout << "Удовлетворяющих элементов: " << temp << endl;
fout << "Удовлетворяющих элементов: " << temp << endl;
```

Если нет часто встречающихся множеств, выполнение программы можно приостановить и вывести сообщение о результатах, прекратить запись в файл:

```
if (temp == 0) {
    cout << "Больше часто встречающихся множеств не найдено." << endl;
    cout << "Результаты записаны в файл AprioriResult.txt." << endl;
    fout.close(); //Закрыли файл.
    system("pause");
};
```

Переходим к работе с часто встречающимися множествами при $k = 1$. Следует обратить внимание на сохранение номеров ID:

```

cout << "Эти элементы:" << endl;
fout << "Эти элементы:" << endl; //вывод в файл
temp = 0; //Сбросим счетчик.
for (int i = 0; i < PrCount; i++){
    if (sup1[i] >= minsup) { //Найдем эти часто встречающиеся множества.
        F1[temp] = C1[i]; //Если условие (поддержка больше или равна
            // минимальной) выполнено, то заносим в память.
        num[temp] = i; // Сохраняем номер столбца из таблицы для данного элемента.
        cout << " {" << F1[temp] << "}, "; //Выводим на экран.
        fout << " {" << F1[temp] << "}, "; //Выводим в файл.
        temp++;
    } //Увеличиваем счетчик для часто встречающихся множеств.
}
cout << endl;

```

Когда все часто встречающиеся множества размера $k = 1$ найдены, производится переход ко второму шагу алгоритма.

```

cout << "Перейдем ко второму шагу, кандидаты: ";
fout << endl << "Кандидаты на втором шаге:";

```

Мы будем реализовывать через одномерный массив, но нам нужны двухэлементные множества. Значит, используем пары соседних ячеек:

```

int temp2=temp*2;
int C2[temp2]; //Выделим память для описанного выше массива кандидатов.
int sup2[temp]; // Для пар нам понадобится подсчитать поддержку.
    // Для пар, а не элементов пар!
for (int i = 0; i < 100; i++){ sup2[i] = 0; }
temp2 = 0; // Временные переменные.
int temp3 = 0;
for (int i = 0; i < temp - 1; i++) { //Мы рассматриваем пары различных элементов,
    // поэтому нам не надо смотреть последний элемент массива.
    for (int j = i + 1; j < temp; j++){ //Так как пары различных элементов,
        //на второй позиции не смотрим первый
        //элемент и т.д.

```

Для создания кандидатов нужны различные множества, чтобы не было, например, {2,3} и {3,2}, поэтому введем следующее условие:

```

if (F1[i] < F1[j]){

```

Переводим в первый элемент кандидата часто встречающееся $(k - 1)$ -элементное множество:

```

C2[temp2] = F1[i];
C2[temp2 + 1] = F1[j]; //Аналогично поступаем со второй позицией в кандидате.
//Выводим получившееся 2-элементное множество.
cout << endl << " {" << C2[temp2] << "," << C2[temp2 + 1] << "}" ;
fout << endl << " {" << C2[temp2] << "," << C2[temp2 + 1] << "}" ;

```

Сразу подсчитаем поддержку. Проверяем все чеки:

```

for (int z = 0; z < TransCount; z++){
    //Используем запомненные номера столбиков.
    if (T[z][num[i]] == 1 && T[z][num[j]] == 1){
        sup2[temp3]++;
    } // Если в обоих столбиках 1, то увеличиваем поддержку.
}
//Выводим поддержку для данного 2-элементного множества.
cout << " sup=" << sup2[temp3];
fout << " sup=" << sup2[temp3];

```

Следует обратить внимание, с чем производились операции. Чтобы продолжить работать со следующей парой, т.е. следующими двумя позициями, надо увеличить на 2, а не на 1:

```

temp2 = temp2 + 2;

```

Но поддержку считаем для пары, а не элемента:

```

temp3++;
}
}

```

Если не удалось составить множества из двух элементов (например, часто встречалось лишь одно множество, где $k = 1$), выведем соответствующее сообщение:

```

if (temp2 == 0){ cout << " не удалось составить множества." << endl;
    fout << " не удалось составить множества." << endl; }
temp = 0; //Временная переменная

```

Снова заранее подсчитаем, сколько выделить места в памяти:

```

//Подсчитываем количество часто встречающихся для k = 2.
for (int i = 0; i <= temp3; i++){ if (sup2[i] >= minsup) temp++; }

```

```
cout << endl<< "Удовлетворяющих элементов: " << temp << endl;
fout << endl<< "Удовлетворяющих элементов: " << temp << endl;
```

Повторяем аналогично выполнению первого шага (сообщение об ошибке):

```
if (temp == 0){
    cout << "Больше часто встречающихся множеств не найдено." << endl;
    cout << "Результаты записаны в файл AprioriResult.txt." << endl;
    fout.close(); //Закрыли файл.
    system("pause");
}
else{
    cout << "Эти элементы:" << endl;
    fout << "Эти элементы:" << endl;
    temp3 = 0;
    int temp4 = 0;
```

Здесь temp2 отлично от прошлой размерности массива, оно получило новое значение:

```
int F2[temp2]; // при работе с кандидатами.
// Отводит место под часто встречающиеся пары.
```

Нам снова придется запоминать номера столбцов, где одна пара — один номер. Использование данной логической операции равноценно умножению на два в первой степени:

```
int num2[temp2 << 1]; // использован побитовый сдвиг.
```

“Шагаем” через пару элементов, так как в C2 они тоже расположены парно:

```
for (int i = 0; i < temp2; i = i + 2){
```

Если поддержка пары больше или равна минимальной, то сохраняем первую позицию, номер столбца для этого элемента, вторую позицию в паре и номер столбца для второго элемента:

```
if (sup2[temp3] >= minsup) {
    F2[temp4] = C2[i];
    num2[temp4] = int(i/2);
    F2[temp4 + 1] = C2[i + 1];
    num2[temp4 + 1] = int(i/2) + 1;
    //Выводим на экран полученные двухэлементные множества.
    cout << " {" << F2[temp4] << "," << F2[temp4 + 1] << "}" << endl;
    fout << " {" << F2[temp4] << "," << F2[temp4 + 1] << "}" << endl;
    temp4 = temp4 + 2; //Увеличиваем на два, так как заполняли парой.
}
temp3++; //На один, так как у пары одно значение поддержки, а не два.
}
```

Переходим к третьему шагу алгоритма.

```
/*Третий шаг*/
cout << "Перейдем к третьему шагу, кандидаты: ";
fout << "Кандидаты на третьем шаге:" << endl;
```

Теперь $k = 3$, т.е. будут рассматриваться тройки товаров, поэтому для отведения памяти количество двухэлементных множеств, которые станут здесь кандидатами, умножили втрое. На самом деле работать будет в основном количестве случаев (достаточно для учебных задач), но не всегда, так как точное решение связано с комбинаторикой, реализуется дополнительными функциями, которые пишутся самостоятельно (в отличие от встроенных функций из комбинаторики в языке R):

```
int C3[temp3*3];
int sup3[temp3];
```

Этого количества для sup3[] должно хватить для большинства случаев. Мы не оставляем константу, чтобы не использовать больше памяти, чем потребуется. С другой стороны, мы ограничиваем себя в решениях.

```
for (int i = 0; i < 100; i++){ sup3[i] = 0; } // Очищаем от "мусора".
temp2 = 0;
```

Отсортируем массив, чтобы исключить повторения во множествах в дальнейшем. Применим обычную сортировку “пузырьком”:

```
int tmp = 0;
for (int i = 0; i < temp4 - 1; ++i) {
    for (int j = 0; j < temp4 - 1; ++j)
    {
        if (C2[j + 1] < C2[j]) //Если элемент меньше соседа, меняем их местами.
        {
            //Это одна из причин, почему взят одномерный массив, — удобство работы.
            tmp = C2[j + 1];
            C2[j + 1] = C2[j]; // Мы используем самые простые действия и операторы.
            C2[j] = tmp;
        }
    }
}
```

Перейдем к формированию троек кандидатов. Аналогично со вторым шагом. Мы рассматриваем различные тройки, поэтому на первой позиции никогда не будет два последних кандидата и т.д. Используем сразу три ячейки и счетчика для них:

```
temp3 = 0;
tmp = 0;
bool flag = false; //Он понадобится всего лишь один раз, чтобы отделить первый проход.
for (int i = 0; i < temp4 - 2; i++) {
    for (int j = i + 1; j < temp4 - 1; j++){
        for (int g = i + 2; g < temp4; g++){
```

Будем рассматривать только те тройки, где элементы в порядке возрастания, чтобы все тройки были различными:

```
if ((F2[i] < F2[j]) && (F2[j] < F2[g])){
```

Если проход не первый и сумма элементов предыдущей тройки равна сумме элементов текущей, то “прыгаем” на метку. Так как у нас идут по возрастанию, то суммы различных троек будут отличаться хотя бы на 1. При учете суммы нас не интересует положение чисел внутри множеств:

```
if ((flag == true) && tmp == F2[i] + F2[j] + F2[g]){ goto rep; }
C3[temp2] = F2[i]; //Сохранили первый элемент тройки, выбрав число
// из множества из часто встречающихся 2-элементных множеств.
C3[temp2 + 1] = F2[j]; // Сохранили второй элемент тройки.
C3[temp2 + 2] = F2[g]; // Аналогично поступили с третьим элементом.
tmp = F2[i] + F2[j] + F2[g]; //Подсчитали сумму, вес всей тройки.
cout << endl << " {" << C3[temp2] << ", " << C3[temp2 + 1] << ", "
<< C3[temp2 + 2] << " }";
// Вывели полученное трехэлементное множество на экран.
fout << endl << " {" << C3[temp2] << ", " << C3[temp2 + 1] << ", "
<< C3[temp2 + 2] << " }";
```

Подсчитаем поддержку для данного множества, $k = 3$:

```
for (int z = 0; z < TransCount; z++){
    if (T[z][num2[i]] == 1 && T[z][num2[j]] == 1 && T[z][num2[g]] == 1){
```

Используя сохраненные номера столбцов, смотрим соответствующие значения в каждом чеке. Если все три 1, то увеличиваем значение поддержки множества:

```
sup3[temp3]++;
};
}
cout << " sup=" << sup3[temp3]; // Вывели окончательно значение поддержки
// для данной тройки.
fout << " sup=" << sup3[temp3];
flag = true; //При первом проходе флаг меняется здесь с false на true.
temp2 = temp2 + 3; // Метка. Увеличиваем индекс ячеек, чтобы перейти к следующей
// тройке(через два элемента от первого в данной тройке).
temp3++; //Подсчитываем количество троек. Увеличиваем на один.
}
}
}
}
}
```

Поступаем аналогично первым двум шагам:

```
if (temp2 == 0){ cout << " не удалось составить множества." << endl;
    fout << " не удалось составить множества." << endl; };
temp = 0;
for (int i = 0; i <= temp3; i++){ //Меньше или равно, так как индексы
// начинались с нуля, а место в массиве – от 1.
    if (sup3[i] >= minsup) temp++;
} // Подсчитываем, сколько выделить памяти под часто встречающиеся
// трехэлементные мн-ва.
cout << endl << "Удовлетворяющих элементов: " << temp << endl;
fout << endl << "Удовлетворяющих элементов: " << temp << endl;
if (temp == 0){
    cout << "Больше часто встречающихся множеств не найдено." << endl;
    cout << "Результаты записаны в файл AprioriResult.txt." << endl;
    fout.close(); //Закрыли файл.
    system("pause");
}
else{ //Сообщение об ошибке и пауза, иначе ищем эти элементы.
    cout << "Эти элементы:" << endl;
    fout << "Эти элементы:" << endl;
```


Кроме вывода в консоль, был реализован вывод в текстовый файл AprioriResult.txt. Для приведенных входящих данных содержимое текстового файла будет следующим:

Было восемь транзакций с шестью разными товарами.

Чек № 1:

1 1 0 0 0

Чек № 2:

0 1 1 1 0

Чек № 3:

0 0 0 1 0

Чек № 4:

1 1 0 0 1

Чек № 5:

0 1 0 0 1

Чек № 6:

1 0 0 1 0

Чек № 7:

0 0 1 1 0

Чек № 8:

1 1 1 0 1

Минимальная поддержка равна 3.

Поддержка на первом шаге у элементов:

{1} sup=4

{2} sup=5

{3} sup=3

{4} sup=2

{5} sup=2

{6} sup=3

Удовлетворяющих элементов: 4

Эти элементы:

{1}, {2}, {3}, {6}

Кандидаты на втором шаге:

{1,2} sup=3

{1,3} sup=1

{1,6} sup=2

{2,3} sup=2

{2,6} sup=3

{3,6} sup=1

Удовлетворяющих элементов: 2

Эти элементы:

{1,2}

{2,6}

Кандидаты на третьем шаге:

{1,2,6} sup=2

Удовлетворяющих элементов: 0

вывод:

Список использованной литературы

1. Anita Wasilewska. "APRIORI Algorithm", Stony Brook University. [Электронный ресурс]. Режим доступа: http://www3.cs.stonybrook.edu/~cse634/lecture_notes/07apriori.pdf.

2. Jian Pei, Jiawei Han, Micheline Kamber. "Data Mining, Second Edition : Concepts and Techniques", 2nd Edition — U.S.A., The University of Illinois at Urbana-Champaign: "Morgan Kaufmann", January 2006, 722 p.

3. Surya P.N. Singh and Scott M. Thayer, "ARMS: AUTONOMOUS UTONOMOUS ROBOTS FOR OBOTS FOR MILITARY ILITARY SYSTEMS A Survey of Collaborative Robotics Core Technologies and Their Military Applications", The Robotics Institute Carnegie Mellon University, Pittsburgh, Pennsylvania. [Электронный ресурс]. Режим доступа: https://www.ri.cmu.edu/pub_files/pub3/singh_surya_2001_2/singh_surya_2001_2.pdf.

4. Wenjing Zhang, Donglai Ma, Wei Yao. "Medical Diagnosis Data Mining Based on Improved Apriori Algorithm", Journal of Networks, Vol 9, No 5 (2014), 1339–1345, May 2014.

5. Калинин И.А., Самылкина Н.Н. Информатика. Углубленный уровень: учебник для 11-го класса. М.: БИНОМ, 2013.

6. Краковецкий Александр (CEO DevRain Solutions), "Анализ рыночной корзины и ассоциативные правила". [Электронный ресурс]. Режим доступа: <http://habrahabr.ru/post/66016/>.

7. Свободная энциклопедия Wikipedia, статья об Apriori. [Электронный ресурс]. Режим доступа: https://en.wikipedia.org/wiki/Apriori_algorithm.

8. Шахиду Акобур, "Apriori — масштабируемый алгоритм поиска ассоциативных правил". [Электронный ресурс]. Режим доступа: <https://basegroup.ru/community/articles/apriori>.



Имитационные модели работы полиграфического комплекса и аэропорта

О.К. Переслегина, учитель информатики ГБОУ ШМК Бибирево, Москва, olya.pereslegina@gmail.com

► В углубленном курсе информатики изучение имитационного моделирования происходит в среде AnyLogic [1, 2]. Среда позволяет реализовать достаточно объемные исследовательские задачи, актуальные и социально значимые для старшеклассников. Предлагаем познакомиться и практически реализовать две модели: первая — работы полиграфического комплекса; вторая — оптимизация работы аэропорта.

Модель 1. Работа полиграфического комплекса

Безусловно, что в настоящее время электронные книги вытесняют бумажные в количественном эквиваленте. Но вместе с тем традиционная бумажная книга становится качественнее, детские, учебные, специализированные

и подарочные издания стали иными и также востребованны.

Процесс выхода бумажной книги в свет проходит по следующей схеме:

1. Рукопись передается в издательство, где она либо принимается, либо нет. В случае положительного решения в издательстве начинается допечатная подготовка макета рукописи;

2. Стадия допечатной подготовки (ввод и обработка текстовой и изобразительной информации, верстка и спуск полос, изготовление печатных форм);

3. Стадия печатной подготовки (подготовка печатной машины, печатание тиража, контроль качества печати, сдача отпечатанных листов в переплетно-брошюровочный цех);

4. Стадия постпечатной обработки (фальцовка, биговка, листоподборка, скрепление, упаковка и погрузка).

Но, несмотря на максимальную автоматизацию данного процесса, зачастую отдельные экземпляры, а то и весь тираж попадают в брак (как по вине человека, отвечающего за тот или иной этап, так и по причине сбоя в технике).

Цель: Нам необходимо выяснить, сколько в среднем книг с тиража объемом в три тысячи экземпляров попадают в брак. При этом мы учитываем, что максимально допустимый процент брака для заданного тиража — 7%.

Для этого мы построим дискретно-событийную модель работы полиграфического комплекса.

Создадим чистую модель (Файл → Создать → Модель), на втором этапе выберем “с нуля” (рис. 1 и 2).

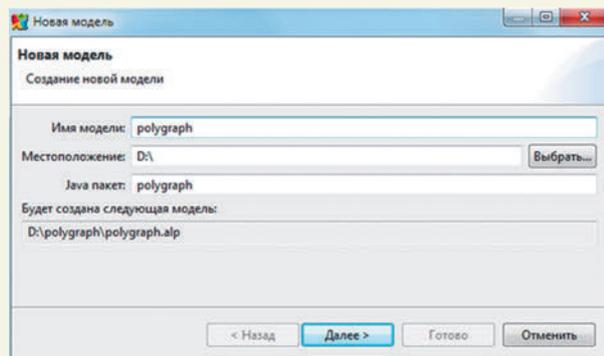


Рис. 1. Создание дискретно-событийной модели

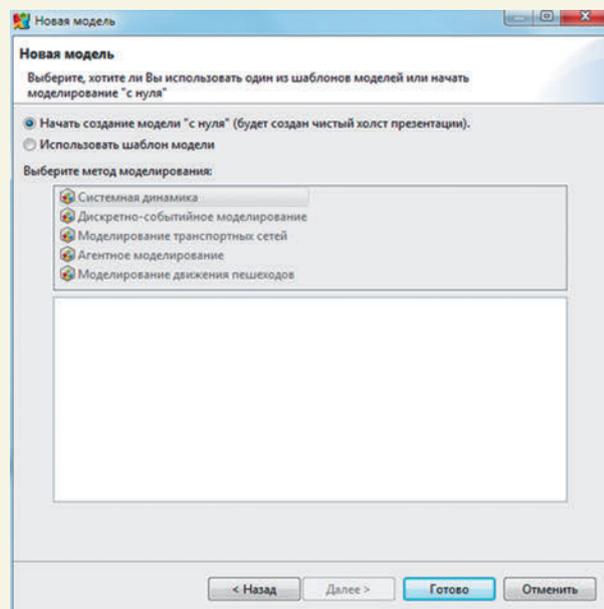


Рис. 2. Выбор пустого шаблона создания модели

Для дискретно-событийной модели воспользуемся объектами с закладки Основная библиотека (рис. 3).

Построение модели начинается с создания заявки: для этого из основной библиотеки перенесем объект **Source**. В нашей модели на основных этапах мы будем использовать объекты **Select Output** (обозначающий процесс, производимый на данном шаге) и **Sink** (Выход), а также объект **Split**.

Разместим объекты в рабочей области так, как показано на рис. 4.

Далее укажем параметры для каждого объекта. Для объекта **manuscriptSource** ограничим количество прибытий (рис. 5 на с. 20). Нам необходима одна заявка, попадающая на вход. В объекте **check** (Select Output) принимается решение: отправляется ли заявка (рукопись) в печать или же заявка признается негодной. Во втором случае происходит выход из модели, причем вероятность такого исхода очень мала. Поэтому в свойствах данного объекта в графе Вероятность мы указываем значение **0,99** (с такой вероятностью наша заявка пройдет дальше, и лишь с вероятностью **0,01** заявка покинет модель). Остальные

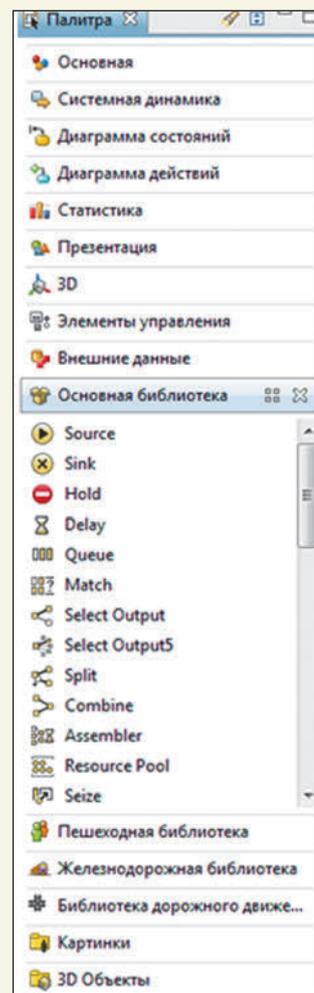


Рис. 3. Основная библиотека

параметры оставляем без изменений. Затем мы переходим к печати рукописи в количестве трех тысяч экземпляров. Вообще говоря, если рукопись проходит проверку, то дальше она переходит на стадию допечатной подготовки, но для нашей модели этот шаг неважен, поэтому мы его опускаем.

Печать тиража в нашей модели разбита на три объекта:

1. Объект **print1**. Здесь мы проверяем (на глобальном уровне), прошел ли тираж проверку качества: тираж напечатан качественно или тираж полностью признан бракованным. Выставляем для него параметры, как на рис. 6 на с. 20.

2. Объект **circulationSplit**. Здесь мы “множим” нашу заявку, для того чтобы далее можно было проверить отдельно каждый экземпляр, в отличие от

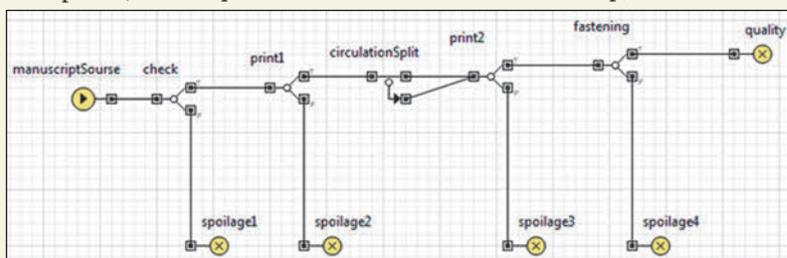


Рис. 4. Размещение объектов в рабочей области

Рис. 5. Параметры объекта Source

Рис. 6. Параметры объекта print1

предыдущего шага, где мы смотрели на тираж как на единое целое. Так как мы рассматриваем тираж в 3000 экземпляров, а у нас есть одна заявка, нам необходимо сделать 2999 копий (рис. 7).

Рис. 7. Параметры объекта Split

3. Объект **print2**. На данном объекте мы проверяем отдельно каждую копию на качество. В случае брака данный экземпляр выходит из модели. Здесь для задания вероятности мы используем функцию `uniform()` (рис. 8).

Затем, в объекте **fastening**, начинается стадия послепечатной подготовки. Здесь шанс, что напечатанный экземпляр будет испорчен, невелик (рис. 9). В случае, если экземпляр испорчен, он попадает на выход, в объект **spoilage4**.

И, наконец, если копия прошла все стадии, не попав в брак, она выходит из модели “естественным” способом. В свойствах объекта **quality** зададим Действие при входе (рис. 10). Этот параметр мы будем использовать в дальнейшем при построении диаграммы.

Для сбора статистики добавим в нашу модель диаграмму, показывающую процент брака для данного тиража.

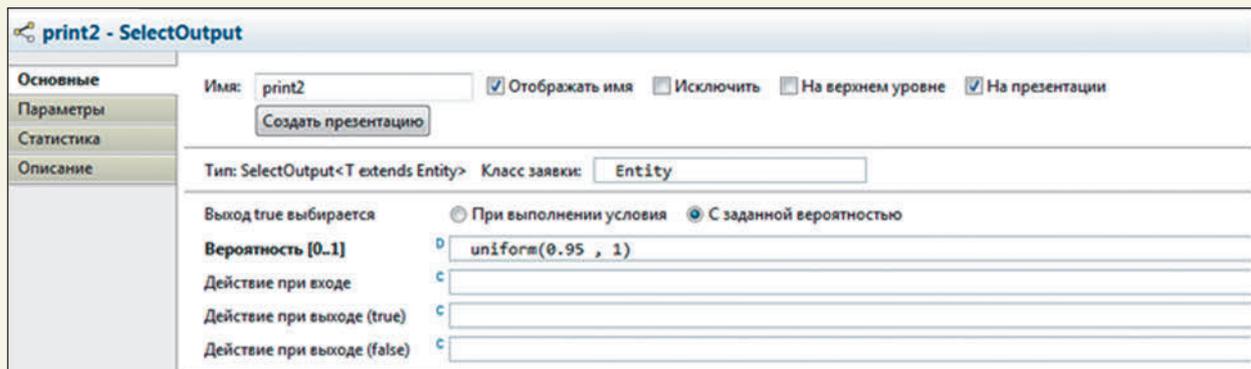


Рис. 8. Объект print2

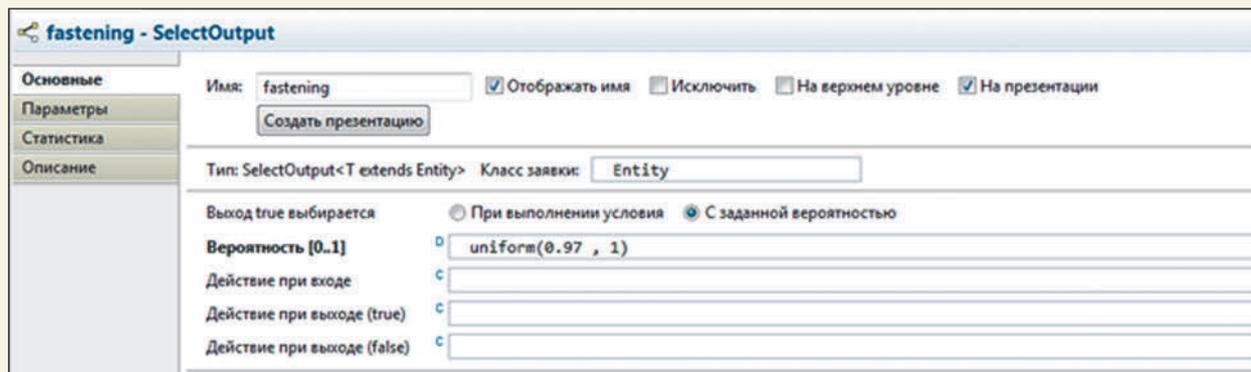


Рис. 9. Свойства объекта fastening

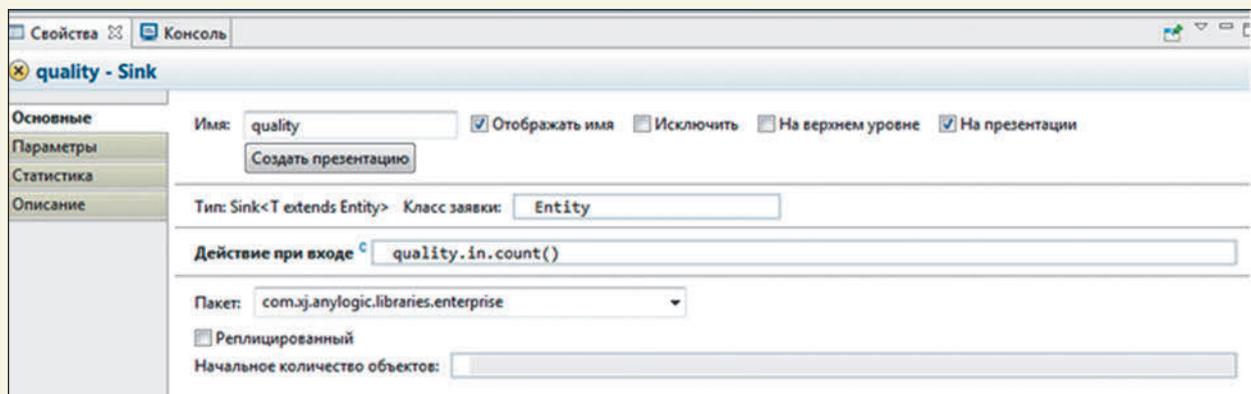


Рис. 10. Свойства объекта quality

Для этого из палитры *Статистика* добавим объект **Круговая диаграмма** и зададим для него параметры, которые отображены на рис. 11 на с. 22. На ней будет отображаться процентное количество бракованного и качественного продукта, полученного на выходе.

Запустим нашу модель (рис. 12).

Таким образом, можно увидеть, что в подавляющем большинстве случаев количество брака не превышает допустимую норму.

Модель 2. Оптимизация работы аэропорта

Директор аэропорта решил выявить недостатки работы своего аэропорта, которые могут возникнуть в процессе обслуживания пассажиров перед вылетом, с целью повысить уровень предоставления услуг.

При этом известно, что:

- пассажиры прибывают в среднем за 2–2,5 часа до вылета самолета (т.е. подавляющая часть пассажиров прибывает в аэропорт в течение получаса);
- среднее число пассажиров одного самолета — 500;
- пассажиры обязаны пройти процедуру регистрации (но мы учитываем, что некоторые пассажиры могли зарегистрироваться на рейс дома в режиме on-line, что исключает необходимость проходить регистрацию непосредственно в аэропорту);

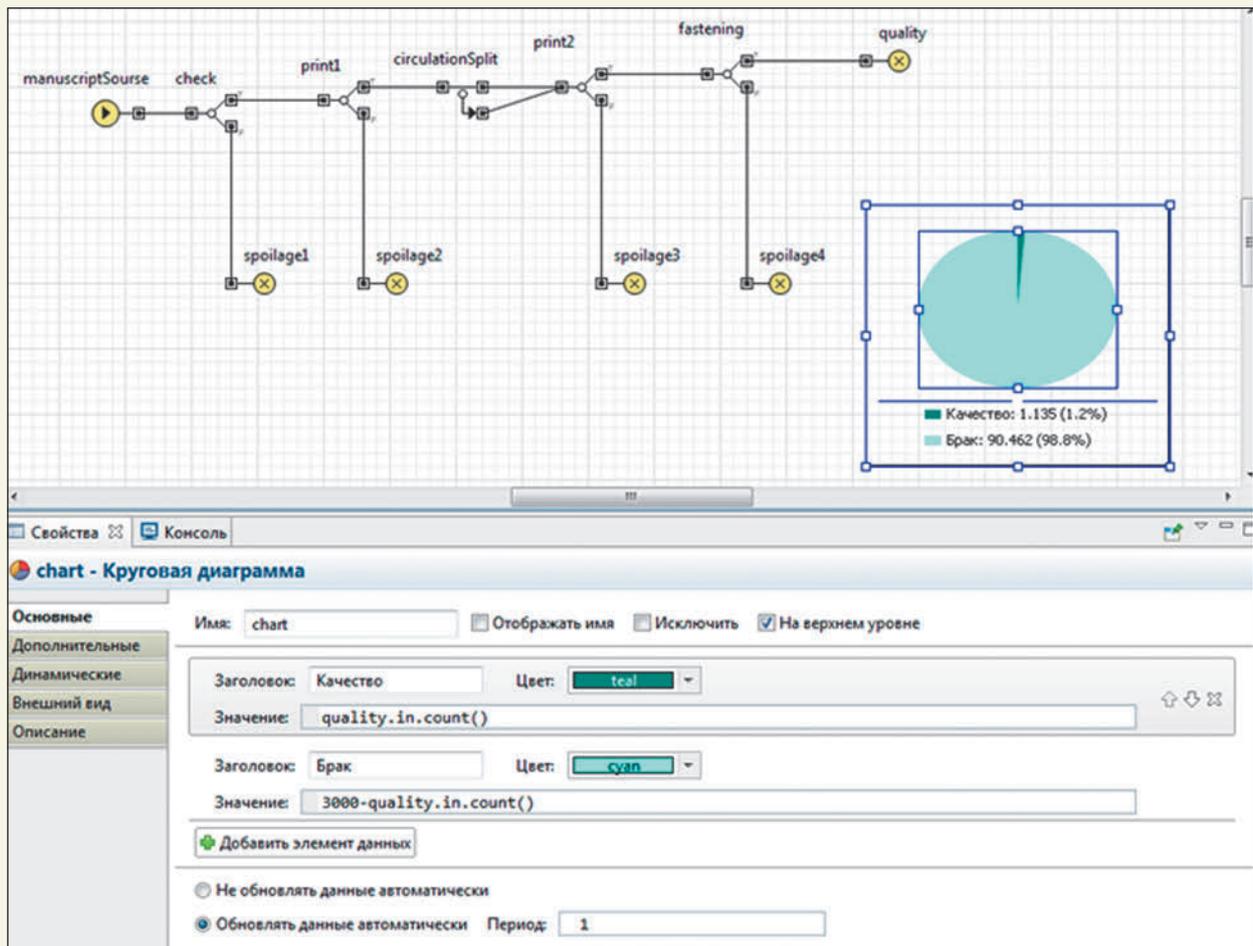


Рис. 11. Сбор данных

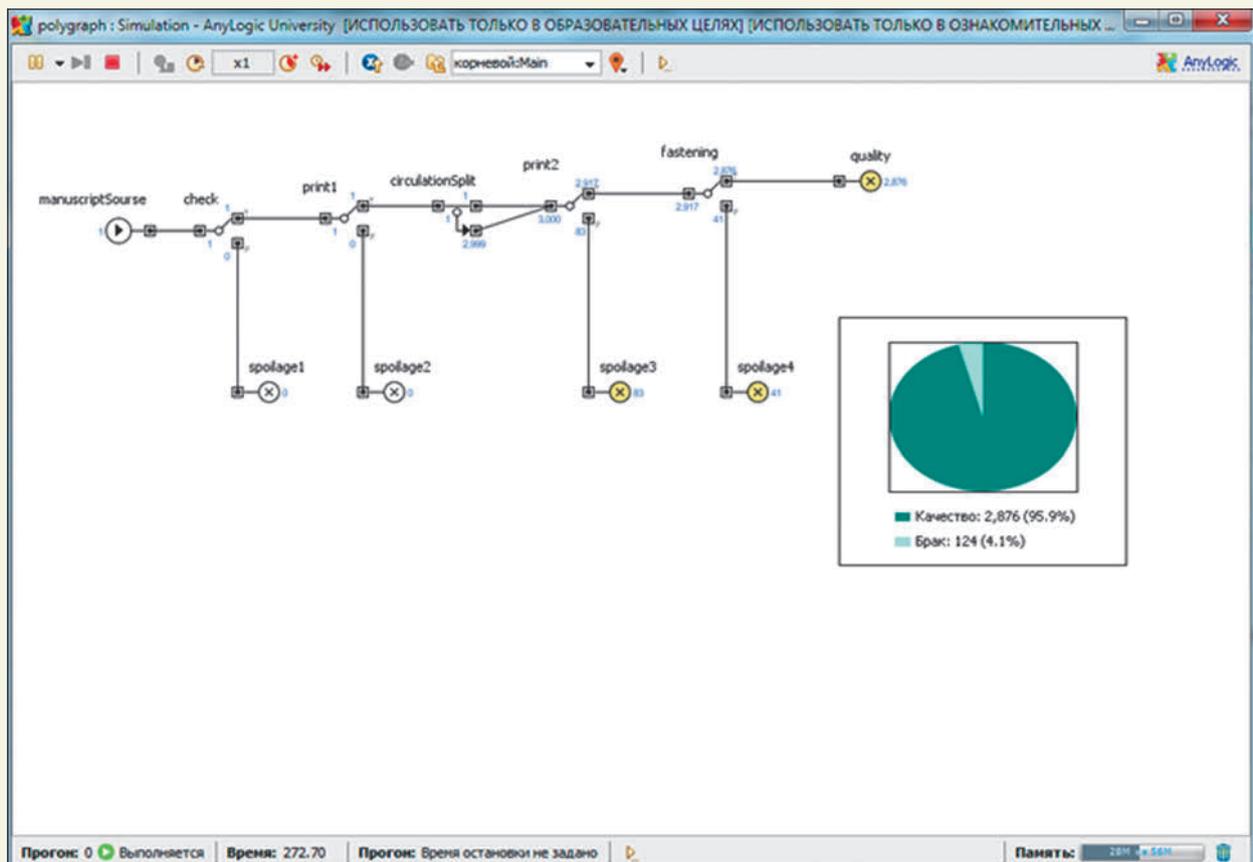


Рис. 12. Запуск модели

- все пассажиры обязаны пройти паспортный контроль и досмотр вещей (как багажа, так и ручной клади);
- пассажиры обязаны сдать багаж, не относящийся к ручной клади (при этом не все пассажиры имеют при себе багаж);
- перелеты совершаются внутри страны (что исключает необходимость проводить таможенный контроль).

Цель: Построить модель прохождения всех инстанций аэропорта перед вылетом с целью выявления недостатков обслуживания и их дальнейшего устранения.

Для удобства будем считать, что никто из пассажиров не “промахнулся” с весом багажа (так как для его веса есть ограничения), не забыл дома паспорт и прочие нужные вещи. В контексте данной модели такие события не являются важными, поэтому мы их не рассматриваем.

Создадим чистую модель (Файл → Создать → Модель), на втором этапе выберем “с нуля” (рис. 13 и 14).

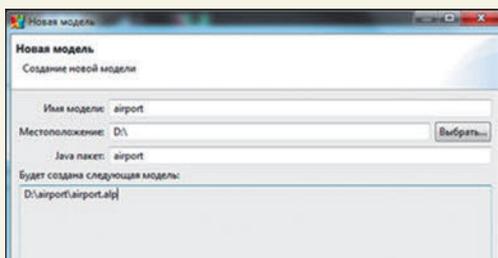


Рис. 13. Создание модели

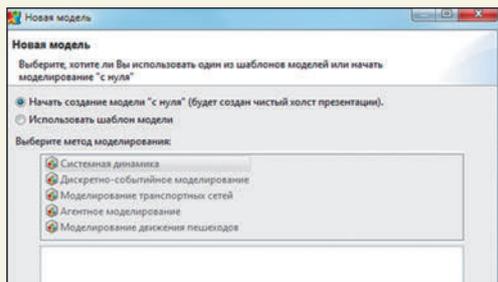


Рис. 14. Выбор пустого шаблона

Для нашей агентной модели воспользуемся объектами с закладки Пешеходная библиотека (рис. 15).

Прежде чем начать строить нашу модель, мы сделаем небольшую схему аэропорта для наглядности. Для этого с закладки Презентация воспользуемся следующими инструментами: Линия, Ломаная, Прямоугольник. Для того чтобы вос-

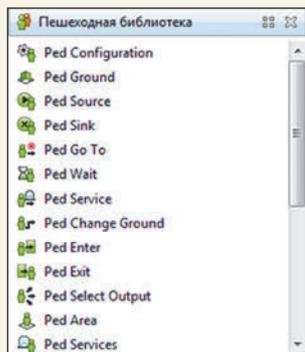


Рис. 15. Палитра “Пешеходная библиотека”

пользоваться инструментом, необходимо дважды щелкнуть на его название (или один раз на значок карандаша, находящийся напротив соответствующего инструмента). Также мы будем использовать Текст, но для его использования необходимо “захватить” его и перенести в необходимую точку рабочего места.

Начнем с постройки стен аэропорта. Для этого выберем инструмент Ломаная и изобразим ломаную линию, как показано на рис. 16. Для того чтобы данная ломаная служила в нашей модели стеной, нам нужно на ней нажать “Группировка → Создать группу” и дать имя созданной группе wall.

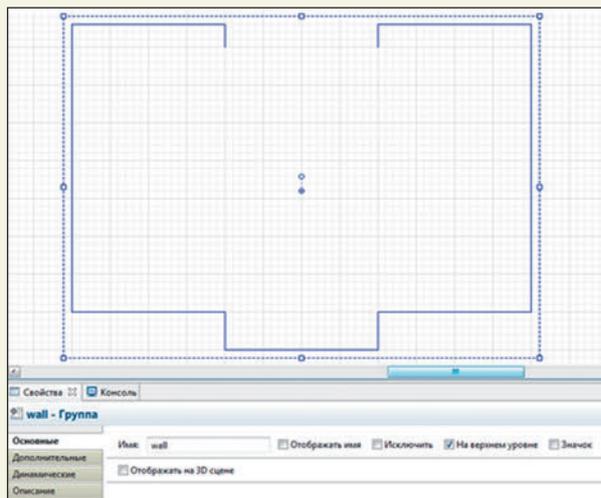


Рис. 16. Стены аэропорта

Затем подпишем с помощью инструмента Текст названия инстанций на том месте, где они будут находиться. Их 4: registration (регистрация), control (паспортный контроль), inspection (досмотр) и baggage (сдача багажа). Вместе с тем разместим рядом с надписями control и baggage прямоугольники (рис. 17).

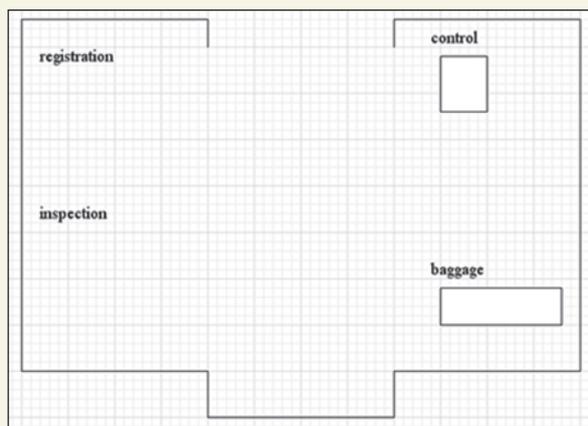


Рис. 17. Обозначение необходимых инстанций

Но одних прямоугольных зон нам будет недостаточно, поэтому внутри каждого прямоугольника построим по одной линии (рис. 18 и 19 на с. 24).

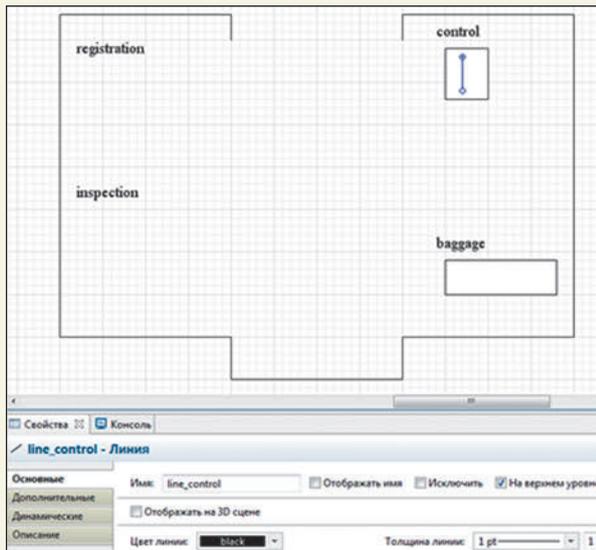


Рис. 18. Построение линии инстанции control



Рис. 19. Построение линии инстанции baggage

Далее поместим линию, где будут появляться наши агенты. Назовем ее input (рис. 20).

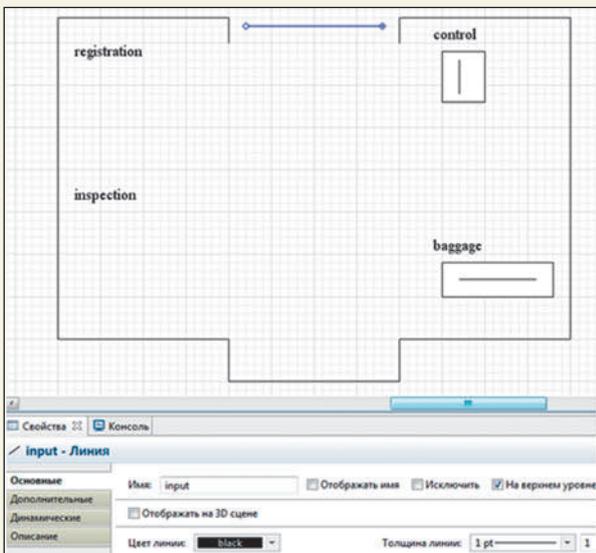


Рис. 20. Место появления агентов

Еще нам потребуется линия выхода. Ее мы назовем output (рис. 21).

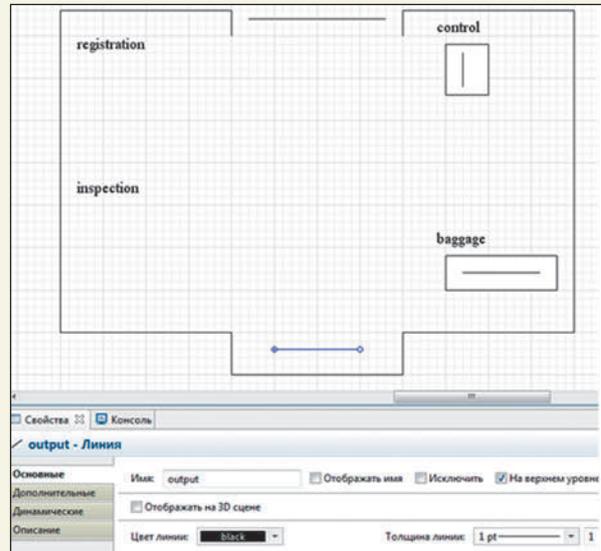


Рис. 21. Место выхода агента из модели

Теперь нам необходимо разобраться с инстанциями registration и inspection. Здесь нам надо построить по два столбца, состоящих из трех линий, расположенных друг над другом, как показано на рис. 22:

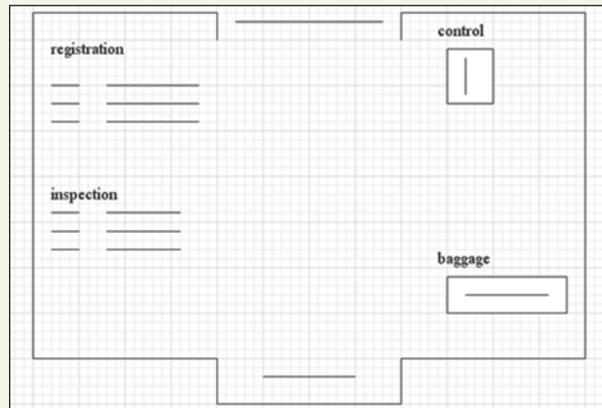


Рис. 22. Построение прямых

Затем объединим их в четыре группы: группы group_reg и group_insp отвечают за стойки обслуживания; queue_reg и queue_insp — за очереди к ним. Для объединения линий в группы необходимо, с зажатой клавишей **Shift**, последовательно выделить все линии, относящиеся к одной группе, кликнуть правой кнопкой мыши, выбрать “Группировка → Создать группу” и дать имя созданной группе, как показано на рис. 23 и 24.

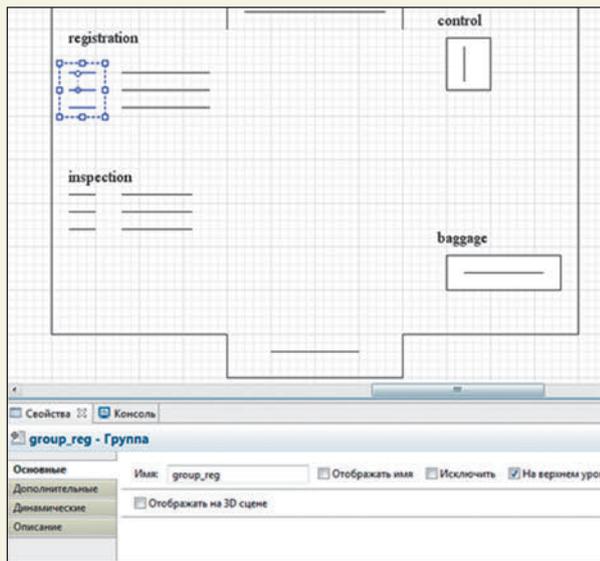


Рис. 23. Создание группы линий group_reg

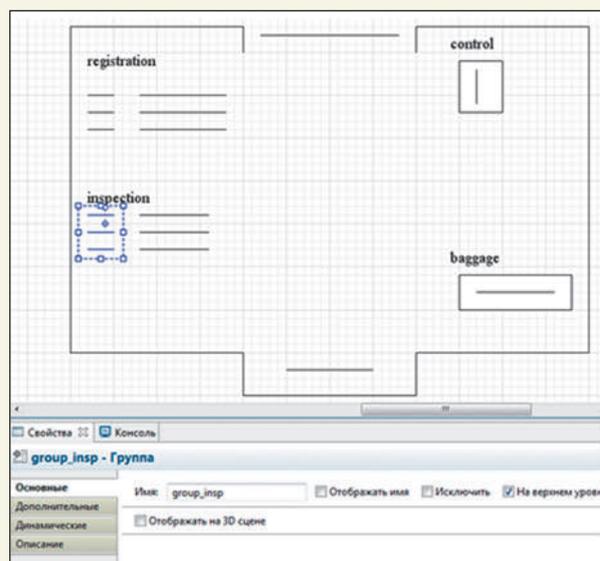


Рис. 25. Создание группы линий group_insp

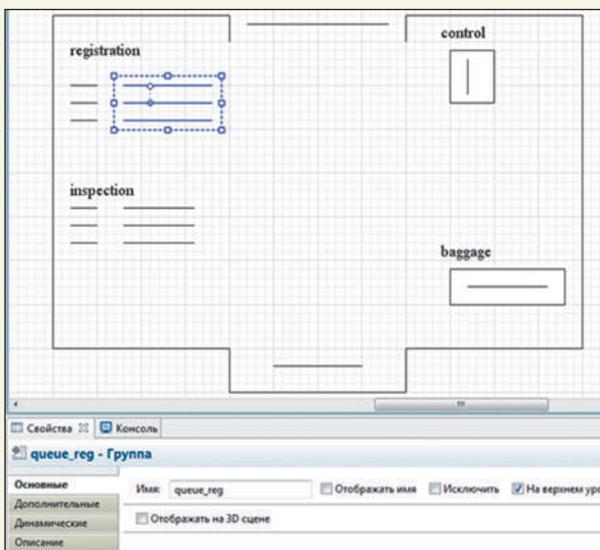


Рис. 24. Создание группы линий queue_reg

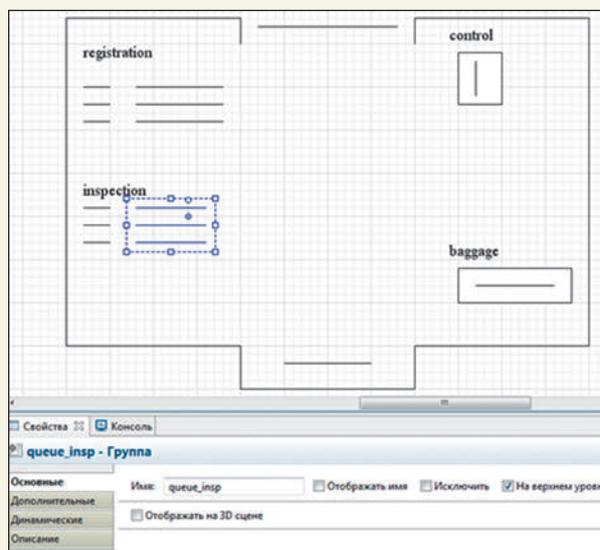


Рис. 26. Создание группы линий queue_insp

Аналогично необходимо создать группы для инстанции inspection (рис. 25 и 26).

Итак, схема нашего аэропорта готова. Теперь приступим к описанию модели. Для начала построим цепочку, как показано на рис. 27:

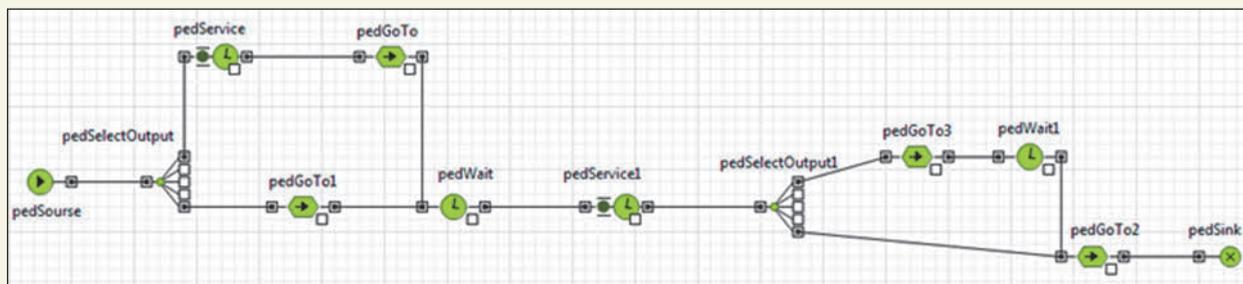


Рис. 27. Построение блок-схемы модели

Для ее построения были использованы следующие объекты пешеходной библиотеки:

- pedSource — начальная точка нашей схемы, здесь создаются пешеходы;
- pedService — создает очередь;
- pedGoTo — задает направление движения;
- pedWait — ожидание (задержка в заданной точке);
- pedSelectOutput — здесь происходит выбор направления движения в зависимости от коэффициента предпочтения;

- pedSink — выход из модели.

Затем добавим некоторые элементы управления моделью, в которых будут задаваться параметры нашей модели (рис. 28).

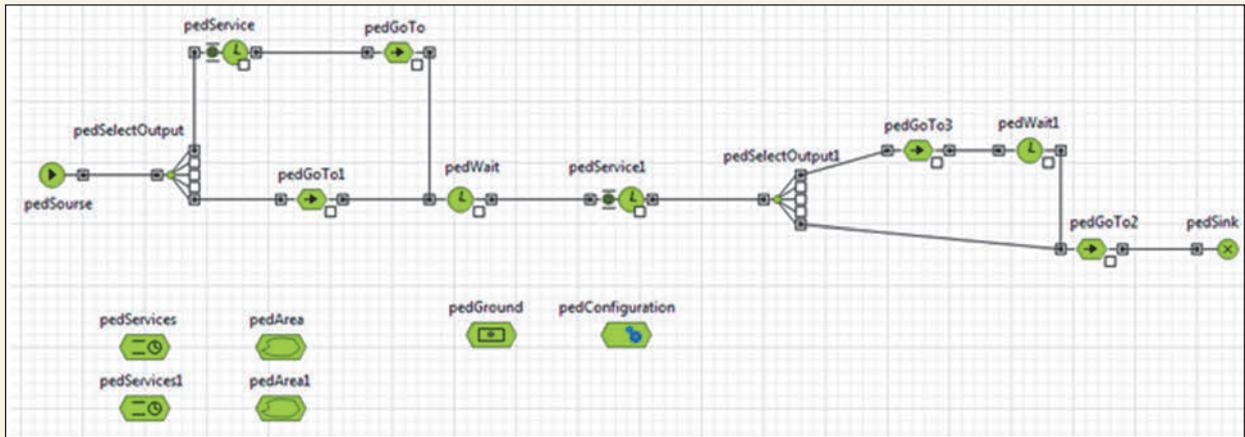


Рис. 28. Дополнительные объекты модели

Модель начинается с объекта **pedSource**. Для него во вкладке свойств нам необходимо установить следующие параметры (рис. 29): интенсивность пешеходов — 1000 в час, поставить галочку напротив параметра “ограничить прибытия” и в соответствующей графе установить значение 500 (среднее количество пассажиров самолета).

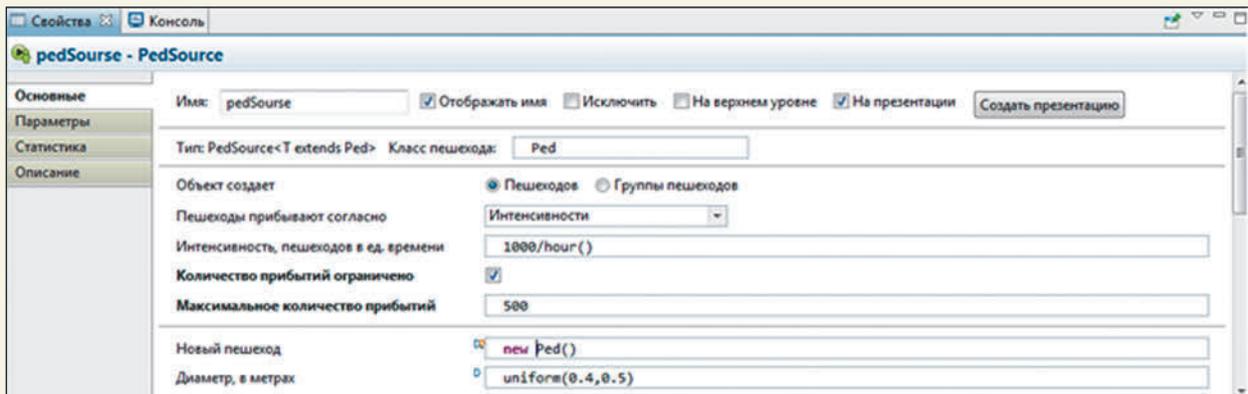


Рис. 29. Свойства объекта pedSource

После того как агент попал в модель (зашел в аэропорт), происходит выбор инстанции, куда ему отправляться: с вероятностью 0,8 он пойдет на стойку регистрации и с вероятностью 0,2 он отправится сразу в паспортный контроль (такой случай возможен, если пассажир купил и распечатал билет дома). Соответствующие данные записываем в свойства объекта **pedSelectOutput** (рис. 30).

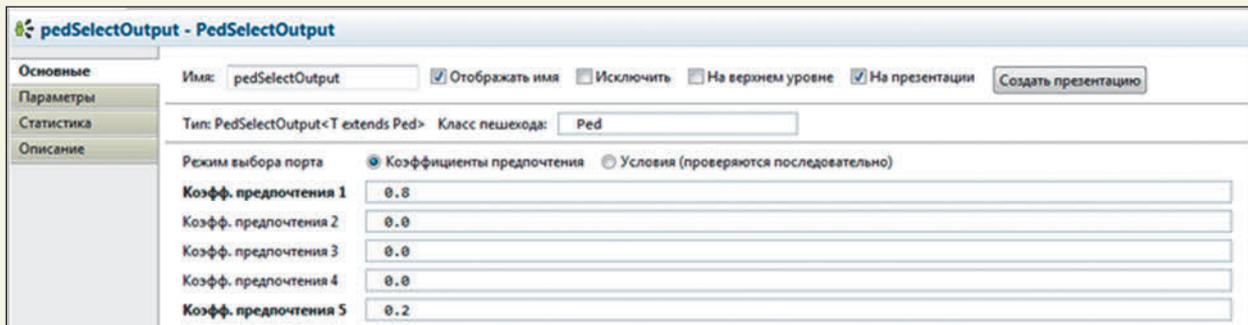


Рис. 30. Объект pedSelectOutput

Элемент нашей модели **pedService** отвечает за очередь к стойке регистрации, но параметры для него указываются в дополнительном объекте **pedServices**. Поэтому в объекте **pedService** необходимо лишь указать сервис, в котором будут указаны все условия, которые должны выполняться на данном этапе в процессе работы модели (рис. 31). Аналогично, объект **pedService1** (очередь на досмотр пассажиров) будет ссылаться на дополнительный объект **pedServices1**.

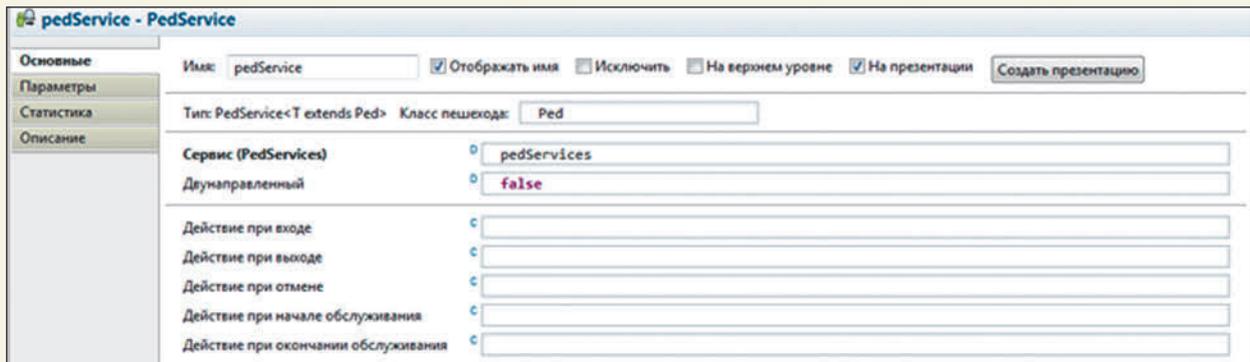


Рис. 31. Свойства элемента pedService

Дальше разберемся с объектами **pedGoTo**. В нашем случае их четыре, и они задают следующие направления движения агентов:

- **pedGoTo** и **pedGoTo1** — направление к стойке паспортного контроля;
- **pedGoTo3** — к сдаче багажа;
- **pedGoTo2** — выход в зону вылета (выход из модели).

Для элементов **pedGoTo** и **pedGoTo1** свойства будут одинаковыми (рис. 32). В случае **pedGoTo3** и **pedGoTo2** цель, которая должна быть достигнута агентом, будет отличаться (линии `line_baggage` и `output` соответственно). Остальные параметры оставляем без изменения.

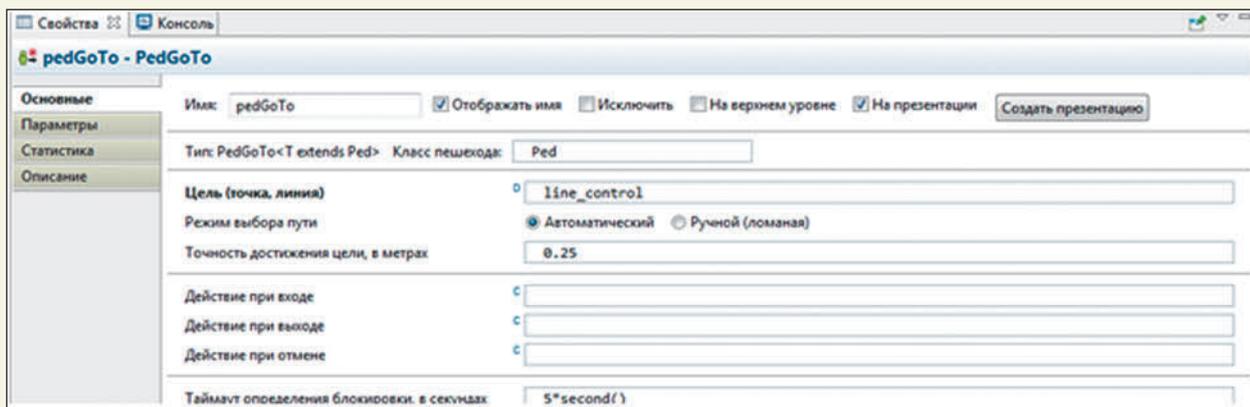


Рис. 32. Параметры объектов pedGoTo и pedGoTo1

Затем рассмотрим объект **pedWait** (паспортный контроль). Здесь необходимо указать время задержки агента у данной стойки. Так как проверка паспортных данных проходит достаточно быстро, установим время от 1 до 2 минут (рис. 33), а область ожидания — **pedArea**. Отметим, что в модели указывается модельное время и одна секунда модельного времени равна одной минуте реального времени.

В объекте **pedSelectOutput1** (рис. 34) происходит выбор: агент направляется к пункту сдачи багажа (с вероятностью 0,9) или же у него отсутствует багаж (только ручная кладь, которую можно взять с собой в салон самолета), с соответствующей вероятностью, равной 0,1.

Агенты, имеющие багаж, попадают в инстанцию сдачи багажа, соответствующую объекту модели **pedWait1**, где время ожидания варьируется от 0,5 до 1 минуты, а область ожидания — **pedArea1** (рис. 35 на с. 29).

И последний элемент схемы нашей модели — **pedSink**. Здесь пока никакие изменения в свойства вносить не нужно.

Рассмотрим теперь “дополнительные” объекты, необходимые для корректного выполнения нашей модели.

В объектах типа **pedServices** нам необходимо указать следующие параметры: сервисы (стойки регистрации), время задержки (время обслуживания одного агента) и очереди. Для объекта **pedServices** данные параметры отображены на рис. 36.

Аналогично заполняем свойства объекта **pedServices1** (рис. 37 на с. 30).

В объекте **pedGround** нам необходимо указать стены нашего аэропорта (рис. 38 на с. 30). За них в нашей модели отвечает группа линий `wall`.

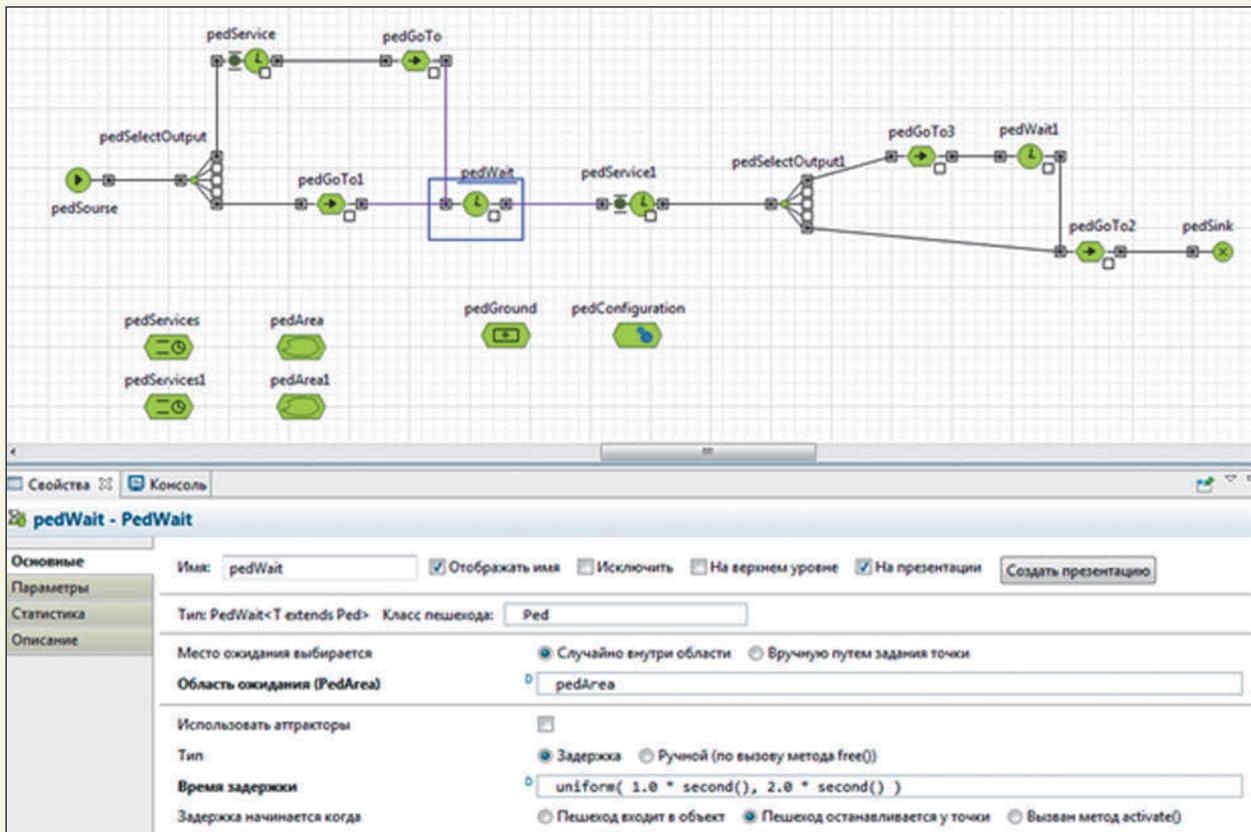


Рис. 33. Свойства объекта pedWait

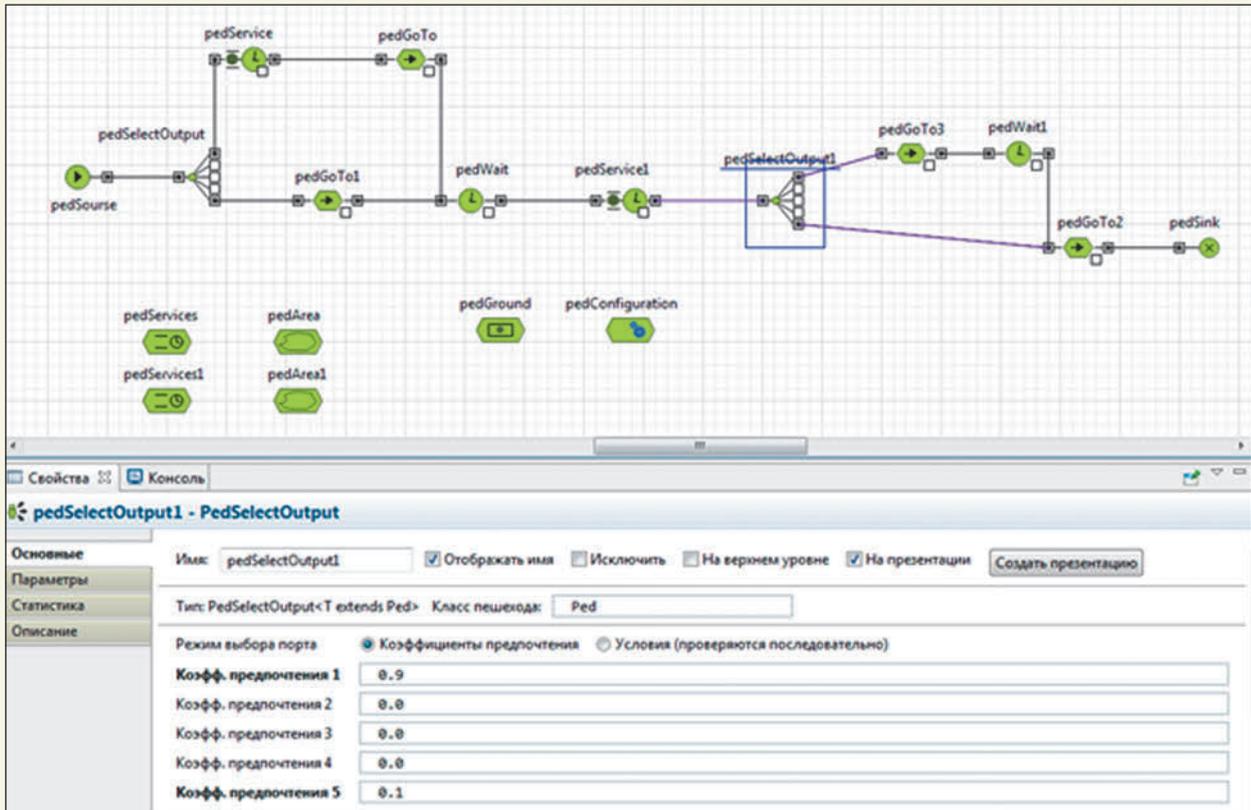


Рис. 34. Свойства элемента pedSelectOutput1

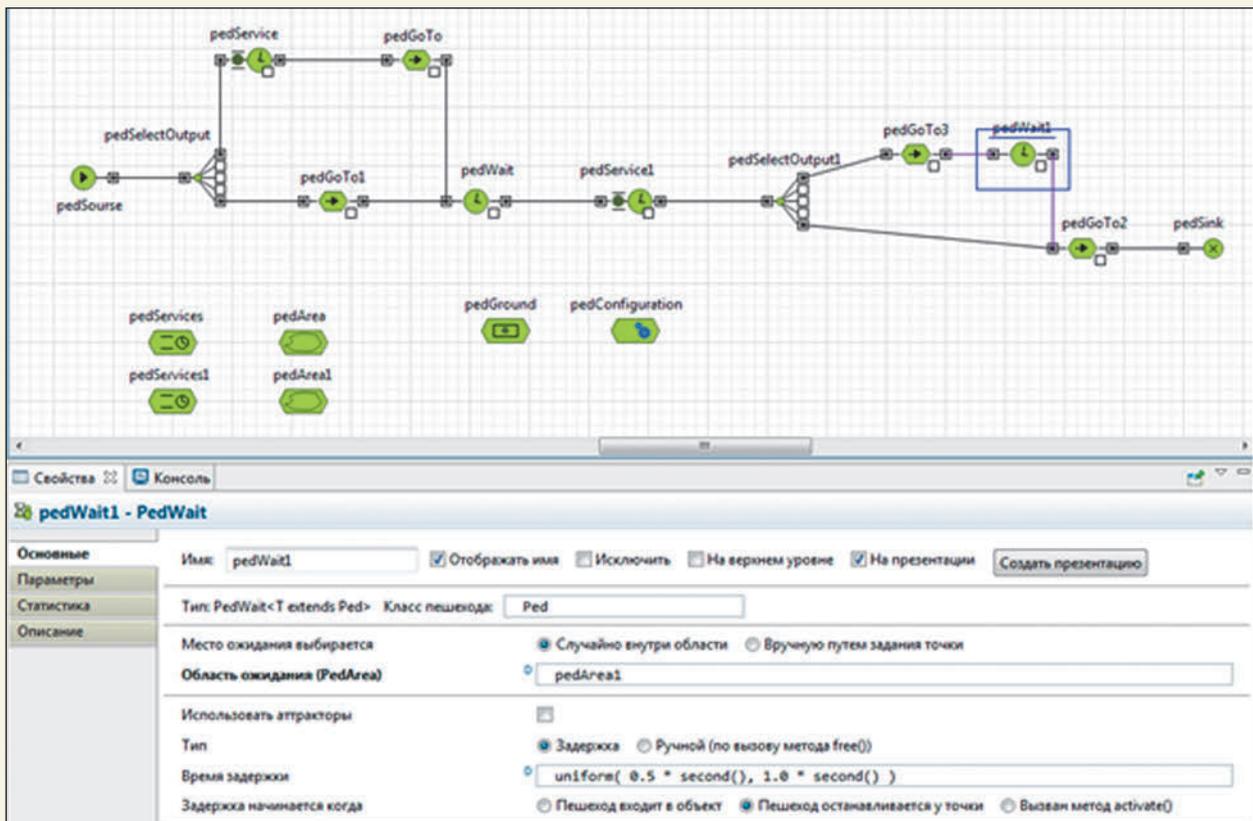


Рис. 35. Свойства объекта pedWait1

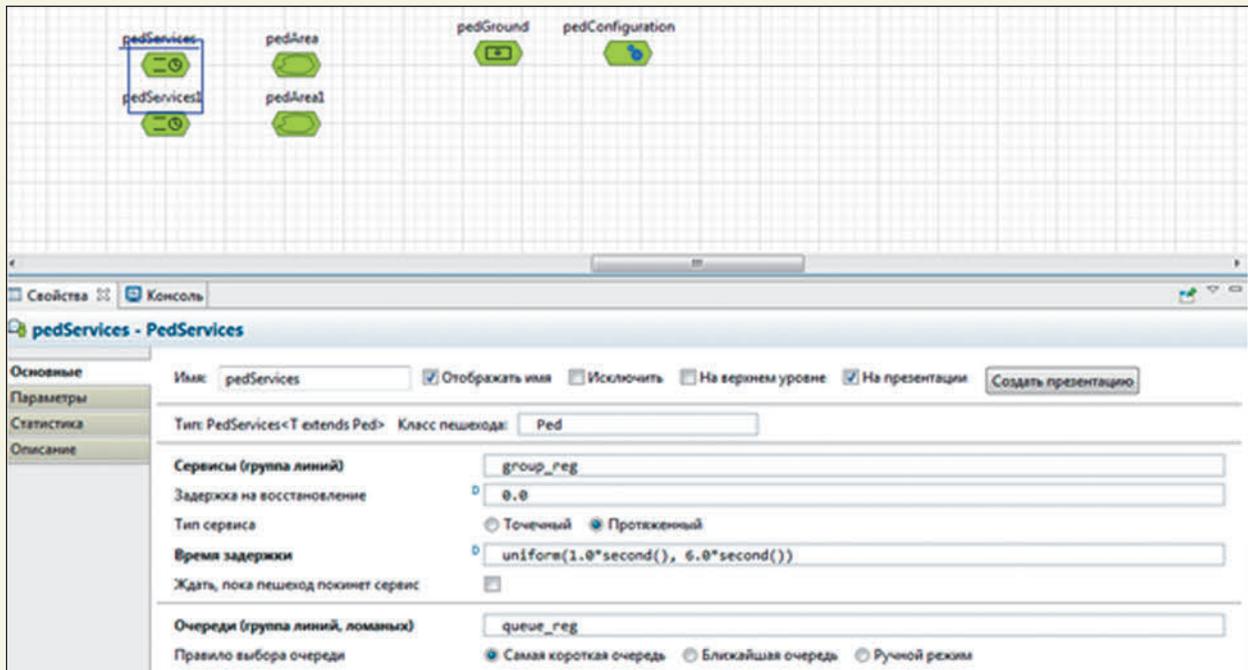


Рис. 36. Свойства объекта pedServices

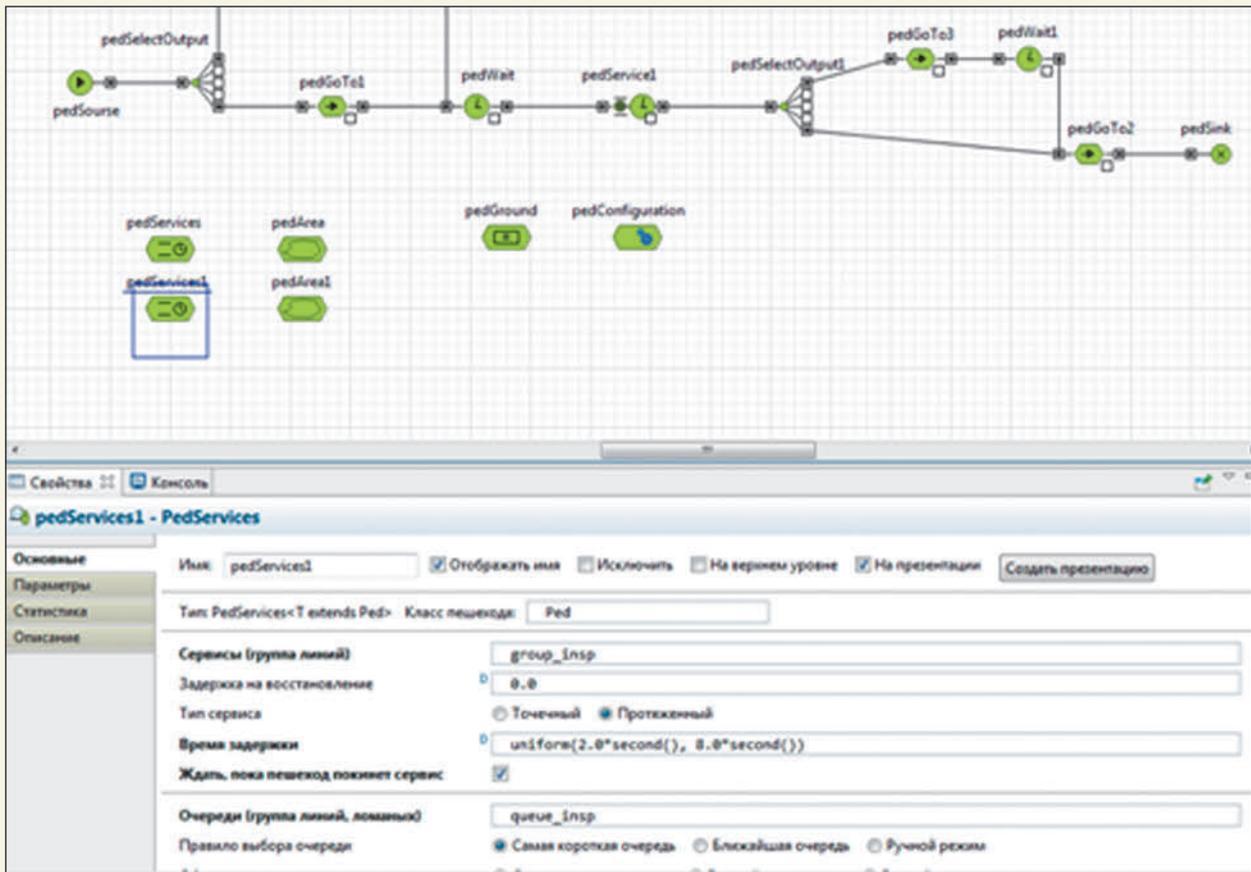


Рис. 37. Свойства объекта pedServices1

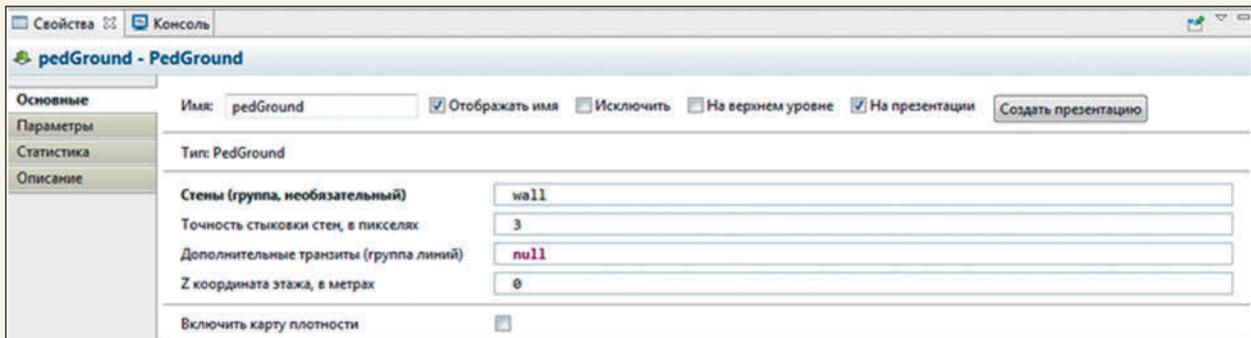


Рис. 38. Параметры объекта pedGround

Для объектов **pedArea** и **pedArea1** необходимо указать фигуру, в области которой будет проходить ожидание, и этаж. В нашем случае этаж для обоих элементов будет один и тот же — **pedGround**, а фигуры, внутри которых агенты будут ожидать обслуживания, **gestangle** (рис. 40) и **gestangle1** соответственно.

Объект **pedConfiguration** оставляем без изменений.

Итак, все объекты нашей модели описаны. Теперь добавим функцию сбора статистики, с помощью которой мы определим среднее время прохода агентом всей модели (всех инстанций аэропорта). С палитры Статистика перетащим объект Статистика на рабочую область и дадим ей название **moveTime**. Для того чтобы данный элемент работал, нам необходимо добавить Java-класс. Для этого во вкладке Проекты нажмем правой кнопкой мыши на название нашей модели, затем Создать → Java-класс.

Дадим ему название **AeroPed**, а в качестве базового класса укажем класс **ped** (рис. 39).

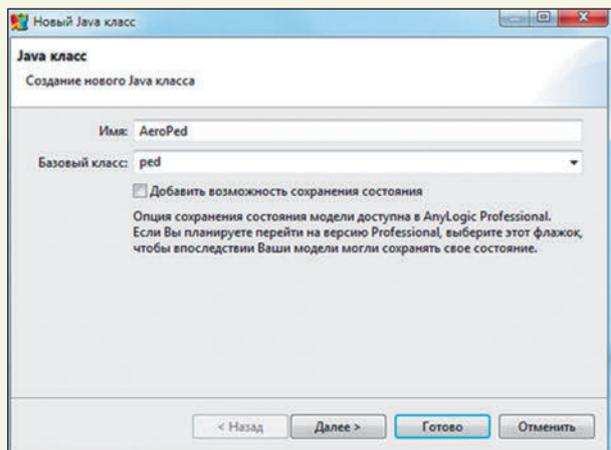


Рис. 39. Создание нового Java-класса

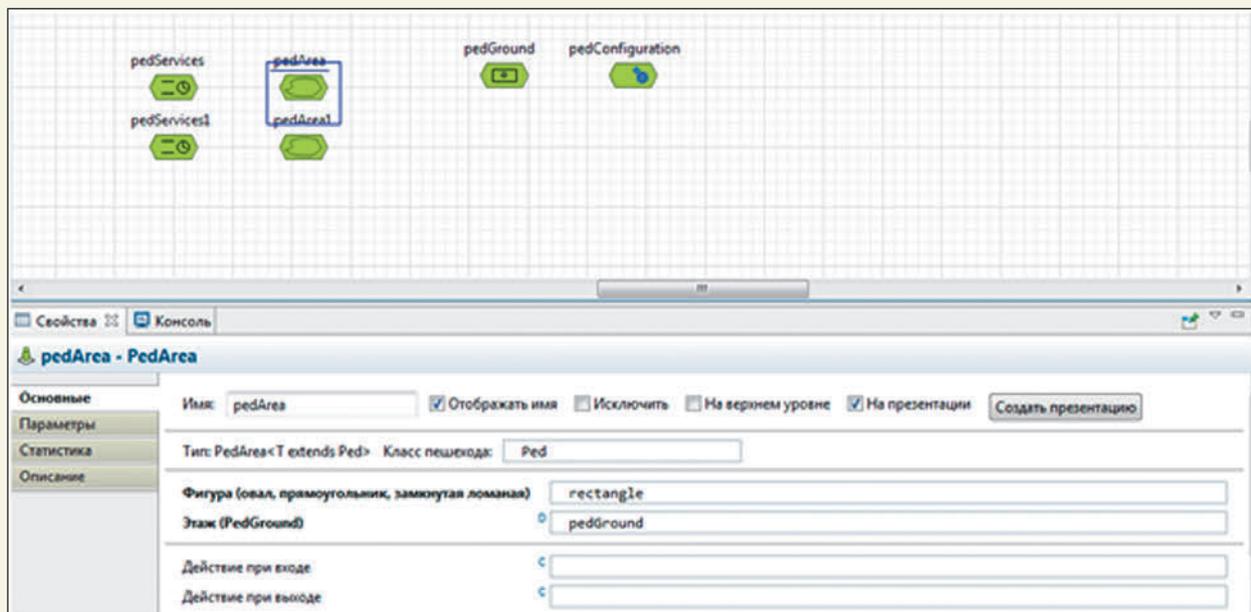


Рис. 40. Свойства объекта pedArea

После этого нажимаем на кнопку Далее и в следующем окне (рис. 41) добавим параметр startMove, в котором будет храниться время входа. Тип данного параметра устанавливаем double, так как модельное время обрабатывается именно так.

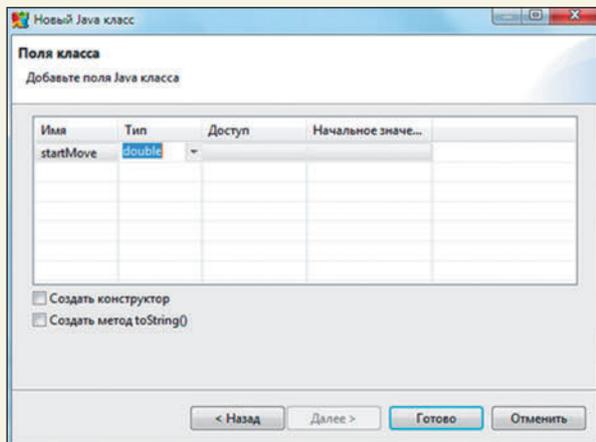


Рис. 41. Добавление параметра startMove

Наш Java-класс готов. Теперь нам необходимо изменить параметры объекта источника:

- Класс пешехода заменим на AeroPed;
- В графе Новый пешеход new Ped() заменим на new AeroPed();
- В графе Действие при выходе запишем время: ((AeroPed)ped).startMove = time();
- Для объекта pedSink необходимо сменить класс пешехода (так же, как для объекта pedSource) и указать действие при входе, добавляющее данные к статистике: moveTime.add(time()-((AeroPed)ped).startMove); данная функция высчитывает разницу текущего времени и времени входа агента в модель.

Запустим нашу модель и посмотрим результат ее выполнения (рис. 42).



Рис. 42. Запуск модели

Итак, в ходе прогона модели мы видим, что в пункте досмотра пассажиров у нас возникают огромные очереди, которые мешают пройти остальным агентам к их цели. При этом среднее время прохождения агентом всех инстанций — около 16 минут (рис. 43).

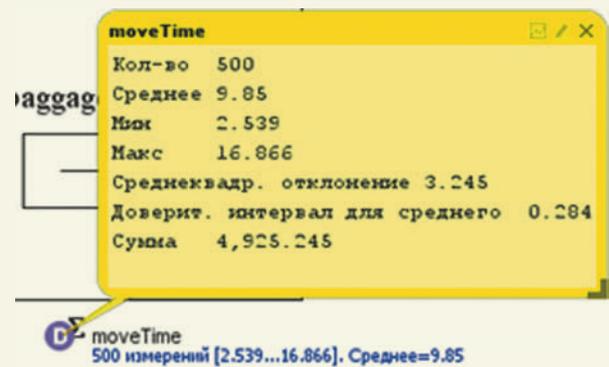


Рис. 43. Время обслуживания одного пассажира

Для исправления данной проблемы создадим еще одну, четвертую очередь в пункте досмотра пассажиров. Для этого в палитре Презентация выберем инструмент Линия и построим их, как показано на рис. 44 и 45.

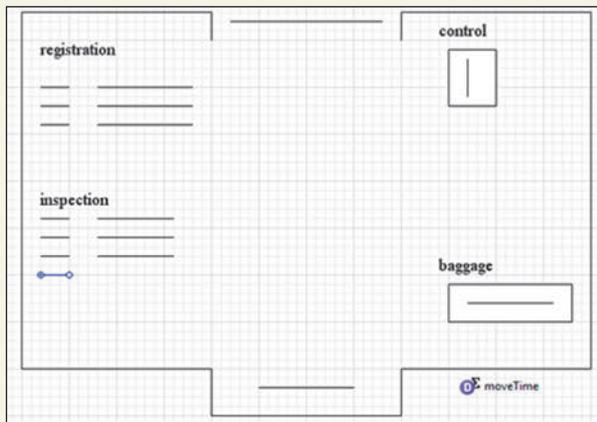


Рис. 44. Добавление стойки обслуживания в пункте досмотра пассажиров

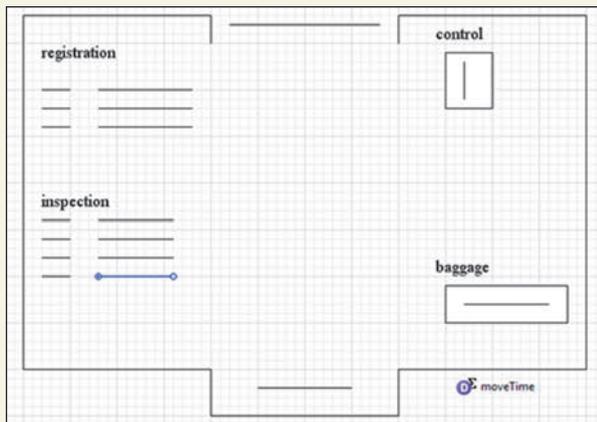


Рис. 45. Добавление очереди к стойке обслуживания в пункте досмотра пассажиров

Но для того чтобы наши линии относились к нужным нам группам, необходимо сделать следующее: на линии, отвечающей за дополнительную стойку обслуживания, щелкнуть правой кнопкой мыши, выбрать Группировка → Добавить в существующую группу. В рабочей области появится следующее (рис. 46):

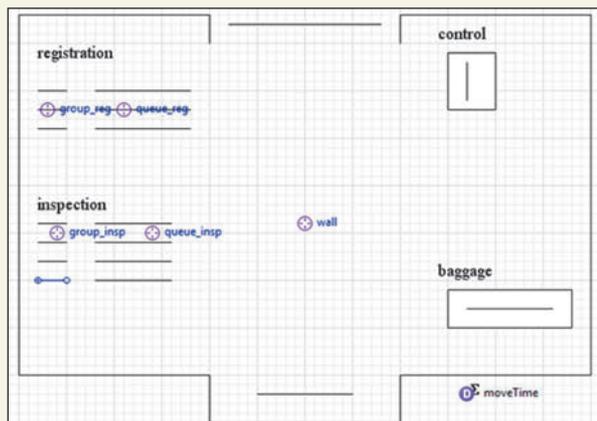


Рис. 46. Отображение всех групп, существующих в данной модели

Для добавления линии в группу достаточно щелкнуть на розовый кружок с делениями, относящийся к той группе, в которую мы хотим добавить линию (в нашем случае это группа group_insp).

Аналогичные действия необходимо совершить для второй линии, но ее нужно добавить в группу queue_insp.

Попробуем снова запустить нашу модель (рис. 47).



Рис. 47. Повторный запуск модели

Очевидно, что проблема с огромными очередями в пункте досмотра решена. При этом среднее время обслуживания одного агента снизилось до восьми минут (рис. 48), что в два раза меньше, чем было изначально.

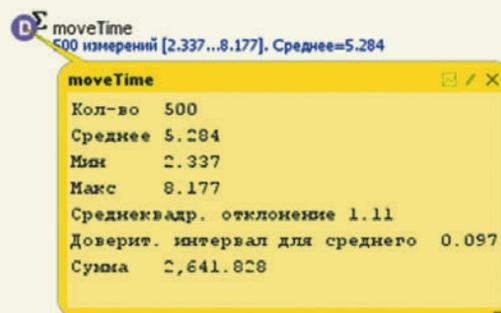


Рис. 48. Время обслуживания агентов

Исходя из данного факта, можно сделать вывод, что директору для улучшения качества обслуживания пассажиров необходимо добавить еще одну стойку в пункте досмотра пассажиров.

Литература

1. Калинин И.А., Самылкина Н.Н. Информатика. 10-й класс. Углубленный уровень. Учебник. М.: Бинном, 2013.
2. Калинин И.А., Самылкина Н.Н. Основы имитационного моделирования / И.А. Калинин, Н.Н. Самылкина // Информатика, № 5, 2013.



- Одновременное использование учебников в электронной форме на трех планшетных или стационарных компьютерах
- Использование учебников без подключения к сети Интернет
- Единые заметки и закладки на всех устройствах, которые вы используете
- Более 500 учебников в электронной форме



ОБЪЕДИНЕННАЯ ИЗДАТЕЛЬСКАЯ ГРУППА

ВСЕРОССИЙСКИЙ КОНКУРС «ЭЛЕКТРОННЫЙ УЧЕБНИК НА УРОКЕ»

апрель — октябрь 2016

Подробнее на efu.drofa.ru

КАК ПЕРЕЙТИ НА LECTA ПОЛЬЗОВАТЕЛЯМ ОРФОГРАФ И АЗБУКА?

- Установить приложение LECTA на своем устройстве: www.lecta.ru
- В разделе «Авторизация» пройти процедуру восстановления пароля с зарегистрированным адресом электронной почты в системах Орфограф и Азбука
- Получить на свой адрес электронной почты новый пароль и произвести вход в приложении LECTA

Все учебники, ранее выданные в системах Орфограф и Азбука, сохраняются!

Не получается? Пишите: start@lecta.ru



#ПЕРЕХОДИНАЛЕКТА



От компьютерной грамотности и внедрения ИКТ к трансформации работы школы

Г.М. Водопьян,
школа № 550,
Санкт-Петербург
(gv@ort.spb.ru),
А.Ю. Уваров,
ИОИ ФИЦ ИУ РАН,
Москва
(auvarov@mail.ru)

► В конце 90-х годов на страницах журнала “Информатика” состоялась дискуссия [1, 2] о связи этого учебного предмета с изменением содержания и методов учебной работы в школе. В ходе нее стало ясно, что эта связь существует, но реализующие ее процессы еще только начинают разворачиваться. К середине 2000-х очертание пространства задач информатизации школы стало общепризнанным [3], а к началу нашего десятилетия прояснились направления и перспективы развития этого процесса. Информатизация образования, которая до этого традиционно трактовалась как “широкое внедрение электронно-вычислительной техники в учебный процесс” [4], превращалась

в систематическую (планомерно организованную) работу по всесторонней трансформации работы школы на основе индивидуализации (персонализации) образовательного процесса [5]. Эти теоретические представления сегодня подталкивают и направляют многие работы по решению непростых проблем, стоящих перед отечественным образованием. В этой статье представлено описание одной из таких работ. Последние шесть лет ее проводит коллектив педагогов экспериментальной площадки ФГАУ “Федеральный институт развития образования” (ФИРО), которая действует на базе ГБОУ СОШ № 550 (Санкт-Петербург). Школа более двадцати лет сотрудничает со Всемирным ОРТ¹, без поддержки которого

¹ АНО Всемирный ОРТ (Образовательные Ресурсы и Технологический тренинг) — является одной из самых крупных негосударственных образовательных организаций в мире. С момента своего основания в 1880 г. он накопил опыт успешной и плодотворной работы в области образования и профессиональной подготовки населения по всему миру.

описанные далее преобразования были бы невозможны.

Трансформация работы школы, в которую превращается сегодня ее информатизация, — комплексный процесс, который захватывает все стороны работы образовательного учреждения, распространяется на все возрастные группы учащихся. Она невозможна без активной лидерской позиции администрации школы и учителей [6]. Одну из ключевых ролей здесь играют преподаватели информатики, которые из узких предметников и технических специалистов, поддерживающих развитие цифровой информационной среды школы, переходят в позицию системных аналитиков и проектировщиков образовательного процесса. Систематическое обсуждение данных вопросов выходит далеко за рамки этой статьи. Мы ограничимся обсуждением того, как в таких условиях трансформируется изучение информатики, если необходимо, чтобы каждый школьник освоил ее в полном объеме.

Глубина трансформации учебного процесса при использовании ИКТ

Информатизация учебного процесса была одним из направлений развития с момента основания школы № 550 в начале 90-х годов [7]. Коллектив школы последовательно работал над:

- созданием ИКТ-насыщенной цифровой образовательной среды;
- развитием педагогической ИКТ-компетентности учителей, подготовкой их к работе с использованием цифровых учебно-методических материалов;
- обеспечением компьютерной грамотности учащихся, формированием у них алгоритмического мышления, развитием способности успешно использовать ИКТ для решения учебных и практических задач.

Масштабы трансформации учебного процесса постепенно нарастали. Это хорошо описывает модель углубления информатизации образовательного процесса² (см. рис. 1). Поясним выделенные здесь уровни на примере.

Замещение (1). Типичный пример “замещения” — переход от чтения текстов в учебнике к чтению их на экране компьютера (нетбука, планшета, смартфона, ридера и т.п.). Бумажная версия текста просто заменяется цифровой. Использование для заметок вместо карандаша и блокнота MS OneNote — еще один пример “замещения”. Работа с персональными устройствами может заинтересовать учащихся и даже повысить учебные результаты, но уровень трансформации учебного процесса здесь минимальный. В прошлом веке на эту тему шутили так: “Раньше птицы сидели на

² За рубежом она получила название SAMR (*Substitution Augmentation Modification Redefinition*). Ее описание для учителей см., например, [8].



Рис. 1. Глубина трансформации процесса при обновлении технологии

телеграфных проводах, а теперь они сидят на антеннах”.

Улучшение (2). Продолжим пример с текстом. В отличие от бумажной книги в цифровой можно посмотреть определение или синонимы незнакомого слова, не отрываясь на поиски в словарях, а просто кликнув на него мышкой (пальцем). При подготовке текста в MS OneNote можно проверить орфографию, вставить в текст фото, видеофрагмент и т.п. Все это — примеры улучшения работы (расширение функциональности), которые несет с собой технология.

Изменение (3). На предыдущих уровнях базовый процесс не менялся. Новая технология лишь упрощала или обогащала его. На уровне “изменение” начинается трансформация процесса, изменение его функциональности. Переопределяются традиционные задачи учебной работы. Например, учащиеся создают мультфильмы и “цифровые повествования”, готовят презентации с использованием MS Office Mix не только для представления результатов выполненного проекта, но и для решения задач взаимного обучения. Изменение связано с обновлением целей и задач учебной работы, персонализацией обучения.

Преобразование (4). На этом самом глубоком уровне решаются задачи трансформации учебного процесса. Они неизбежно затрагивают не только отдельные учебные дисциплины, но и работу всей школы. Эти задачи невозможно решить без широкого использования информационных технологий при подготовке учебного процесса, управлении им и его проведении.

Одной из таких задач для нас стала задача результативного обучения информатике каждого учащегося старшей школы. За ее решение мы взялись в 2009 году. До этого основное внимание в школе уделялось освоению ИКТ как инструментов учеб-

ной работы. При этом базовая модель учебного процесса оставалась практически неизменной.

Казалось, что освоение новых программных и инструментальных средств повысит заинтересованность школьников, поможет улучшить качество обучения, которое мы оценивали на экзаменах по завершении образовательного процесса. Однако опыт вновь и вновь свидетельствовал, что технологические изменения не гарантируют достижение желаемой цели, а ориентация на констатирующий контроль не позволяет оперативно выявлять и корректировать недостаточно эффективные приемы учебной работы, учебные инструменты и материалы.

Стало ясно, что без трансформации модели учебного процесса не обойтись. Описанию опыта решения этой задачи посвящена данная статья.

Индивидуализация и персонализация обучения

Коллектив школы поставил перед собой цель добиться результативного освоения информатики на профильном уровне каждым школьником путем трансформации учебного процесса за счет его индивидуализации. Обсуждая эту задачу, мы столкнулись с тем, что термин “индивидуализация” в педагогической среде часто трактуют по-разному. Утверждение некоторых педагогов о том, что “процесс индивидуализации образования — это институализация страны” [9], мало помогало на практике. Трактовки индивидуализации учебной работы оказались сильно размыты, в том числе при обсуждении привносимых цифровой революцией возможностей организовать самостоятельную работу учащихся с различными обучающими программами, использовать индивидуальные образовательные траектории при организации дистанционного обучения и т.п.

Сегодня редко вспоминают не только разработки моделей индивидуализированного обучения конца XIX и начала XX века (“Пуэбло-план”, “Санта-Барбара-план”, “Норт-Денвер-план”, “Батавиа-план”, “Дальтон-план”, “Виннетка-план” и др.), но и популярные проекты середины прошлого века

(“Индивидуально предписанное обучение”, “Индивидуально направляемое обучение”, “Система полного усвоения” и др.), а также программированное обучение.

Англоязычные педагоги пытаются решить эту проблему, введя дополнительный термин “персонализированное” (personalized) или “лично ориентированное” обучение. В нашем проекте мы используем следующие рабочие определения.

“Традиционное” обучение. Здесь, как правило, одно содержание учебной работы, один способ его предъявления, один темп учебной работы распространяются на всех учащихся в классе.

Дифференцированное обучение. Здесь, как правило, одно содержание учебной работы, один способ его предъявления, один темп учебной работы распространяются на специально выделенную группу учащихся. Типичный пример — выделение учащихся в классы углубленного изучения предмета.

Индивидуализированное обучение. В этом случае обычно разное содержание учебной работы и разные способы его предъявления (*дифференциация*), а также различный темп учебной работы используются для разных учащихся с учетом их индивидуальных особенностей. Например, учитель приспособливает свою работу, материалы к нуждам отдельного ученика (тренажер, другой учебник, дополнительное время и т.п.)

Лично ориентированное обучение (персонализация). Обычно имеется в виду, что разное содержание учебной работы и разные способы его предъявления (*дифференциация*), а также различный темп учебной работы используются с учетом индивидуальных особенностей учащихся (*индивидуализация*) с ориентацией на их личные интересы, мотивы и жизненные цели.

Табл. 1 поясняет различие между трактовками индивидуализированного (в том числе индивидуального) и персонализированного обучения.

Заметим (рис. 2), что индивидуализированное обучение, как правило, ведут индивидуально (“Обучение попугая”), что совсем не обязательно для персонализированного обучения (“Устный счет”).

Таблица 1. Индивидуализированное и персонализированное обучение

Индивидуализированное обучение	Персонализированное обучение
Конечные цели обучения — общие для всех учащихся	Конечные цели обучения отдельных учащихся могут различаться
Использование различных подходов и методов для развития (формирования) требуемых компетенций	Использование различных подходов и методов для развития (формирования) требуемых компетенций личностного потенциала
Учебную программу определяет педагог	Учащийся включен в определение учебной программы
Ориентация на развитие познавательного потенциала учащегося	Развитие всех видов потенциала учащегося (познавательного, личностного и т.п.)
Развитие способности управлять собственным учением является важной вспомогательной целью	Развитие способности управлять собственным учением является главной целью
Ключевую роль играет учитель	Ключевую роль играет воспитатель (наставник, тьютор)

Переход к индивидуализированному обучению требует существенных изменений в организации работы традиционной массовой школы. Персонализация обучения, в свою очередь, требует дополнительных изменений, которые связаны не только с подготовкой учебных материалов и других средств обеспечения учебного процесса, но и с изменениями в отношениях в учебном коллективе, более глубокого использования инструментов педагогики сотрудничества.

Как мы убедились на практике, традиционное, дифференцированное, индивидуализированное и персонализированное обучение — это дидактические идеализации (модели), которые в реальной жизни не следуют друг за другом и не противостоят друг другу, а могут успешно сосуществовать, дополняя и поддерживая друг друга. Результирующая педагогическая практика в конечном итоге зависит прежде всего от коллективной установки педагогов, учащихся и руководства школы, а также принятых целей и ожидаемых результатов учебной работы.

Содержание обучения информатике в старшей школе

В начале 90-х годов школа № 550 уже имела компьютерный класс³ и позиционировала себя как школа с углубленным изучением информационных технологий. Однако никаких требований к соответствующей начальной подготовке учащихся в связи с этим не предъявлялось. В результате в старших классах, где проводилась наша работа, учащиеся естественно распались на три группы:

- не интересующиеся информатикой (хотя ее изучение обязательно);
- готовые ее изучать, но не считающие это для себя важным;
- интересующиеся информатикой.

Год от года численность этих групп меняется.

Однако в среднем соотношение между группами сохраняется примерно на уровне 20–50–30% от общего количества учащихся в классе.

³ Компьютерный класс появился в школе как дар автономной некоммерческой организации Всемирный ОРТ. Школа продолжает сотрудничать с ОРТ уже более 20 лет.



“Обучение попугая”



“Устный счет”

Рис. 2. Образы индивидуализированного (индивидуального) и лично ориентированного обучения

“Вторая грамотность — это не только умение писать команды для машин, но и воспитание человека, решительного и предусмотрительного вместе”.

“Программирование — вторая грамотность”

А.П. Ершов, Конгресс ИФИП, 1981

эффективности своей учебной и профессиональной работы, заложить у него основы алгоритмического мышления, развить способность решать практические задачи. Каждый выпускник, например, должен уметь пользоваться электронной таблицей со ссылками и условиями, написать инструкцию для обслуживания технологического процесса, спроектировать переходы в меню сайта и т.п.

Сегодня уверенность в приоритетном значении обучения учащихся

программированию укрепились, в том числе, благодаря все более широкому проникновению в нашу жизнь различных программируемых устройств и роботов. Этот процесс затрагивает все развитые страны мира, а обучение школьников программированию все чаще формулируется как приоритетная национальная задача⁴.

Мы считаем, что возрождение в нашей школе интереса к изучению программирования и формированию алгоритмического мышления неизбежно. Это связано и с созданием приложений в социальных сетях, и с программированием смартфонов и планшетов, и с программированием учебных роботов. За тридцать лет с момента появления у нас курса информатики мир информационных технологий качественно преобразился. Облачные вычисления, социальные сети, “интернет вещей” меняют наши представления о жизни, а средства массовой информации активно популяризуют новые ИКТ. В результате общекультурную составляющую традиционного курса информатики естественно обсуждать на уроках истории и обществоведения, а освоение

⁴ Например, руководители США стремятся сделать обучение программированию обязательным в каждой школе. Президентская программа предусматривает выделение на это более четырех миллиардов долларов на три года для создания и распространения учебных материалов, дополнительной подготовки учителей, налаживания партнерства школ с производственными структурами [10].



Рис. 3. Модули учебного курса

информационных технологий — включать во все школьные предметы, начиная с первого класса.

Основной задачей нашего курса является подготовка учащихся на уровне требований образовательных стандартов среднего (полного) общего образования по информатике и информационным технологиям с учетом стандартов нового поколения. Курс рассчитан на изучение в 10–11-х классах школы с углубленным изучением информационных технологий в течение 34 учебных недель в году⁵. Он сочетает классические основы программирования с освоением инженерного подхода к решению возникающих задач (потребность — цель — способ — результат). Традиционное в целом содержание дополнено серией проектных работ (в том числе с использованием Arduino) и средствами систематического формирующего и итогового оценивания.

Модули, блоки и лестница учебного курса

Курс представляет собой последовательность учебных модулей. Каждый модуль, в свою очередь, состоит из двух-трех содержательных блоков (рис. 3).

Освоение учащимися материалов отдельного блока означает достижение запланированных учебных результатов. Все результаты каждого блока сформулированы операционально, в соответствии с требованиями педагогического дизайна [11]. Один блок включает в себя, как правило, от трех до пяти результатов. Каждому блоку соответствует набор учебных материалов, проверочных заданий и примерное время, которое необходимо учащимся для достижения запланированных учебных результатов. Как показала практика, изучение блока требует обычно не более месяца, а освоение модуля в среднем занимает 2–3 месяца.

⁵ С рабочей программой курса можно ознакомиться на сайте школы (<https://portal.ort.spb.ru/>).

Для каждого учебного результата разработаны контрольные материалы. После изучения каждого блока проводится оценка успешности его освоения. Для каждого учебного результата сформированы группы контрольных заданий, которые состоят из не менее чем трех–пяти разнотипных задач. Считается, что учащийся достиг требуемого результата, если он успешно справился как минимум с двумя из трех предложенных заданий.

Освоение каждого учебного модуля завершается итоговым контролем. Здесь оценивается достижение учащимися ключевых целей курса. Предлагаемые задания позволяют оценить, в какой мере учащийся способен решить практическую задачу. Для этого, например, мы предлагаем школьникам продемонстрировать выполнение практической работы с использованием “робота”⁶, который собран на базе Arduino. Нами подготовлен набор заданий, который постоянно продолжает пополняться. Считается, что учащийся справился с заданием, если он, например, активно работал в составе малой группы, которая коллективно выполняла работу, а также успешно выступил перед одноклассниками, объяснив алгоритм решения задачи так, что все присутствующие (включая учителей) признали его объяснение убедительным.

Таким образом, учебный курс можно представить себе в виде лестницы, состоящей из отдельных ступеней (рис. 4).

Каждая ступень имеет “высоту”, “ширину” и “глубину”⁷. “Высота” ступени изображает ожидаемый учебный результат. “Глубина” характеризует время, которое требуется затратить учащимся для достижения этого результата. “Ширину” ступени можно связать с объемом осваиваемого учебного материала. Отдельные ступени лестницы курса могут разли-

⁶ Здесь “робот” — любое программируемое устройство (светофор, сумматор, бегущая строка, управляемая тележка и т.п.).

⁷ Идея лестницы описания образовательных результатов заимствована нами у Д.Бакли (<http://www.danbuckley.net/>).

чатся по любому из параметров. Естественно, что ступени для модулей базового, расширенного и продвинутого уровней отличаются “шириной” ступенек: для базового уровня она минимальна, для продвинутого — максимальна. Программа курса едина, но требования к учебным результатам отдельных школьников могут быть дифференцированы. Учащиеся могут осваивать материалы каждого модуля на базовом, расширенном и продвинутом уровнях. Это позволяет учесть упомянутое выше деление школьников на три группы. Как показывает опыт, это деление не абсолютно. Одни и те же школьники нередко осваивают один модуль на одном уровне, а другой — на другом, в зависимости от того, в какой мере их заинтересовал соответствующий материал.

Таким образом, все учащиеся осваивают один и тот же набор учебных тем, но каждый может выбрать ту или иную глубину ее изучения.

Проиллюстрируем сказанное на примере учебного блока “Сумматор и триггер”. Учебные результаты на базовом уровне здесь формулируются следующим образом:

- Ученик, работая в паре с другим учеником, создает из последовательности снимков экрана компьютера видеоролик, в котором рассказано о разработке логической схемы одноразрядного сумматора (составление таблицы истинности, составление логических функций сумматора, упрощение логических функций, работающая модель в программе Widget WorkShop). Ученик, используя ролик, доходчиво (возможно, с небольшими неточностями) рассказывает одноклассникам о процессе построения и работе одноразрядного сумматора.

- Ученик моделирует в программе Widget WorkShop работу асинхронного триггера и безошибочно указывает учителю то его состояние, в котором происходит хранение информации.

Учебные результаты на расширенном уровне формулируются иначе:

- Ученик, работая в паре с другим учеником, создает из последовательности снимков экрана

компьютера видеоролик, в котором рассказано о разработке логической схемы двухразрядного сумматора (составление таблицы истинности, составление логических функций сумматора, упрощение логических функций [упрощенная схема должна быть меньше исходной минимум на четыре логических элемента], работающая модель в программе Widget WorkShop). Ученик, используя ролик, доходчиво (возможно, с небольшими неточностями) рассказывает одноклассникам о процессе построения и работе двухразрядного сумматора.

- Ученик моделирует в программе Widget WorkShop работу триггера, который имеет логический вход синхронизации, и безошибочно указывает учителю то его состояние, в котором происходит хранение информации.

Учебные результаты на продвинутом уровне формулируются следующим образом:

- Ученик, работая в паре с другим учеником, создает в программе Widget WorkShop модель логической схемы, состоящей из двухразрядного сумматора, состояние выходов которого запоминается регистром памяти, построенным на трех тактируемых триггерах. Ученик демонстрирует учебной группе работу модели.

Внутришкольные нормативы образовательных достижений

Итоговые учебные результаты по курсу информатики являются составной частью внутришкольных нормативов образовательных достижений (ВНОД). За прошедшие годы аналогичные нормативы были разработаны в школе и по другим учебным предметам (курс физики для старшей школы, математика и русский язык для начальной школы). ВНОД — это комплект учебно-методических и измерительных материалов, которые, по нашему мнению, должны охватывать все стороны учебной и воспитательной работы школы. ВНОД служат инструментом управления качеством и позволяют поддерживать результа-

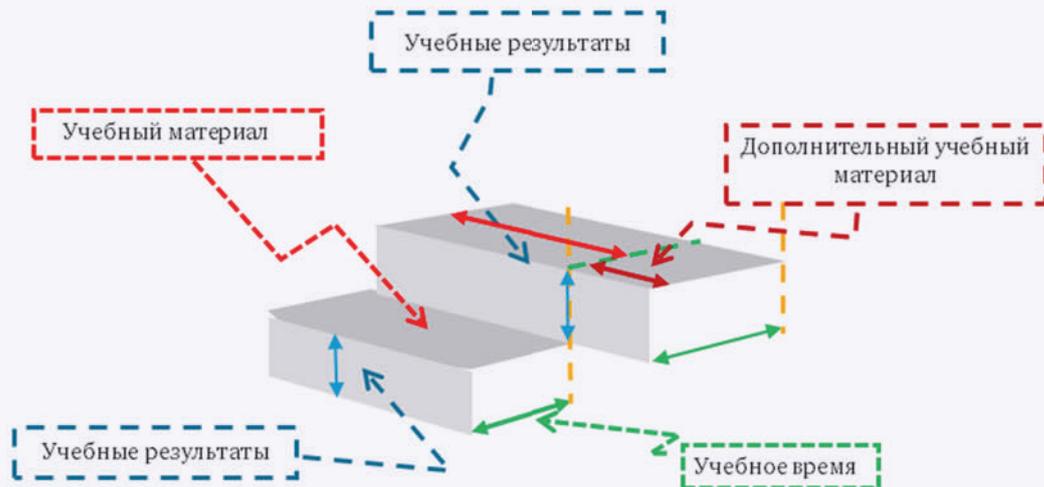


Рис. 4. Ступени лестницы учебного курса

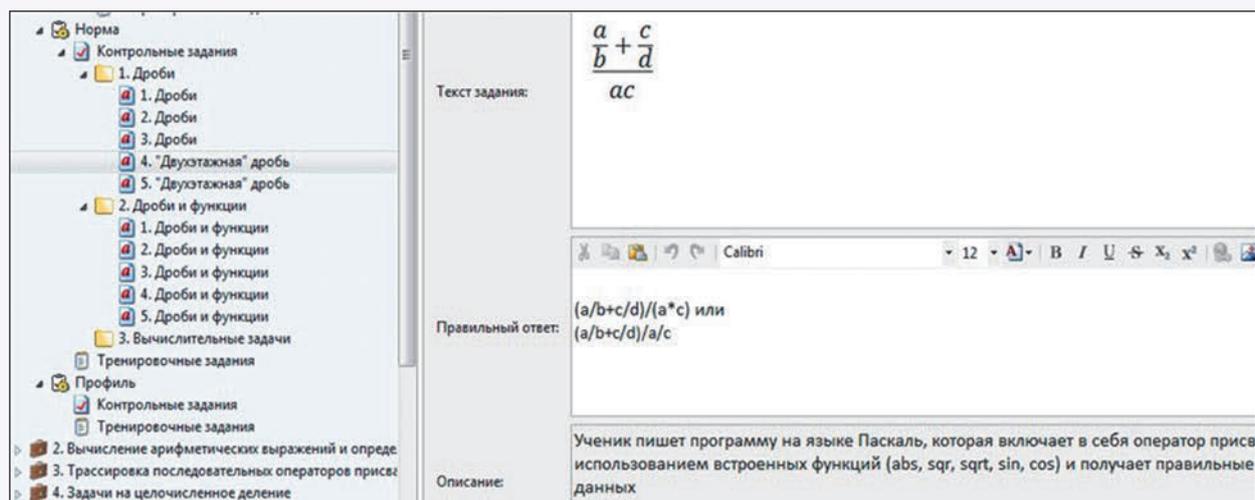


Рис. 5. Фрагмент интерфейса клиентской программы для подготовки ВНОД

тивность учебной и воспитательной работы в школе в заранее определенных границах вне зависимости от организации учебного процесса. Они особенно важны в тех случаях, когда школа хочет ориентировать свою работу на достижение образовательных результатов, а не только на соблюдение принятой методики учебно-воспитательной работы.

Разработка ВНОД — результат коллективных усилий. Они не зависят от отдельного учителя и разрабатываются школьной командой с учетом конкретных условий школы (действующие на местах стандарты и регламенты, образовательные потребности школьников и родителей, квалификация учителей, материальные возможности школы и т.п.). Подготовка ВНОД — составная часть работ по выработке образовательной программы школы, которая помогает сделать работу школы доказательно результативной.

Инструменты для подготовки и использования ВНОД

Идея разработки внутришкольных нормативов образовательных достижений и внутренних стандартов школы не нова. Однако попытки реализовать ее в прошлом наталкивались на непреодолимые трудности подготовки и использования подобных материалов. Эти материалы включают тысячи страниц описания показателей, инструментов оценивания и методических указаний, которыми трудно ежедневно пользоваться на практике. В отличие от промышленного предприятия даже крупная школа не может позволить себе ввести должности главного технолога, главного контролера и создать при них соответствующие службы. Разработка и введение ВНОД невозможны без опоры на современные информационные технологии. К началу нашего проекта в школе № 550 такие условия были созданы при поддержке компании Microsoft, которая предоставила программное обеспечение для создания школьного портала. Для разработки

ВНОД и их использования была развернута специализированная база данных⁸, которая позволила педагогу в режиме онлайн строить лестницы результатов, готовить, хранить и редактировать проверочные (контрольные) задания для каждого учебного результата.

Для каждого учебного результата подбираются задания как минимум трех типов, не менее чем по четыре задания каждого типа. Когда все контрольные задания подобраны и занесены в базу данных, учитель может автоматически сформировать для учащихся контрольные работы в нескольких равноценных вариантах (рис. 5). Доступ к базе данных ограничен. В нее можно войти только с учительских компьютеров с помощью клиентской программы.

Учебные материалы для каждого блока курса, которые необходимы для организации работы школьников, готовятся и хранятся на школьном сервере в библиотеках системы управления обучением MOODLE. Подчеркнем, что это не отдельный курс дистанционного обучения, а постоянно пополняемая библиотека образовательных ресурсов, которую учащиеся используют для работы по курсу как на занятиях в школе, так и дома. Здесь же они выкладывают результаты выполнения своих заданий. Можно сказать, что курс использует модель смешанного обучения, а материалы на школьном сервере позволяют этот курс дифференцировать.

В нашем курсе учитель обязан фиксировать на школьном портале не только корректно выполненные школьником тренировочные и контрольные задания, но и достигнутые им учебные результаты. Эти данные необходимы для формирования динамических учебных групп, определения темпа прохождения курса каждым школьником, характера его учебной траектории, предоставления информации родителям. Кроме того, эти данные важны для совершенствования и развития методического

⁸ Разработана специалистами мультимедиа-студии «Март».

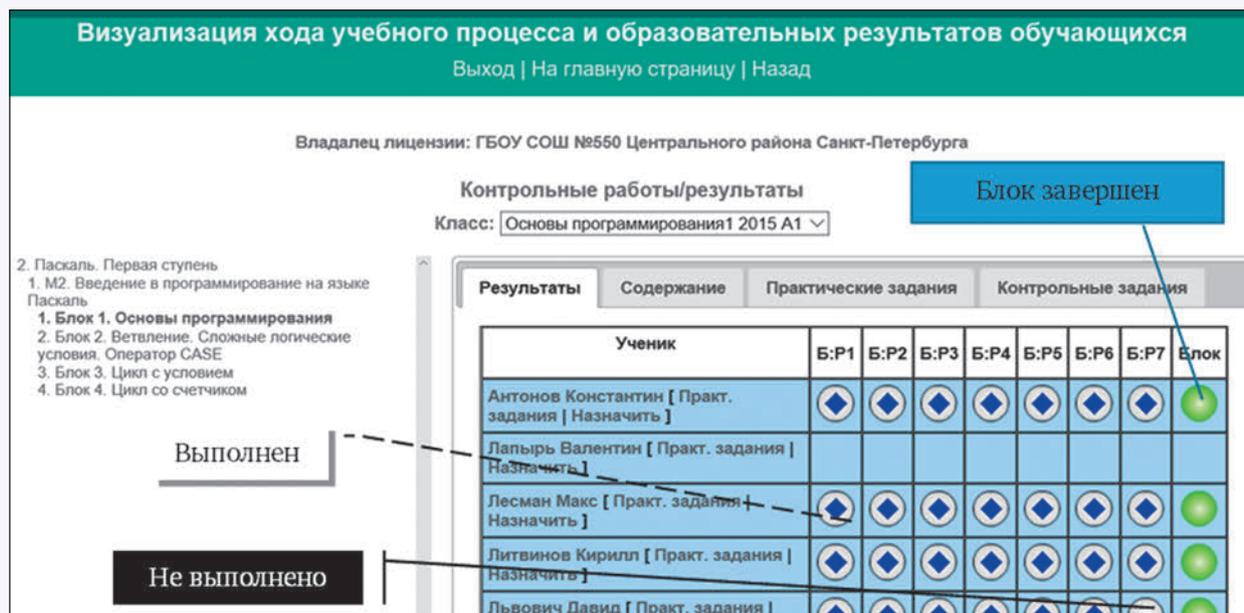


Рис. 6. Фрагмент веб-интерфейса классного журнала нового поколения

каркаса и оценки эффективности предлагаемых учащимся учебных материалов. Для решения этой задачи был разработан специализированный веб-инструмент, который можно назвать электронным классным журналом нового поколения (рис. 6). Сегодня он активно используется для сбора и хранения достигнутых учебных результатов и представления их в графическом виде.

В левом поле журнала (рис. 6) указаны модуль и блоки курса. В правом поле можно увидеть содержание блока с разработанными для него практическими и контрольными заданиями, назначить задания учащимся, а также увидеть ход их выполнения.

Цикл учебной работы школьника

Главная причина, заставившая нас взяться за описываемую разработку, — снижение учебных результатов старшеклассников. Многочисленные попытки изменить методику, подобрать другой учебник или написать его самим, упростить задачи, снизить уровень требований и т.п., которые предпринимала школа в 2004–2008 гг., не давали результата. Самостоятельно написать работоспособный алгоритм решения задачи были не способны более половины школьников. Снижение учебных результатов происходило на фоне резкого падения интереса школьников к изучению информатики. Мы предположили, что основной причиной низких учебных результатов является потеря мотивации. Мы решили, что фронтальная модель учебной работы не подходит для обучения основам программирования в классе, где не было предварительного отбора учащихся. Слабые ученики продвигаются медленно, а сильные — скучают и не получают

того, что могло бы их заинтересовать. Кроме того, учащиеся, вопреки ожиданиям, с которыми они пришли из основной школы, не видят в курсе программирования никакой практической пользы. В итоге и те, и другие теряют интерес к учебе.

Мы предположили, что результативный для каждого школьника курс информатики сможет поддерживать мотивацию к изучению программирования, если каждый ученик сможет работать в своем темпе, а успешность освоения им учебного материала будет постоянно оцениваться. В результате проектирования мы пришли к описываемому варианту личностно ориентированной модели обучения (ЛОМО).

Основной единицей учебной работы на лестнице курса служит учебный блок. Его освоение теоретически (в среднем) длится около месяца. Однако в реальности школьники болеют, пропускают занятия, несвоевременно выполняют тренировочные задания. В результате им не всегда хватает времени для достижения всех запланированных результатов обучения. Следуя доказательно-результативному подходу, мы предлагаем им дополнительный двухнедельный учебный цикл. Этого достаточно, чтобы с учебным материалом справилось абсолютное большинство школьников.

Однако некоторым ученикам для достижения каждого результата необходимо предоставить дополнительный недельный или двухнедельный цикл. Как показывает опыт, ЛОМО позволяет успешно справляться с этими случаями, но от учителя (как воспитателя) здесь требуется дополнительная⁹ методическая и воспитательная работа.

⁹ Для описания этого аспекта работы учителя требуется подготовить отдельную статью (или книгу).

“...наш ребенок бы умел и знал, но он прежде всего не хочет...”
Л.Выготский

В курсе используется традиционная структура базового учебного цикла, которая показана на рис. 7.



Рис. 7. Базовый цикл учебной работы

Он начинается с постановки проблемы (крючок), которая может заинтересовать ученика, мотивировать его к дальнейшей работе. Затем он выполняет учебные задания, которые включают в себя в том числе изучение теоретического материала. Здесь широко используется работа в парах и в малых группах. На третьем этапе учащиеся представляют результаты своей работы группе, учителю или всему классу. Те, кто успешно справился с тренировочными заданиями, выполняют зачетное задание. На заключительном этапе учитель подводит итоги, пытается вдохновить школьника, подготовить его к изучению следующего учебного блока.

Курс содержит большое количество учебных проектов. При их выполнении мы используем “проектные циклы” длительностью от двух до восьми занятий. Структура этих циклов соответствуют модели учебного исследования BSCS 5E [12]. Как показал опыт, эта модель позволяет достаточно успешно поддержать стремление учащихся к творчеству.

О некоторых результатах проекта

Использование ЛОМО при обучении информатике продолжается уже более пяти лет. Каждый год мы проводим среди учащихся оценку удовлетворенности такой работой. Для этого используются “круглые столы”, а также проведение специальных интервью. Главный вопрос формулируется следующим образом: “Помогла ли вам новая лично ориентированная модель обучения при изучении информатики? Если да, то в чем именно? Если нет, то что именно вам мешало?”.

Все выпускники, включая тех, кому ЛОМО сначала казалась очень неудобной и странной, все эти годы отвечают на этот вопрос положительно. Среди главных достоинств ЛОМО они указывают следующие:

- понятное каждому описание целей и результатов обучения;

- прямая связь между достигнутыми результатами и результатами оценивания;
- возможность самостоятельно планировать свое учебное время.

Выпускники нередко говорят, что преподаватели высшей школы высоко оценивают их подготовку по информатике.

Около 20–25% старшеклассников школы ежегодно выбирают ЕГЭ по информатике. Средний балл ЕГЭ у выпускников школы стабильно выше среднего балла ЕГЭ по г. Санкт-Петербург.

Для многих учащихся весьма важным оказался сам опыт учебы в рамках ЛОМО. Многие впервые задумались о том, как можно и нужно учиться. Это проявилось, например, в ходе встречи с методистами города, которые пришли знакомиться с опытом работы школы. На многих заметное впечатление произвела попытка учащихся объяснить своим учителям, как лучше организовать обучение по различным предметам. Появление ЛОМО на информатике раскрыло им глаза на то, что учиться можно по-разному. В результате значительная часть старшеклассников (до 40%) стала завершать изучение курса заметно быстрее, чем это предусмотрено программой.

У большинства учащихся переход к новой модели организации учебной работы не вызвал затруднений. Однако этого нельзя сказать об учителях. В ходе обсуждений мы зафиксировали суть изменений в виде двух тезисов:

- раньше мы учили всех, теперь надо учить каждого: с каждой контрольной работой рано или поздно должны справиться все до единого ученика;
- раньше ответственность за результаты обучения лежала на учителе, теперь большую часть ответственности надо передать ученику и его родителям.

Однородного класса не стало: на занятия приходит, как минимум, две группы учеников — те, кто хочет заниматься программированием, и те, кто не хочет. В результате учитель получает сразу два вызова:

- 1) постоянно придумывать и решать самые различные задачи по программированию;
- 2) найти свое место в новой организации учебного процесса, увидеть себя в новой роли.

Профессиональной подготовки членов школьной проектной команды вполне достаточно, чтобы успешно отвечать на первый вызов. Гораздо больше проблем возникло со вторым.

Для формирования новой культуры учителям приходится готовить учебные и методические материалы, операционализировать учебные результаты. И здесь им мешает недостаточный уровень подготовки в области педагогического дизайна. Для разработки ВНОД по информатике можно привлечь дополнительных консультантов, но материалы к занятиям учителям приходится готовить самим, строго следовать ВНОД и требованиям ЛОМО в рамках доказательно-результативного подхо-

да к обучению. При наличии в классе нескольких групп учеников, которые продвигаются по учебной программе с разным темпом, к каждому уроку приходится готовить несколько разных планов. В условиях экстремальной загруженности учитель вынужден ежедневно формировать новые материалы, приносить их на занятия без редактирования и экспертизы. Естественно, что качество материалов страдает, они расходятся с ВНОД, получить запланированный результат не удастся. Конечно, с течением времени накапливается опыт, ситуация исправляется, но проблема остается. Технологический прогресс не стоит на месте, без регулярного обновления материалов не обойтись, а времени и ресурсов для этого опять недостаточно.

Среди наиболее трудных проблем учителя называют:

- организацию на уроке результативной работы нескольких учебных групп, которые движутся по своим учебным траекториям с разной скоростью;
- согласование оценивания по существующей пятибалльной системе с требованием результативной учебной работы (достиг/не достиг результата);
- выполнение учебной программы отстающими, формирование их учебной мотивации.

Как показал опыт, учителя склонны время от времени произвольно упускать из вида некоторые ключевые компоненты новой модели работы. В итоге модель перестает быть результативной. Поэтому требуется организация постоянной методической поддержки учителей, оказание им помощи в выборе необходимых методических средств. Без формирования в коллективе атмосферы сотрудничества, взаимной поддержки это сделать практически невозможно.

Накопленный опыт позволяет сделать еще один вывод: при всех достоинствах ЛОМО, ее невозможно ввести простым приказом сверху. Для ее освоения необходима многоаспектная трансформация образовательного процесса, которую невозможно провести без подвижнической творческой работы всего педагогического коллектива, сотрудничества родителей и учеников, компетентной методической поддержки.

В заключение отметим, что приведенное в статье описание нашего многолетнего проекта, как минимум, слишком конспективно. Накопленный опыт заслуживает того, чтобы им делиться. Наши следующие публикации на эту тему можно будет найти на страницах журналов “Наука и Школа”, “Информатика и образование” и в Интернете.

Литература

1. Уваров А.Ю. Чему учить на уроках информатики // “Информатика”, № 1 (194), 1999 (<https://inf.1september.ru/1999/art/uv1.htm>).
2. Сенокосов А.И. Виртуальная школа // “Информатика”, № 18 (211), 1999 (<http://inf.1september.ru/1999/art/virt1.htm>).
3. Уваров А.Ю. Пространство задач информатизации школы // “Информатика”, № 23 (386), 2003.
4. Постановление ЦК КПСС и Совета министров СССР от 28 марта 1985 года № 271 “О мерах по обеспечению компьютерной грамотности учащихся средних учебных заведений и широкому внедрению электронно-вычислительной техники в учебный процесс” // Вопросы образования, 2005, № 3. <http://ecsocman.hse.ru/data/2011/01/17/1214868235/22post0.pdf>.
5. Асмолов А.Г., Семенов А.Л., Уваров А.Ю. Новый этап информатизации отечественной школы // Информатика, № 15 (616), 2010.
6. Уваров А.Ю. Как перевести школу в инновационный режим // “Управление школой”, № 2, 2013.
7. Водопьян Г.М. Об опыте освоения инновационных цифровых учебно-методических материалов в общеобразовательной школе / В кн. “Бюллетень лаборатории математического, естественно-научного образования и информатизации”. Московский городской педагогический университет, Научно-исследовательский институт столичного образования, “Научная книга”, 2012, т. 2.
8. A Guide for Bringing the SAMR Model to iPads. Available: <https://www.edsurge.com/news/2015-02-06-a-guide-for-bringing-the-samr-model-to-ipads>.
9. Теньчук И. Лекция “Индивидуализация образования” Т.М. Ковалевой. Общественные обсуждения “эффективности введения ФГОС”, г. Томск, 23.03.2012. Инновационная образовательная сеть “Эврика”. <http://www.eurekanet.ru/ewww/info/16174.html>.
10. The Reality of Coding Classes. <http://www.theatlantic.com/education/archive/2016/02/obamas-push-for-computer-science-education/459276/>.
11. Никитин Н.В., Уваров А.Ю. Телекоммуникация, образование, профессионализм. М.: Логос, 2008.
12. The BSCS 5E Instructional Model: Origins, Effectiveness and Applications. BSCS, 2006. http://bscs.org/sites/default/files/_media/about/downloads/BSCS_5E_Executive_Summary.pdf.



четырёх байт, причем каждый байт записывается в виде десятичного числа. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. Например, если IP-адрес узла равен 231.32.255.131, а маска равна 255.255.240.0, то адрес сети равен 231.32.240.0.

Для узла с IP-адресом **224.128.112.142** адрес сети равен **224.128.96.0**. Чему равен третий слева байт маски? Ответ запишите в виде десятичного числа.

Собственно, текст задачи занимает всего две строки, выделенные выше жирным шрифтом. Как же ее решать?

Если поискать в Интернете, то описание решения найти несложно — как на том же портале “Решу ЕГЭ”, так и на сайте К.Полякова. Но... как правило, эти объяснения — чисто текстовые, недостаточно наглядные, а потому учащиеся не сразу “схватывают” суть. Поэтому ниже предлагается “типовое” объяснение решения этой задачи с использованием наглядных схем, что делает это решение легкопонятным и доступным для запоминания.

ЕГЭ

ЕГЭ: новый “маскарад в Сети”

О.Б. Богомолова,
д. п. н., учитель
информатики
и математики
ГБОУ СОШ № 1360,
Восточный округ
г. Москвы,

Д.Ю. Усенков,
Московский
государственный
институт
индустрии
туризма,
Москва

44

► Задачи с масками IP-адресов, когда по значению IP-адреса компьютера и значению IP-маски требуется определить адрес сети, адрес компьютера в сети или же количество компьютеров в сети, уже хорошо знакомы учащимся (см., например: Богомолова О.Б. Информатика. Новый полный справочник для подготовки к ЕГЭ. М.: АСТ, Астрель, 2015). Но недавно появились новые разновидности такой задачи, которые, хотя в общем-то так же несложны, все же вызывают поначалу трудности у школьников.

Задача (задание № 7696, образовательный портал “Решу ЕГЭ”, <http://inf.reshuege.ru/test?pid=7696>).

В терминологии сетей TCP/IP маска сети — это двоичное число, меньшее 2^{32} ; в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места нули. Маска определяет, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес, — в виде

Решение

1. Сравниваем оба адреса IP по байтам **слева направо**:

$\xrightarrow{\hspace{1.5cm}}$
 224. 128. 112. 142
 224. 128. 96. 0

Первые два байта одинаковы. Разница — в третьем байте слева (112 или 96), ну и, конечно, в четвертом байте, но он нас и не интересует.

2. Переводим оба числа (третьих байта) в двоичную систему счисления, обязательно **дополняя двоичное число слева нулями до 8 битов**:

$112_{10} = 01110000_2$
 $96_{10} = 01100000_2$

3. Подписав полученные двоичные числа друг под другом, сравниваем их по битам **слева направо**, при этом для одинаковых битов в результате пишем биты “1”, а начиная с первой же несовпадающей пары битов — пишем “0”:

0	1	1	1	0	0	0	0
0	1	1	0	0	0	0	0
1	1	1	0	0	0	0	0

4. Получили двоичное значение третьего байта маски: **11100000**.

5. Преобразуем его в десятичное число:
 $11100000_2 = 224_{10}$.

Ответ: 224.



Общероссийский проект Школа цифрового века

Издательский дом «ПЕРВОЕ СЕНТЯБРЯ»

2016/17 учебный год

Материалы проекта:

- предметно-методические журналы
- вебинары
- курсы повышения квалификации
- электронные книги и методические брошюры
- электронные учебники (ЭУ) для учителей и учеников

Стоимость участия в проекте:

Стандартный вариант*
(все материалы, исключая ЭУ)



12 тысяч рублей от школы
за весь учебный год независимо
от количества учителей в школе

Расширенный вариант**
(все материалы, включая ЭУ)



18 тысяч рублей от школы
за весь учебный год независимо
от количества учителей в школе

* При оплате до 30 июня 2016 года – 10 тысяч рублей от школы

** При оплате до 30 июня 2016 года – 16 тысяч рублей от школы

Подробная информация и регистрация на сайте:

digital.1september.ru

Участие в проекте общеобразовательной организации и педагогических работников удостоверяется соответствующими документами.



13 лет вместе

“В мир информатики”, последний выпуск которого вы, уважаемый коллега, читаете, — первое и единственное в мире специализированное “издание” по информатике для учащихся. Его публикация была начата в сентябре 2003 года по инициативе главного редактора тогда еще газеты “Информатика” С.Л. Островского. Я хорошо помню тот майский день, когда Сергей Львович предложил мне стать редактором этого раздела. В ответ на его предложение я попросил время подумать. А когда приехал домой, уже получил письмо: “Дмитрий Михайлович! Спасибо, что согласились ☺” (Сергей Львович хорошо знал меня и понимал, что эта работа — “для меня”).

Когда жена узнала о моей новой работе, она сказала: “Ты сел на иглу”. А Станислав Ильич Перверзев из Московского института открытого образования определил работу так: “Это все равно, что купить корову”. Дело в том, что нужно было каждую неделю (“Информатика” была еженедельной), независимо ни от каких-либо обстоятельств, “выдавать” шесть страниц материала формата А4 в нескольких рубриках. А авторов, которые предлагали бы материал для публикации в разделе для учащихся, на первом этапе не было. Приходилось быть и автором, и редактором, и секретарем, и, так сказать, “жюри конкурсов” (коллеги со стажем, наверное, помнят наш знаменитый конкурс № 41 “Чемпионат СНГ по переливанию воды ☺”, проходивший в 6 туров и в котором было предложено решить 64 задачи на переливание; в этом конкурсе приняли участие более 200 учащихся). Да, было трудно. Но помогало чувство, что “В мир информатики” нужен и полезен. Помогали и поддерживали письма учителей, благодаривших за работу, письма ребят с ответами и решениями (“Мама уехала в Белгород, а нам принесли «Информатику», и я начал решать задачи”), среди которых попадались и такие: “...что соответствует 121₅, или, **по-русски**, — 36”. Меня и редколлегию ребята поздравляли с Новым годом и с другими праздниками, с нами делились радостными новостями (“Я стала чемпионом Башкортостана

среди школьников по художественной гимнастике”, — сообщала Галия Гайсина из Уфы; кстати, к миру информатики Галию “приобщил” ее старший брат Рашит, также активный участник наших конкурсов).

Поэтому могу уверенно сказать, что, несмотря на число в названии данной заметки, время работы с разделом “В мир информатики” считаю для себя счастливым.

В заключение хотел бы назвать коллег (возможно — не всех), которые на протяжении многих лет не жалели времени, чтобы ознакомить своих учеников с материалами раздела, помочь ребятам в решении и в оформлении ответов:

— Артамонову Викторию Владимировну, средняя школа поселка Новопетровский Московской обл.;

— Бабаеву Наталью Александровну, Тюменская обл., Ямало-Ненецкий автономный округ, г. Ноябрьск, школа № 2;

— Багаутдинову Зилу Назиповну, Республика Башкортостан, г. Стерлитамак, школа № 31;

— Багаутдинову Р.Р., Республика Татарстан, г. Казань, МБОУ “Татарская гимназия № 1”;

— Багрову Ольгу Алексеевну, средняя школа г. Пионерский Калининградской обл.;

— Белову И.Л., Санкт-Петербург, школа № 639;

— Богданову Ларису Михайловну, Республика Карелия, поселок Надвоицы, школа № 1;

— Болдыреву Светлану Владимировну, Республика Башкортостан, г. Уфа, гимназия № 3 им. А.М. Горького;

— Брунова Александра Сергеевича, средняя школа села Ириновка, Новобураский р-н Саратовской обл.;

— Валиеву Данию Ильясовну, средняя школа села Сулеево им. Р.Г. Галеева, Республика Татарстан, Альметьевский р-н;

— Валиеву Раушанию Нурлимановну, Чистопольско-Высельская средняя школа, Республика Татарстан, Чистопольский р-н;

— Васнину Ольгу Васильевну, Костромская обл., Буйский р-н, г.п.п. Чистые Боры, школа № 1;

— Воеводину Розу Васильевну, Чувашская Республика, г. Канаш, Канашский педагогический колледж;

— Волкова Юрия Павловича и Волкову Татьяну Петровну, Владимирская обл., г. Струнино, школа № 11;

— Воронкову Ольгу Борисовну, Ростовская обл., г. Новочеркасск, Суворовское училище МВД РФ;

— Воронову Тамару Николаевну, Алтайский край, г. Рубцовск, школа № 1;

— Гагарину Эльвиру Васильевну, Республика Карелия, г. Сегежа, школа № 6;

— Гаврилову Марину Ивановну, г. Пенза, школа № 512;

— Галиеву Рузилю Ферролевну, Республика Татарстан, Актанышский р-н, село Актаныш, средняя школа № 1;

— Гильзер Н.В., Республика Башкортостан, г. Уфа, лицей № 60;

— Гилязову Гульнару Маликовну, Республика Татарстан, Актанышский р-н, село Актаныш, средняя школа № 2;

— Гранаткину Ольгу Михайловну, Республика Коми, г. Сыктывкар, МОУ “Лицей народной дипломатии”;

— Гриднева Александра Борисовича, Вознесенская основная школа Красногвардейского р-на Оренбургской обл.;

— Дибирова Магомеда Дибировича, Республика Дагестан, г. Махачкала, лицей № 3;

— Дмитриеву Ольгу Валентиновну, г. Стерлитамак, Республика Башкортостан, школа № 12;

— Долгову Галину Александровну, средняя школа села Восточное Нижегородской обл.;

— Дубовицкую Наталью Вячеславовну и Павлову Татьяну Анатольевну, г. Новокузнецк Кемеровской обл., гимназия № 44;

— Дудину Татьяну Валентиновну, г. Архангельск, школа № 9;

— Евдокимову Алевтину Ивановну, средняя школа поселка Осиновка, Алтайский край;

— Жарикову Елену Николаевну, Совхозная средняя школа, Московская обл., Серебряно-Прудский р-н, поселок Успенский;

— Загафуранову Альбину Фингатовну, средняя школа села Сейтяково Балтачевского р-на, Республика Башкортостан;

— Задубровского В.Ф., Сапожковская средняя школа имени Героя России А.И. Тучина, Рязанская обл.;

— Зорину Елену Михайловну, Санкт-Петербург, г. Зеленогорск, лицей № 445;

— Зудина Василия Павловича, Ардатовский коммерческо-технический техникум, поселок Ардатов Нижегородской обл.;

— Игошеву Анну Аркадьевну, Свердловская обл., г. Ревда, школа № 10;

— Искандарову Айгуль Римовну, Республика Башкортостан, г. Уфа, школа № 18; школа № 54 (Центр дистанционного обучения);

— Казанцеву Оксану Вячеславовну, Республика Коми, г. Усинск, Центр дополнительного образования детей, “Школа программистов”;

— Каликину Татьяну Валерьевну, Республика Карелия, поселок Надвоицы, школа № 1;

— Келлер Елену Викторовну, г. Прокопьевск Кемеровской обл., школа № 32;

— Клопову Елену Владимировну, средняя школа села Дохновичи Брянской обл.;

— Комбарову Светлану Ивановну, г. Воронеж, лицей № 2;

— Кордину Нину Евгеньевну, Смоленская обл., г. Демидов, школа № 1;

— Корнееву Марину Викторовну, Краснодарский край, г. Приморско-Ахтарск, школа № 22;

— Красненкову Любовь Алексеевну, средняя школа поселка Ерофей Павлович, Амурская обл., Сковородинский р-н;

— Куликову Т.Б., Московская обл., г. Мытищи, школа “Логос М”;

— Липину Веру Мурадовну, Москва, школа № 827;

— Матыкина Вячеслава Юрьевича, Воронежская обл., г. Новохоперск, гимназия № 1;

— Меньшикова Виталия Владимировича, Республика Карелия, г. Сегежа, школа № 5;

— Муравьеву Ольгу Викторовну, средняя школа деревни Муравьево, Вологодская обл.;

— Мусатову Ирину Борисовну, Челябинская обл., г. Златоуст, школа № 9;

— Павлову Наталью Николаевну, Красноярский край, г. Канск, школа № 5;

— Пичугина В.В., средняя школа рабочего поселка Пинеровка, Саратовская обл., Балашовский р-н;

— Пучкину Светлану Александровну, Республика Башкортостан, г. Стерлитамак, гимназия № 5;

— Пышную Елену Александровну, Кабардино-Балкарская Республика, г. Нальчик, школа № 32;

— Родикову Раису Дмитриевну, г. Смоленск, школа № 29;

— Рябченко Наталью Родионовну, Ставропольский край, Кочубеевский р-н, станица Барсуковская, школа № 6;

— Сенюта Елену Ивановну, г. Фрязино Московской обл., школа № 4;

— Сергеева Сергея Александровича, Москва, кадетская школа-интернат № 5 “Преображенский кадетский корпус”;

— Тимофееву Ирину Анатольевну, основная школа села Именево, Чувашская Республика, Красноармейский р-н;

— Тишинину Яну Николаевну, г. Ярославль, гимназия № 1;

— Товмасына Мурада Грантовича, средняя школа села Урман, Республика Башкортостан, Иглинский р-н;

— Толмачеву Н.П., г. Рубцовск Алтайского края, школа № 1;

— Цикину Елену Николаевну, г. Ярославль, школа № 33;

— Цыганкову Ию Юрьевну, Вадьковская средняя школа, Брянская обл., Погарский р-н;

- Чапкевич Ирину Михайловну, г. Орел, лицей № 4 им. Героя Советского Союза Г.Б. Злотина;
- Чернову Ларису Ивановну, средняя школа села Сердар, Республика Марий Эл;
- Чернышеву Ирину Анатольевну, г. Воронеж, школа № 5 им. К.П. Феоктистова;
- Широкову Ларису Владимировну, г. Волгоград, лицей № 9;
- Шитову Любовь Алексеевну, средняя школа села Горелово Тамбовской обл.;
- Шишигину Ольгу Владимировну, Куминская средняя школа, Тюменская область, Ханты-Мансийский автономный округ — Югра, Кондинский р-н;
- Шнейдер Елену Вильгельмовну, Москва, школа № 1399;

- Щербакову Ольгу Борисовну, рабочий поселок Яя Кемеровской обл., школа № 2;
 - Юртаеву Галину Юрьевну, г. Челябинск, школа № 124;
 - Ямкину Елену Владимировну, г. Ульяновск, школа № 37;
 - Ярцева Виктора Александровича, Свердловская обл., Красноуфимский р-н, Тавринская средняя школа;
 - Ярцеву Ольгу Владимировну, г. Ярославль, школа № 33.
- Спасибо всем!

Ваш Дмитрий Михайлович Златопольский

ЗАДАЧНИК

Ответы, решения, разъяснения к заданиям, опубликованным в декабрьском выпуске “В мир информатики”

Задача “Лесная школа танцев”

Напомним условие: “В лесную школу танцев пришли слон, волк и лев. Партнершами у них были выбраны мышшка, белочка и лисичка. Учитель-жираф должен расставить их в пары. Сколько вариантов составления есть у него, если белочка боится, что ее съест волк, а слон — что он раздавит мышку?”

Решение

Можно составить схему, на которой изобразить все возможные варианты формирования пар танцоров.



Примечание. Приведена схема, подготовленная Алиной Сучок, ученицей школы г. Пионерский Калининградской обл., учитель **Багрова О.А.**

Из нее следует, что возможны семь вариантов пар.

Можно также рассуждать так. Каждый из трех “кавалеров” может танцевать с каждой из “дам”, то есть всего возможны $3 \times 3 = 9$ вариантов формирования пар. Но две пары (белочка и волк, мышшка и слон) составить нельзя. Значит, можно — $9 - 2 = 7$ вариантов пар.

Кроме Алины, ответы представили:

- Абдувахидова Алина, Абдувахидова Софья, Базанов Илья, Милушкин Дмитрий, Фролова Александра и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;
- Авдеенко Дмитрий, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;
- Бардина Анастасия, г. Ярославль, гимназия №1, учитель **Тишинина Я.Н.**;

- Баркалова Екатерина, Бражникова Ксения, Булгаков Владислав, Воронков Андрей, Гулевских Анна, Котюхова Альбина, Попов Тимур и Чурикова Ирина, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

- Береза Антонина и Трофимов Сергей, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

- Бородюк Анна и Василенко Татьяна, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

- Дощик Константин, Лазуренко Глеб и Назаркина Татьяна, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

- Лазаренко Нина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

- Лобанов Артем, г. Орел, лицей № 4 им. Героя Советского Союза Г.Б. Злотина, учитель **Чапкевич И.М.**;

- Лютикова Александра, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

- Мекле Иван, Мусатов Тимофей и Трофимова Наталья, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

- Новикова Марина, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

- Новосельцева Александра, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

- Орлова Анна, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

- Парамонов Павел, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**

Головоломка “Получить верное равенство”

Напомним, что было предложено, переложив одну спичку, из приведенного ниже неверного равенства получить верное.



Решение

Надо сделать так, чтобы получилось равенство $7 - 1 = 6$.

Правильные ответы прислали:

- Авдеенко Дмитрий, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;
- Базанов Илья, Гурская Александра, Добрыднева Ксения, Милушкин Дмитрий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;
- Бардина Анастасия, г. Ярославль, гимназия № 1, учитель **Тишининова Я.Н.**;
- Баскакова Ирина и Маркина Светлана, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;
- Баукова Татьяна, Булатов Владимир, Базарнова Наталья, Мекле Иван, Мусатов Тимофей, Пегачкова Мария и Трофимова Наталья, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;
- Дегтярь Анатолий и Новиченко Владимир, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;
- Красненков Александр, средняя школа поселка Ерофей Павлович, Амурская обл., Сквородинский р-н, учитель **Красненкова Л.А.**;
- Лагоша Артем, Республика Коми, г. Усинск, Центр дополнительного образования детей, “Школа программистов”, педагог **Казанцева О.В.**;
- Лютикова Александра, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;
- Мазаева Софья, Огнев Ярослав, Хоченкова Дарья, Худокормова Мария и Чекалин Егор, г. Орел, лицей № 4 им. Героя Советского Союза Г.Б. Злоти-на, учитель **Чапкевич И.М.**;
- Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;
- Орлова Анна, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;
- Прохоров Макар, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;
- Трофимов Сергей, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

Задача “Сказочные гуси”

Напомним, что в задаче речь шла о сказке, в которой над озерами летели гуси. На каждом озере садилась половина гусей и еще полгуся, остальные летели дальше. Все гуси сели на семи озерах. Нужно было определить, сколько было гусей.

Решение

Составлять уравнение (обозначив искомое исходное число гусей x) нецелесообразно — уравнение будет сложным. Лучше начать решение, так сказать, “с конца”.

После седьмого озера, на котором села половина прилетевших гусей и еще полгуся, дальше никто не полетел. Значит, на это озеро с шестого озера прилетел один гусь. Определим, сколько гусей

“прибыли” с пятого озера на шестое. Обозначив это количество x , можем записать:

$x/2$	–	$1/2$	=	1
<i>Осталось над пятым озером после того, как села половина гусей</i>		<i>и еще полгуся</i>		<i>Полетели дальше к пятому озеру</i>

— откуда $x = 3$.

И вообще, зная, сколько гусей улетели с некоторого озера, можно определить, сколько их прилетело на него. Для этого надо к числу улетевших прибавить полгуся и сумму удвоить. (Убедитесь в этом правиле!)

Продолжая такие расчеты, можно получить, что на первое озеро прилетели 127 гусей.

Можно также определить, сколько гусей осталось на том или ином озере, а затем просуммировать эти количества.

Правильные ответы представили:

- Абдувахидова Алина, Абдувахидова Софья и Базанов Илья, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;
- Антипов Анатолий, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;
- Баркалова Екатерина, Бражникова Ксения, Булгаков Владислав, Воронков Андрей, Гулевских Анна, Котюхова Альбина, Попов Тимур и Чурикова Ирина, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;
- Волохов Дмитрий, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;
- Дегтярь Анатолий, Новиченко Владимир и Усачева Екатерина, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;
- Ещеркина Виктория и Кейлер Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;
- Зубов Владислав, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;
- Лютикова Александра и Пурышева Светлана, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;
- Маркина Светлана и Рыжиков Артем, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;
- Прохоров Макар, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;
- Тимофеев Иван, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**

Кроссворд

Ответы

По горизонтали: 3. Плата. 7. Вирус. 9. Автор. 10. Триггер. 11. Моделирование. 14. Логика. 16. СМИ. 18. Трафик. 20. Дек. 21. Исток. 22. Бод. 23. Призма. 24. Курсив. 28. Вариант. 29. Порядок.

По вертикали: 1. Ввод. 2. Пролог. 4. Алгоритм. 5. Чтение. 6. Граф. 8. Стрелка. 9. Арбалет. 12. Про-

бел. 13. Список. 15. Клавиша. 16. Состав. 17. Иголка. 19. Регистр. 23. Порт. 25. Вода. 26. Два. 27. Икс.

Ответы прислали:

— Аникин Степан, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Брызгалов Артем и Артемьев Антон, средняя школа поселка Ерофей Павлович, Амурская обл., Сковородинский р-н, учитель **Красненкова Л.А.**;

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Добродеев Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Докучаева Валерия, Куминская средняя школа, Тюменская область, Ханты-Мансийский автономный округ — Югра, Кондинский р-н, учитель **Шишигина О.В.**;

— Донцов Виктор, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Дощик Константин, Лазуренко Глеб, Назаркина Татьяна и Сучок Алина, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Дубовик Илья, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Ещеркина Виктория, Кейлер Виктория, Малых Кирилл, Матвеева Диана и Сидоренкова Анастасия, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Карамышев Константин, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Мигурская Анастасия и Пранайтис Анастасия, г. Ярославль, гимназия № 1, учитель **Тишинина Я.Н.**;

— Чуркин Сергей, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

Задача “Три студента”

Ответ приведен в таблице (задача решается с помощью трех таблиц):

Фамилия	Из города	Факультет	Увлекается
Андреев	Тверь	иностранных языков	футболом
Борисов	Калязин	физико-математический	баскетболом
Воронов	Вышний Волочёк	исторический	волейболом

Правильный ответ прислали:

— Абдувахидова Алина, Абдувахидова Софья, Базанов Илья, Милушкин Дмитрий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Аникин Степан и Воронова Екатерина, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Баркалова Екатерина, Бражникова Ксения, Булгаков Владислав, Воронков Андрей, Гулевских Анна, Котюхова Альбина, Попов Тимур и Чурико-

ва Ирина, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Бородюк Анна и Василенко Татьяна, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Васильева Мария, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Громова Елизавета, г. Ярославль, гимназия № 1, учитель **Тишинина Я.Н.**;

— Добродеев Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Дубовик Илья, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Ещеркина Виктория и Кейлер Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Лазаренко Нина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

— Мекле Иван, Терешкова Елена и Трофимова Ксения, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

— Мигурская Анастасия, Пранайтис Анастасия и Шахвердова Дилара, г. Ярославль, гимназия № 1, учитель **Тишинина Я.Н.**;

— Назаров Алексей, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Новосельцева Александра, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

— Сучок Алина, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**

Головоломка “Три кучки спичек”

Напомним, что требовалось, имея три лежащие на столе кучки спичек (из одиннадцати, семи и трех спичек) и перекладывая спички из одной кучки в другую по определенному правилу, сделать так, чтобы в каждой кучке было бы по восемь спичек (это возможно, так как общее число спичек — 24 — делится на три без остатка). Правило следующее: к любой кучке разрешается дополнять ровно столько спичек, сколько в ней есть. Например, если в кучке шесть спичек, то и добавлять к ней можно только шесть, если в кучке четыре спички, то и добавлять к ней можно только четыре.

Решение

Первым ходом (перекладыванием) к первой кучке добавлять спички нельзя ни из какой другой, а ко второй — из третьей. Значит, возможны варианты первого хода:

- 1) из первой кучки — ко второй;
- 2) из первой кучки — к третьей;
- 3) из второй кучки — к третьей.

В последнем варианте после первого хода во второй кучке останется всего одна спичка, что явно

потребуется большого числа последующих ходов. Поэтому его рассматривать не будем.

Ситуация после каждого из двух первых вариантов показана в таблицах:

1)

Кучка	Начальное состояние	Первый ход
Первая	11	4
Вторая	7	14
Третья	6	6

2)

Кучка	Начальное состояние	Первый ход
Первая	11	5
Вторая	7	7
Третья	6	12

Добавим в эти две таблицы ситуацию после последнего хода:

1)

Кучка	Начальное состояние	Первый ход	...	После последнего хода
Первая	11	4		8
Вторая	7	14		8
Третья	6	6		8

2)

Кучка	Начальное состояние	Первый ход	...	После последнего хода
Первая	11	5		8
Вторая	7	7		8
Третья	6	12		8

Нетрудно понять, что после предпоследнего хода в каждой кучке должно быть четное число спичек, причем в одной из кучек должно находиться четыре спички, в другой — восемь, в оставшейся — двенадцать (убедитесь в этом!). Этот важный вывод наводит на мысль, что первый вариант предпочтительнее. Анализ этого варианта показывает, что из двух возможных ситуаций после предпоследнего хода:

Кучка	Начальное состояние	Первый ход	...	После предпоследнего хода	После последнего хода
Первая	11	4		4	8
Вторая	7	14		12	8
Третья	6	6		8	8

и

Кучка	Начальное состояние	Первый ход	...	После предпоследнего хода	После последнего хода
Первая	11	4		4	8
Вторая	7	14		8	8
Третья	6	6		12	8

— последняя ситуация приводит к решению за минимальное число ходов (перекладываний):

Кучка	Начальное состояние	Первый ход	Второй ход	Третий ход
Первая	11	$11 - 7 = 4$	4	$4 + 4 = 8$
Вторая	7	$7 + 7 = 14$	$14 - 6 = 8$	8
Третья	6	6	$6 + 6 = 12$	$12 - 4 = 8$

Ответы представили:

— Антоненко Кирилл, Блошук Александр, Истомина Полина, Топычканов Денис и Трофимова Ксения, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

— Базанов Илья, Милушкин Дмитрий, Пыров Егор и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Бардина Анастасия, Мигурская Анастасия, Пранайтис Анастасия и Шахвердова Дилара, г. Ярославль, гимназия № 1, учитель **Тишинина Я.Н.**;

— Васильева Мария и Кулиджанов Рустам, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Волохов Дмитрий, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Дубовик Илья, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Лежнева Александра, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Мазаева Софья, Хоченкова Дарья и Худокормова Мария, г. Орел, лицей № 4 им. Героя Советского Союза Г.Б. Злотина, учитель **Чапкевич И.М.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

— Орлова Анна, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Товмасын Лиана и Широков Дмитрий, средняя школа села Урман, Республика Башкортостан, Иглинский р-н, учитель **Товмасын М.Г.**;

— Торопов Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Шальнева Ирина, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

Заметим, что в ряде ответов задача решалась за большее число ходов.

Головоломка “Переместить карточки”

Напомним, что требовалось, перемещая восемь карточек с буквами, образующих слово ЛЕПЕСТОК, получить слово ТЕЛЕСКОП. Задача решается за 48 перемещений.

Ответы (с большим числом перемещений) представили:

— Антипов Анатолий, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Бардина Анастасия и Пранайтис Анастасия, г. Ярославль, гимназия № 1, учитель

Тишининова Я.Н. (решение Анастасии Пранайтис — наиболее близкое к оптимальному);

— Бердыева Алия, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Добродеев Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Назаров Алексей, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Шаров Иван, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Юрасов Даниил, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

Ответы на вопросы, связанные с поиском информации (“Про птиц и собаку”), прислали:

— Абдувахидова Алина и Абдувахидова Софья, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Авдеев Даниил, Блюденев Кирилл, Долгова Анна, Кокорева Виктория, Недикова Алиса и Раджабова Алина, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Бардина Анастасия, Пранайтис Анастасия и Шахвердова Дилара, г. Ярославль, гимназия № 1, учитель **Тишининова Я.Н.**;

— Воронова Екатерина, Зубков Андрей и Широкова Анна, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Герасимова Наталья и Костина Евгения, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Глушков Кирилл, Голубева Илона и Истомина Полина, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

— Евграфова Ксения, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Ещеркина Виктория и Кейлер Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Измайлова Екатерина, Огнев Ярослав и Чекалин Егор, г. Орел, лицей № 4 им. Героя Советского Союза Г.Б. Злотина, учитель **Чапкевич И.М.**;

— Кобзарь Роман, Лазуренко Глеб, Луцук Максим, Назаркина Татьяна и Чунин Павел, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Лазаренко Нина, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Ливанова Инна и Харитоновна Мария, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**;

— Новосельцева Александра и Тимофеев Иван, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

— Шаламов Алексей, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

Отметим ответы учащихся средней школы г. Пионерский Калининградской обл., приведших источники информации, и лицея № 4 г. Орла, снабдивших их красочными иллюстрациями.

Задача “В городе Обыворпол”

Напомним условие: “В городе Обыворпол живут только обыватели, воры и полицейские (поэтому он так и называется). Полицейские всегда врут обывателям, воры — полицейским, а обыватели — ворами. Во всех остальных случаях жители Обыворполя говорят правду. Однажды несколько обыворполцев водили хоровод и каждый сказал своему правому соседу: «Я — полицейский». Сколько обывателей было в этом хороводе?”

Решение

Допустим, что в хороводе есть хотя бы один обыватель. Слева от него могут находиться его “однопартиец”, вор или полицейский. Но первые два скажут обывателю о себе правду, то есть не могут заявить: “Я — полицейский”. А полицейский в таком случае соврет и также этого не скажет. То есть при нашем допущении должен быть хотя бы один обыворполец, который не скажет о себе: “Я — полицейский”. А поскольку по условию так сказали все участники хоровода, то в нем обывателей не было.

Правильные ответы представили:

— Абаева Ольга, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Абдувахидова Алина, Абдувахидова Софья, Базанов Илья и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Авдеев Николай, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Антипов Анатолий, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Баркалова Екатерина, Бражникова Ксения, Булгаков Владислав, Воронков Андрей, Гулевских Анна, Котюхова Альбина, Попов Тимур и Чурикова Ирина, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Дощик Константин, Назаркина Татьяна и Сучок Алина, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Дунаева Ксения, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Ещеркина Виктория и Кейлер Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Лежнева Александра, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Шаламов Алексей, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

Задача “Робот и книга”

Напомним, что в задаче речь шла о роботе и была приведена программа для него из 13 команд.

Ответы

В результате выполнения программы робот возьмет книгу, развернется, сделает шесть шагов и опустит книгу.

Приведенную программу можно записать короче, если использовать так называемый “цикл” (повторение команд):

1. Вытянуть правую руку.
2. Разжать пальцы правой руки.
3. Сжать пальцы правой руки (предполагается, что между командами 2 и 3 мы положили книгу в руку робота).

4. Опустить правую руку.

5. Повернуться на 180°.

6. Повторить три раза:

Поставить правую ногу впереди левой.

Поставить левую ногу впереди правой.

7. Вытянуть правую руку.

8. Разжать пальцы правой руки.

В системе команд робота — семь команд.

Ответы представили:

— Абаева Ольга, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Авдеенко Николай, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Баков Константин, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Зубков Андрей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Зубов Владислав, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Новосельцева Александра и Тимофеев Иван, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

— Потапенко Андрей, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Шаламов Алексей, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

Японские головоломки sudoku, опубликованные в декабрьском выпуске “В мир информатики”, решили:

— Авдеенко Николай, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Ануфриева Дарья, Мусатов Тимофей, Плотникова Валерия, Сатыбалдина Ксения и Цепилова Мария, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

— Баков Константин, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Башлыков Алексей, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Дощик Константин и Сучок Алина, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юргаева Г.Ю.** (Виктория нашла два решения первой головоломки);

— Зубков Андрей и Пройдаков Алексей, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Морозова Елена, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Овчинников Виктор, Ростовская обл., г. Новочеркасск, Суворовское училище МВД РФ, преподаватель **Воронкова О.Б.**;

— Потапенко Андрей и Усков Лев, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Тимофеев Иван, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**, а головоломки “сан-го-ку”:

— Абдувахидова Алина, Абдувахидова Софья, Базанов Илья, Медведева Анастасия, Милушкин Дмитрий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Авдеенко Дмитрий, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Ануфриева Дарья, Голубева Илона, Мусатов Тимофей, Плотникова Валерия, Сатыбалдина Ксения и Цепилова Мария, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**;

— Власенко Ольга, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Докучаева Валерия, Малых Кирилл, Матвеева Диана и Сидоренкова Анастасия, Куминская средняя школа, Тюменская область, Ханты-Мансийский автономный округ — Югра, Кондинский р-н, учитель **Шишигина О.В.**;

— Крюкова Арина, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Лагоша Артем, Республика Коми, г. Усинск, Центр дополнительного образования детей, “Школа программистов”, педагог **Казанцева О.В.**;

— Ладнова Наталья и Костина Евгения, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Морозова Ирина и Шаламова Алевтина, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Орлова Анна, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Тимофеев Иван, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

— Трофимов Сергей, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

Задание “Три вопроса”

Напомним заданные вопросы:

- 1) “Как, не производя никаких арифметических действий, найти остаток от деления любого двоичного числа на 2?”;

2) “Как проще всего умножить любое двоичное число на 3?”;

3) “Как проще всего умножить любое шестнадцатеричное число на десятичное число 24?”.

Ответы

1. Остаток от деления любого двоичного числа на 2 равен его последней цифре.

2. Чтобы умножить любое двоичное число на 3, достаточно сначала умножить его на 2 (приписав справа ноль — см. выше), а затем прибавить к полученному числу исходное, например: $110100101_2 \cdot 3_{10} = 1101001010_2 + 110100101_2 = 10011101111_2$.

3. Чтобы умножить любое шестнадцатеричное число на 24_{10} , достаточно найти сумму произведений этого числа на 16_{10} и на 8_{10} . Первое произведение (на 16_{10}) можно получить, приписав к заданному числу справа ноль. Второе произведение равно половине первого.

Можно также сложить данное число с его половиной и к полученной сумме приписать 0. Однако при этом возникает проблема расчета половины заданного числа. Конечно, когда делимое число четное, найти частное от его деления на 2 легко. А когда оно заканчивается на нечетную цифру? Умножить же заданное число на 8 нерационально (для этого надо помнить таблицу умножения в шестнадцатеричной системе). Рациональный прием для определения произведения шестнадцатеричного числа A на 8 состоит в следующем. Для вычисления $A_{16} \cdot 8$ надо перевести числа A и 8 в двоичную систему счисления (превратив каждую шестнадцатеричную цифру в тетраду¹ двоичных). Так как $8_{16} = 1000_2$, то для нахождения произведения дописываем в записи числа A справа три нуля и с помощью тетрад возвращаемся в шестнадцатеричную систему счисления. Например:

$$\begin{aligned} 137_{16} \cdot 24_{10} &= 137_{16} \cdot 18_{16} = 1370_{16} + 137_{16} \cdot 8_{16} = \\ &= 1370_{16} + 100110111_2 \cdot 1000_2 = \\ &= 1370_{16} + 1001\ 1011\ 1000_2 = 1370_{16} + 9B8_{16}. \end{aligned}$$

прислали

— Бородай Петр, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Волчек Елена, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Зубов Владислав, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Потапенко Андрей, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Товмасын Лиана и Широков Дмитрий, средняя школа села Урман, Республика Башкортостан, Иглинский р-н, учитель **Товмасын М.Г.**;

— Торопов Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Шаламов Алексей, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**

Ответы на задания для самостоятельной работы, предложенные в статье “Через тернии к «звездам»”:

¹ Группу из четырех цифр.

— Бердыева Алия, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Лежнева Александра, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Новосельцева Александра, Республика Карелия, поселок Надвоицы, школа № 1, учитель **Каликина Т.В.**;

— Омельченко Степан, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**;

— Парамонов Павел, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Сорокина Мария, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Чухонцев Павел, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Шальнева Ирина, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

Вторую часть числовых ребусов, записанных в четверичной системе счисления, правильно решили:

— Абакумов Станислав, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Алешкова Яна, средняя школа поселка Ерофей Павлович, Амурская обл., Сковородинский р-н, учитель **Красненкова Л.А.**;

— Бородай Петр, средняя школа поселка Осиновка, Алтайский край, учитель **Евдокимова А.И.**;

— Дегтярь Анатолий, средняя школа поселка Новопетровский Московской обл., учитель **Артамонова В.В.**;

— Стороженко Степан, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Торопов Александр, средняя школа деревни Муравьево, Вологодская обл., учитель **Муравьева О.В.**;

— Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Шугай Илларион, средняя школа села Сердар, Республика Марий Эл, учитель **Чернова Л.И.**;

— Яковлев Антон, средняя школа села Восточное Нижегородской обл., учитель **Долгова Г.А.**

Марат Хозин также правильно решил числовой ребус со словом-числом **МАССИВ**.

Списки читателей, приславших ответы на задания, опубликованные в январском выпуске “В мир информатики”*

Задание “Три вопроса” (рубрика “Поиск информации”)

Ответы

— Абдувахидова Алина, Абдувахидова Софья, Медведева Анастасия, Микаев Никита и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учителя **Волков Ю.П.** и **Волкова Т.П.**;

* Настоящий выпуск готовился в середине февраля, поэтому в списке учтены ответы, полученные к этому времени.

— Аноп Марина, Арак Владислава, Богданова Алина, Булгак Мария, Буричева Дарья, Ветрова Галина, Гудаева Эвелина, Даминава Анастасия, Джуссоева Яна, Дмитриев Роман, Доронина Полина, Железко Дарина, Колпин Василий, Коханская Анна, Красильников Александр, Кульгова Елизавета, Малкова Ольга, Омельченко Мария, Пашков Глеб, Переверзева Ксения, Птушко Анастасия, Свирид Мария, Семерикова Юлия, Смирнов Дмитрий, Соколов Павел, Фадеев Егор, Федосенко Алена, Шилова Наталья, Республика Карелия, г. Сегежа, школа № 6, учитель **Гагарина Э.В.** (из этой же школы получены также ответы, подписанные “Лиза” и “Ierochka”);

— Барановская Татьяна и Жукова Ирина, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Блюденев Кирилл, Кокорева Виктория и Раджабова Алина, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Деревянченко Дарья, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Коробова Анна, Мадеевская основная школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**;

— Лежнева Александра, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Луцук Максим, Назаркина Татьяна и Шикалович Ростислав, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Мироненко Елизавета, средняя школа села Дохновичи Брянской обл., учитель **Клопова Е.В.**

Задача “Странный мальчик”

— Богданова Алина, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Гулевских Анна и Обухова Элина, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Деревянченко Дарья, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Зубов Владислав, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Крысанов Виктор, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Миноцкий Ян, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Трус Анастасия, средняя школа села Дохновичи Брянской обл., учитель **Клопова Е.В.**

Кроссворд

— Абдувахидова Алина, Абдувахидова Софья и Микаев Никита, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Баженов Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Байбуза Дарья, Горелкина Лариса и Кузнецов Семен, средняя школа села Ириновка, Новобураский р-н Саратовской обл., учитель **Брунов А.С.**;

— Баркалова Екатерина, Попов Тимур и Сорокина Диана, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Трус Анастасия, средняя школа села Дохновичи Брянской обл., учитель **Клопова Е.В.**;

— Шаповалова Анастасия, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

Задача “Школьные учителя”

— Абдувахидова Алина, Абдувахидова Софья, Добрыднева Ксения, Матренина Екатерина, Медведева Анастасия, Микаев Никита, Миноцкий Ян и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учителя **Волков Ю.П.** и **Волкова Т.П.**;

— Богданова Алина, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Гулевских Анна и Обухова Элина, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Деревянченко Дарья, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Крысанов Виктор, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Кулюдо Ксения, Мадеевская основная школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**

Задача “Блондины и брюнеты в классе”

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Гулевских Анна, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Зубов Владислав, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

Задача “Ночной переход по мосту”

— Баженов Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Гулевских Анна и Попов Тимур, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Зубов Владислав, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**

Задача “Подарки Деда Мороза”

— Баженов Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Гулевских Анна и Сорокина Диана, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Данилова Анна, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**

Задача “Семья”

— Ашеко Валерия, Мадеевская основная школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**;

— Богданова Алина и Макаревич Елена, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Бородюк Анна и Василенко Татьяна, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Гулевских Анна и Обухова Элина, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Медведева Анастасия, Владимирская обл., г. Струнино, школа № 11, учитель **Волкова Т.П.**

Задача “Странное изображение”

— Абовян Гаянэ, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Гулевских Анна, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Филиппенко Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

Задача “Лилии в пруду”

— Гулевских Анна и Обухова Элина, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Данилова Анна, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Добрыднева Ксения, Матренина Екатерина и Микаев Никита, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Крысанов Виктор, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

Задача “Четыре сосуда с жидкостями”

— Абдувахидова Алина, Абдувахидова Софья, Добрыднева Ксения, Медведева Анастасия, Микаев Никита, Миноцкий Ян и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учителя **Волков Ю.П.** и **Волкова Т.П.**;

— Абовян Гаянэ, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Григорьев Николай, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Гулевских Анна, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Цыганков Евгений, Вадьковская средняя школа, Брянская обл., Погарский р-н (информатику изучает с мамой, **Цыганковой И.Ю.**, учителем другой школы).

Евгений Цыганков также правильно решил задачу “Улов”, опубликованную в февральском выпуске журнала.

Задача “Покупка лошади”

— Абдувахидова Алина, Абдувахидова Софья, Добрыднева Ксения и Матренина Екатерина, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**;

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Гулевских Анна и Попов Тимур, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Данилова Анна, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Деревянченко Дарья, г. Ярославль, школа № 33, учитель **Ярцева О.В.**

Задача “Друзья и города”

— Богданова Алина, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Гулевских Анна и Сорокина Диана, г. Воронеж, лицей № 2, учитель **Комбарова С.И.**;

— Филиппенко Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**

Судоку

— Байбуза Дарья, Горелкина Лариса, Кузнецов Семен и Михайлова Алена, средняя школа села Ириновка, Новобураский р-н Саратовской обл., учитель **Брунов А.С.**;

— Деревянченко Дарья, г. Ярославль, школа № 33, учитель **Ярцева О.В.**;

— Диков Андрей, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Дубовиков Тимофей, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Селиверстова Наталья, Челябинская обл., г. Златоуст, школа № 9, учитель **Мусатова И.Б.**

Ребусы в четверичной системе счисления

— Гируцкий Павел, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Мироненко Елизавета, средняя школа села Дохновичи Брянской обл., учитель **Клопова Е.В.**;

— Новиков Филипп и Цыплаков Евгений, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.**

Числовой ребус “ИКС на ИКС”

— Багрова Анастасия, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**;

— Баженов Михаил, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Ещеркина Виктория, г. Челябинск, школа № 124, учитель **Юртаева Г.Ю.**;

— Цыплаков Евгений, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

Задания, предложенные в рамках конкурса № 114, выполнили:

— Абдураев Вадим и Воронцов Кирилл, средняя школа села Ириновка, Новобураский р-н Саратовской обл., учитель **Брунов А.С.**;

— Якушев Константин, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

Вадим, Кирилл и Константин будут награждены дипломами. Поздравляем!

Переложить одну спичку

— Абдувахидова Алина, Абдувахидова Софья, Добрыднева Ксения, Матренина Екатерина, Медведева Анастасия, Милушкин Дмитрий и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учителя **Волков Ю.П.** и **Волкова Т.П.**;

— Назаркина Татьяна, средняя школа г. Пионерский Калининградской обл., учитель **Багрова О.А.**

Судоку, опубликованные в февральском выпуске, решили:

— Диков Андрей, г. Пенза, школа № 512, учитель **Гаврилова М.И.**;

— Дубовиков Тимофей, средняя школа села Горелово Тамбовской обл., учитель **Шитова Л.А.**;

— Кулюдо Ксения, Мадеевская основная школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**

Ответы на задание “Четыре вопроса” (февральский выпуск, рубрика “Поиск информации”) представили:

— Коробова Анна, Мадеевская основная школа, Брянская обл., Погарский р-н, учитель **Цыганкова И.Ю.**;

— Лежнева Александра, г. Пенза, школа № 512, учитель **Гаврилова М.И.**

Задания конкурса № 115 выполнили и будут награждены дипломами Абдувахидова Алина, Абдувахидова Софья, Милушкин Дмитрий, Миноцкий Ян, Рыжов Антон и Хозин Марат, Владимирская обл., г. Струнино, школа № 11, учитель **Волков Ю.П.** Поздравляем!

Ряд заданий для самостоятельной работы выполнили ученики средней школы г. Пионерский Калининградской обл., учитель **Багрова О.А.**:

— предложенные в статье “Трилистник и другие”: Дощик Константин, Коляков Дмитрий, Лебедева Екатерина, Назаркина Татьяна, Сивуда Максим, Султаншин Ансар, Сучок Алина, Чигарев Дмитрий, Чунин Павел и Шашкин Данил;

— предложенные в статье “Еще один «растительный» график”: Ананьев Александр, Коляков Дмитрий, Лебедева Екатерина, Назаркина Татьяна, Сучок Алина и Шашкин Данил;

— связанные с подсчетом так называемых “счастливых” билетов в среде программы Microsoft Excel: Султаншин Ансар, Сучок Алина, Чунин Павел и Шашкин Данил.

Спасибо всем приславшим ответы!

Ответы на задания для самостоятельной работы учащихся в статье Д.М. Златопольского “О принципах делимости” (“Информатика” № 12/2015)

1.

1) число $(a_n a_{n-1} \dots a_1 a_0)_2$;

— делится на 2, если его последняя цифра a_0 равна нулю;

— делится на 4, если обе его последние цифры a_1 и a_0 равны нулю;

2) число $(a_n a_{n-1} \dots a_1 a_0)_3$;

— делится на 2, если сумма его цифр есть четное число;

— делится на 3, если его последняя цифра a_0 равна нулю;

3) число $(a_n a_{n-1} \dots a_1 a_0)_4$;

— делится на 2, если на 2 делится его последняя цифра a_0 ;

— делится на 3, если сумма его цифр кратна трем;

— делится на 4, если его последняя цифра a_0 равна нулю;

4) число $(a_n a_{n-1} \dots a_1 a_0)_6$;

— делится на 2, если на 2 делится его последняя цифра a_0 ;

— делится на 3, если на 3 делится его последняя цифра a_0 ;

— делится на 5, если сумма его цифр кратна пяти;

— делится на 6, если его последняя цифра a_0 равна нулю;

5) число $(a_n a_{n-1} \dots a_1 a_0)_8$;

— делится на 2, если на 2 делится его последняя цифра a_0 ;

— делится на 4, если на 3 делится его последняя цифра a_0 ;

— делится на 7, если сумма его цифр кратна семи;

— делится на 8, если его последняя цифра a_0 равна нулю;

6) число $(a_n a_{n-1} \dots a_1 a_0)_{12}$;

— делится на 8, если на 8 делится число $(a_1 a_0)_{12}$, образованное его двумя последними цифрами;

— делится на 9, если на 9 делится число $(a_1 a_0)_{12}$, образованное его двумя последними цифрами;

— делится на 11, если на 11 делится сумма его цифр, т.е. $a_n + a_{n-1} + a_{n-2} + \dots + a_1 + a_0$;

7) q -ичное число делится на $(q - 1)$, если сумма его цифр кратна $(q - 1)$.

2. Число, кратное семи в десятичной системе, будет таким же в любой другой системе счисления.

3. Разность $A - B$ кратна числу 9 (убедитесь в этом самостоятельно!). Поэтому ответ — 9.

4. Аналогично десятичной системе счисления, в q -ичной системе разность $A - B$ кратна $q - 1$, так как сумма ее цифр кратна $q - 1$ (см. чуть выше). Согласно условию, сумма цифр разности равна 2013. Разложим число 2013 на простые множители:

$$2013 = 3 \cdot 11 \cdot 61.$$

Так как в условии фигурирует цифра 3, то возможные значения оснований систем счисления $q \geq 4$. Следовательно, $q = 12$ или $q = 62$.

Ответ: 74.

Екатерина Логвиновна Ющенко

Когда говорят о связи двух понятий — “программирование” и “женщины”, то чаще всего имеют в виду двух женщин — Аду Лавлейс [1–2] и Грейс Хоппер [3].

Но есть еще одна женщина, чей вклад в теорию и практику программирования не менее значителен. Это Екатерина Логвиновна Ющенко (девичья фамилия Рвачева).

Е.Л. Ющенко (8 декабря 1919 – 15 августа 2001) — кибернетик, основатель первой советской школы теоретического программирования, автор одного из первых в мире языков программирования высокого уровня.

Екатерина Логвиновна родилась на Украине, в бывшей столице запорожских казаков городе Чигирине, в семье учителя истории и географии Логвина Федоровича Рвачева, выходца из Подмоковья.

Училась на первом курсе физико-математического факультета Киевского университета имени Т.Г. Шевченко. Но в 1937 году, из-за того, что ее отца посчитали украинским националистом (хотя по-украински он говорил с явным акцентом, сразу выдававшим его российское происхождение) и арестовали², Катю из университета исключили.

Е.Л. Ющенко впоследствии рассказывала [4]: “Летом 1938 года я подала документы в Московский университет. Мне сообщили, что меня приняли, но без предоставления общежития. Это меня не устраивало. От кого-то узнала, что университет в Воронеже никому не отказывает в жилье. Приехала туда в сентябре, уже шли занятия. Ознакомившись с моими документами, в которых было сказано, что родители арестованы, мне сказали, что могут принять и поселить в общежитии, но стипендию не назначат.

Случайно, выходя из института, увидела на доске объявлений телеграмму в адрес Воронежского университета, присланную из Самарканда. В ней сообщалось, что Узбекский государственный университет объявляет набор студентов, причем обеспечивает их общежитием, стипендией и оплачивает дорогу.

После всех мытарств и унижений при попытках продолжить учебу — а другого я просто не мыслила — это показалось возможным спасением.

К счастью, так и получилось. Я вся отдалась учебе. Не пропускала ни одной лекции, читала и перечитывала учебники, перерешала все задачи, имевшиеся в учебниках и задачниках. И так было все три года. Возможность завершить наконец свое образование окрылила меня и помогла пережить нагрянувшую беду. Но я всегда помнила своих родителей и так далекую от Самарканда Украину. Иног-

да мне было особенно тяжело, и тогда я пыталась выразить свои чувства стихами”.

Вот эти проникновенно-пронзительные строчки, написанные в мае 1939 года в Самарканде, когда Катя заканчивала первый курс университета, проведя целый год вдали от родины (приводятся на языке оригинала):

Журба по Україні (Тоска по Украине)

*Я знаю, що таке журба по рідній, милій Україні
Я знаю, як рида душа в далекім краї, на чужині.
Я знаю, тяжко зустрічатъ пору весняну за горами,
Зітхатъ та мріять, як дзюрчатъ струмки весняні
під ногами.
Зима змішалась з літнім днем, цвіте урюк, дощ
не змовкає,
І на сирий, слизький дивал, вітер плюсточки
збиває...
Не чути запахів квіток і соловейка в темнім гаю,
Не чути гуркиту річок, як то бувало в ріднім краю...
О краю рідний, щирій мій, ти, краю вірний, серцю
милий,
Нащо ж шукаю я собі другого місця для могили?
Ні, не залишила тебе, я тут — душа моя з тобою,
І раз вернувшись назавжди вже не розстанусь я з
тобою...³*

*Нет, не забыла я тебя, моя душа — с тобой.
И однажды вернувшись к тебе, — уже не расстанусь...*

“Весной 1941 года я досрочно сдала экзамены за третий курс, — продолжала рассказ Екатерина Логвиновна, — и поехала к матери, которая жила в Овруче. По дороге я рассчитывала заехать к брату, что жил под Киевом в селе Бородянка.

На Киевский вокзал поезд пришел поздно ночью 21 июня. Пригородным, ровно в полночь я добралась до Бородянки. Полустанок был безлюден. К селу вела грунтовая дорога. Подождав немного и никого не увидев, я пошла по ней, едва таща тяжелый чемодан с подарками — фруктами, орехами, книгами. Через километр дорога раздвоилась. По какой идти? Решила ждать до утра. Оттащила чемоданы на бугорок с травой, присела сама и... уснула. Разбудил сильный гул над головой. Прямо надо мной один за другим пролетели семь необычного вида самолетов с крестами на крыльях. «Санитарные, что ли?» — подумала я. Когда вдалеке раздались глухие взрывы, удивилась: «Вот работяги — и в воскресенье работают!». Наступал рассвет. Появился прохожий. Дом брата оказался недалеко. Мне были очень рады, угостили молоком, порасспрашивали и все легли досыпать. Проснулись поздно. Брат вышел на улицу и быстро вернулся:

— Говорят, Германия объявила нам войну и утром бомбили Киев!

³ Последние две строки стихотворения (перевод Д.М. Златопольского):
*Нет, не забыла я тебя, моя душа — с тобой.
И однажды вернувшись к тебе, — уже не расстанусь...*

² В 50-х годах Л.В. Рвачев и его жена были реабилитированы.

Так я встретила первый день войны... Я вернулась в Самарканд. Оказалось, что университет объединили со Среднеазиатским государственным университетом в Ташкенте, и мне пришлось поехать туда. Появилась в университете как раз к началу занятий. Одновременно устроилась на работу на военном предприятии, выпускающем прицелы для танковых пушек. Ко мне приехали мать, сестра и брат.

Через год, когда я закончила университет, встала проблема — как жить дальше. Войне не видно конца, а назначения на работу я не получила.

Еще когда я училась в университете, студентам был зачитан призыв ЦК ВЛКСМ прийти на помощь шахтерам Ангренского угольного комбината, чтобы увеличить добычу угля, так необходимого стране, поскольку Донбасс был оккупирован фашистами.

Город Ангрэн был в 150 км от Ташкента. Я решила, что в такое трудное для страны время мне надо быть там, вместе со всеми, кто откликнулся на призыв. Мне определили профессию — запальщик-взрывник. Обычно на нее подбирали опытных шахтеров, проработавших полтора-два года. А тут — сразу...

В 1943 году на комбинат пришел приказ — всех, имеющих законченное педагогическое образование, вернуть в школы! Приказы тогда неукоснительно соблюдались. Нас собрали, поблагодарили за самоотверженную работу. Я вернулась в Ташкент и 1943–1944 учебный год работала преподавателем математики и физики в средней школе.

Сестра получила вызов из родного села, уже освобожденного от оккупантов. Всей семьей мы вернулись в Украину, я устроилась учительницей в этом селе и вела уроки математики, физики, черчения и даже украинской литературы. В следующем году по приглашению старшего брата, учителя физики, переехала в г. Стрый и 1945–1946 учебный год работала в городской средней школе.

Узнав, что во Львове открылся филиал Института математики АН Украины, я поехала туда, и — тут мне крупно повезло — первым, к кому я обратилась, оказался Борис Владимирович Гнеденко, будущий академик АН Украины. Увидев в моем дипломе одни пятерки, он тут же предложил работу в его отделе теории вероятностей...

Я занималась специальными вопросами теории вероятностей. Мною были получены оригинальные результаты по многомерным устойчивым законам распределения, важным для развития квантовой механики. Я подготовила и в 1950 году успешно защитила кандидатскую диссертацию. Мой первый оппонент, академик Андрей Николаевич Колмогоров, дал — мне приятно сказать это — высокую оценку моей работы.



В 1950 году Гнеденко избрали действительным членом АН Украины, его отдел перевели в Киевский институт математики АН Украины. Я к этому времени уже имела вполне приличную квартиру. Когда Борис Владимирович предложил мне переехать в Киев, я сразу же согласилась, не спросив, где придется жить, хотя уже ждала ребенка. Наверно, это было опрометчиво с моей стороны, и я почувствовала это сразу. Мы поселились в маленькой комнатухе. Для новорожденного сына не хватало ни света, ни тепла, зато было вдоль сырости, и это сказалось на его здоровье.

Институт математики был в эти годы на взлете. Его директор, академик Александр Юрьевич Ишлинский, занимался теорией гироскопов — очень актуальной проблемой на то “предспутниковое” время. Ему надо было производить сложные расчеты. Для этой цели был закуплен комплект счетно-аналитических машин и установлен в подвале здания президиума АН Украины. Меня назначили руководителем вновь созданной вычислительной лаборатории, пообещав позаботиться о лучшем жилье.

Когда Сергея Алексеевича Лебедева перевели в Москву, а его лабораторию — в Институт математики, директором которого в 1952 году стал Гнеденко, он прикомандировал меня, моего мужа Алексея Андроновича Ющенко, Юрия Владимировича Благовещенского и Владимира Семеновича Королюка к этой лаборатории, расположенной под Киевом в одном из зданий бывшего монастыря в Феофании.

А.Ю. Ишлинский выполнил свое обещание. Сначала меня переместили в большую комнату дома на Крещатике, а затем предоставили финский домик в Феофании. Так я оказалась в историческом месте, где под руководством Лебедева в 1948–1951 годах была создана первая в континентальной части Европы ЭВМ МЭСМ⁴.

В должности старшего научного сотрудника Института математики АН УССР Е.Л. Ющенко работала до 1957 года, после чего стала заведующей отделом Института кибернетики АН УССР.

С 1954 года она работала в области вычислительной математики, разрабатывала алгоритмы для МЭСМ, в том числе алгоритмы решения задач внешней баллистики для ракетно-космических комплексов, разработки планов производства и т.п.

Екатерина Ющенко становится первым в СССР доктором физико-математических наук, которому эта степень была присвоена за работы по программированию. Она предложила один из первых в мировой практике языков программирования — “Адресный язык”, конструкции которого вошли в состав современных языков.

⁴ Малая электронная вычислительная машина.

Екатерина Логвиновна была в числе авторов (с Б.В. Гнеденко и В.С. Королюком) первого учебника по программированию в СССР — книги “Элементы программирования”, изданной в 1961 году, переизданной в 1964 году, а также в переводе: в 1964 году в ГДР и Венгрии, а в 1969 году во Франции.

Под ее руководством и с непосредственным участием была создана серия инструментально-технологических комплексов. За работы в этой отрасли ей были присуждены две Государственные премии Украины и премия Совета Министров СССР, а за теоретические разработки в области компьютерной алгебры — академическая премия имени Глушкова. Также Е.Л. Ющенко награждена Орденом княгини Ольги. Была членом-корреспондентом Национальной академии наук Украины, почетным членом Академии наук Молдавии.

Е.Л. Ющенко получила пять авторских свидетельств, ею разработано восемь Государственных стандартов Украины. Она является автором более двухсот научных работ, в том числе 23 монографий и учебных пособий, часть которых была переведена и издана в Германии, Чехии, Венгрии, Франции и Дании.

Ею подготовлено 45 кандидатов и 11 докторов наук.

Была членом редколлегии журналов “Кибернетика и системный анализ” и “Проблемы программирования”, членом Научного совета по проблеме “Кибернетика” Национальной академии наук Украины, членом квалификационного совета по присуждению ученых степеней.

В апреле 1943 г. тогда шахтер-запальщик Катя Рвачева, находясь в далеком-далеком от Украины городе Ангрене, где она добывала уголь, так необходимый стране для победы над фашистскими захватчиками, словами стихотворения “Зори” выразила свою заветную мечту:

*...Вечер огни разослал по долине,
Чудится: звезды в Ангрене в плену.
Хочется мне поскорее к вершине,
К звездам поближе... Сама не пойму.*

Ее мечта исполнилась...

Литература

1. Ада Лавлейс. / “Информатика”, 2008, № 5.
2. Ада Байрон. / “Информатика”, 2012, № 6.
3. Грейс Хоппер. <https://ru.wikipedia.org>.
4. Малиновский Б.М. История вычислительной техники в лицах. К.: Фирма “КИТ”, ПТОО “А.С.К.”, 1995.
5. http://www.icfcst.kiev.ua/MUSEUM/Ushchenko-memoirs_r.html.

ЭТО ПОЛЕЗНО ЗНАТЬ

Осторожно — “фишинг”

Многие читатели, конечно, слышали о так называемом “фишинге” — одном из видов интернет-мошенничества. О том, что его название происходит от английского слова *fishing*, означающем “рыбная ловля”, “выуживание”, известно. Но менее известно, что по-английски интернет-фишинг пишется как *phishing*, то есть не совсем так, как “рыбная ловля” (хотя звучит так же — “фишинг”). Почему? Первая буква *p* связана со словом *password* — *пароль*, потому что целью мошенников является получение идентификационных данных пользователей (паролей и логинов). Получается что-то типа “выуживание паролей”, что соответствует действительности.

Организаторы фишинг-атак часто используют массовые рассылки электронных писем от имени популярных фирм, сервисов и др. Рассчитывая на невнимательность или неинформированность пользователей, в эти письма они вставляют ссылки на фальшивые сайты, являющиеся почти точной копией настоящих. Оказавшись на таком сайте, пользователь может сообщить преступникам ценную информацию, например, данные, позволяющие управлять его банковским счетом из Интернета (имя пользователя и пароль для доступа, номер своей кредитной карты). В Интернете был описан случай получения письма клиентом системы Яндекс. Деньги о том, что счет его забло-

кирован. Для повторной активации счета пользователю рекомендовали обратиться по адресу <http://passport.yandex.ru/passport/?mod...> За исключением имени домена *yandex.ru*, который из-за невнимательности легко можно принять и за *yandex.ru*, все реквизиты известной фирмы указаны верно. А для обращения с запросом по поводу восстановления счета достаточно было только щелкнуть мышью на этом адресе. В результате пользователь, передав якобы в фирму “Яндекс” все свои реквизиты, передавал их на самом деле мошенникам и, тем самым, давал им возможность манипулировать со своим счетом в банке.

Другой пример. Среди молодых людей популярна социальная сеть ВКонтакте. А на любом ресурсе может быть ссылка, в которой говорится: “Добавь меня в друзья и получи за это 10 голосов в ВКонтакте”. Согласитесь, заманчиво? Вы переходите по этой ссылке, попадаете на сайт, вроде бы ничем не отличающийся от сайта ВКонтакте, вводите свой логин и пароль. Попадаете к себе на страничку — голосов не прибавилось. Долго думаете, но в итоге забываете о дополнительных голосах. Но данные вашей учетной записи уже у совершенно посторонних людей... Все дело в том, что, увидев страницу авторизации, вы не обратили внимание на адрес сайта. Вместо *vk.com* там было указано *vk.org*.

Или. Ваша девушка или парень дает вам ссылку на сайт с предложением “Лайкни Кота”.

Вы “без задней мысли” переходите по ссылке, также вводите логин и пароль, и... ваша “половинка” теперь знает ваши учетные данные и сможет посмотреть, с кем это вы там переписываетесь.

Так что если вас где-то просят ввести пароль — проверяйте внимательно доменный адрес. И вообще, перефразируя известное выражение, “Люди, в Интернете будьте бдительны!”.

Подготовил В.П. Константинов,
Москва

МИР ИНТЕРНЕТА

Википедия — 15 лет, 15 фактов

В январе 2016 года исполнилось 15 лет крупнейшей и влиятельнейшей интернет-энциклопедии — Википедии (англ. Wikipedia).

Название энциклопедии образовано от английских слов *wiki* (вики, технология, лежащая в основе функционирования сайта; в свою очередь, слово заимствовано из гавайского языка, в котором оно имеет значение “быстро”) и *encyclopedia* (энциклопедия).

Главной особенностью энциклопедии Википедия (благодаря технологии *wiki*) является то, что создавать и редактировать ее статьи может любой соблюдающий правила Википедии пользователь Интернета, причем в абсолютном большинстве случаев даже без регистрации на сайте энциклопедии. Поэтому ее часто называют “народной энциклопедией”. Все вносимые такими добровольцами в какую-либо статью этой энциклопедии изменения незамедлительно становятся видными всем посетителям сайта. В декабре 2013 года в заявлении ЮНЕСКО по случаю награждения Джимми Уэйлса, основателя Википедии, Золотой медалью Нильса Бора про Википедию было сказано, что она является “символом эпохи взаимодействия, в которую мы живем, и это не просто инструмент, это воплощение мечты, столь же древней, как человеческий интеллект и собрания Александрийской библиотеки”.

Ниже приводятся 15 интересных фактов о Википедии, причем далеко не все эти факты взяты из самой энциклопедии⁵.

1. Википедию создали философ Ларри Сэнглер и финансист Джимми Уэльс из США, которые, по собственному признанию, строили свой проект на утопической идее свободного доступа каждого ко всей сумме знаний человечества и жертвованиях со всего мира.

Можно сказать, они преуспели в своих начинаниях: к январю 2016 года на сайте энциклопедии написано более 38 млн статей, более 4 млн пользователей вносили информацию, которую правили более 2 млрд раз; этими правками, среди прочего, занимались 4 тысячи человек, имеющих звание администратора.

2. Десять языков, на которых написано больше всего статей, — английский (более 5 млн), шведский (более 2,5 млн), немецкий (1,9 млн), нидер-

ландский (1,85 млн), себуано (1,85 млн), французский (1,7 млн), русский (1,28 млн), варайский (1,25 млн), итальянский (1,25 млн) и испанский (1,23 млн).

При этом количество правок на упомянутых филиппинских языках — себуано и варайском — существенно меньше, чем у других лидеров списка, что косвенно говорит о недостаточной проработанности и завершенности многих из созданных статей.

3. Ни одной статьи, кроме главной страницы Вики, нет в разделах на языках гереро (используется на территории ЮАР) и канури (на котором говорят в Нигере и Нигерии).

4. У языков, использующих больше одной системы письма — например, сербского (кириллица и латиница) или уйгурского (кириллица, латиница и арабская вязь), — есть возможность дублировать статьи в соответствующих алфавитах; на практике, однако, отнюдь не все статьи на подобном языке будут иметь свои симметричные дубликаты во всех используемых письменностях.

5. Как минимум три языка, пользующиеся одним алфавитом, имеют по две версии Википедии: речь об английском (стандартная и “упрощенная” версия), норвежском (букмол и нюнорск) и белорусском (официальная, “школьная”, грамматика и так называемая “тарашкевица”). В ряде случаев описание одних и тех же объектов и людей может радикально отличаться.

6. Из искусственных языков Википедия существует на эсперанто (более 200 тыс. статей), вояпюке, идо и интерлингве. При этом до использования языка эльфов из произведений Джона Толкиена Википедия, в отличие от сервиса “Яндекс.Переводчик”, еще не дошла.

7. В 2006–2007 годах существовала Википедия на “сибирском языке”, или говоре, — выдуманном языке, при этом легко понимаемом носителями русского языка. Из-за обилия нецензурной лексики и искажения фактов эта “энциклопедия” была удалена в ноябре 2007 года, накопив более 6 тыс. статей, большинство которых были пустыми.

8. Возросший с годами авторитет энциклопедии сыграл с ней дурную шутку. В 2009 году в английской версии статьи о французском композиторе, лауреате премии “Оскар” Морисе Жарре сделал правку студент-ирландец, сообщив о смерти творца. Креативный юноша очень убедительно описал кончину Жарра и выдумал последние слова “успешного”, чем спровоцировал шквал сообщений в СМИ, включая уважаемые издания. Когда

⁵ Статья подготовлена по материалам сайта <http://www.bbc.com>.

обман раскрылся, всем участникам скандала пришлось опровергать свои сообщения.

9. Как утверждает радиоведущая Рэйчел Мэрсден, основатель Википедии Джимми Уэльс в 2008 году лично отредактировал ее биографическую страницу, чтобы написать, что они расстались, причем она не знала о его решении прекратить отношения до того, как прочитала статью. Уэльс отрицал, что редактировал статью, однако другие редакторы нашли его имя в числе тех, кто правил информацию. В отместку Мэрсден продала свитер и футболку, принадлежавшие, как утверждала девушка, Уэльсу, с молотка.

10. Когда международная группа социологов проанализировала 10 версий сетевой энциклопедии на разных языках, то выяснила, что самыми спорными — подвергающимися наибольшему числу правок — статьями являются описания политиков (бывшего президента США Джорджа Буша-младшего — на английском и французского политика-женщины Сеголен Руаяль — на французском), религиозные фигуры (Иисус и пророк Мухаммед — в ряде языков).

11. Упомянутая статья о Джордже Буше-младшем на английском языке является абсолютным лидером по числу правок за все годы существования интернет-энциклопедии: ее текст менялся 45 862 раза. В российском сегменте, помимо ежегодных списков умерших, лидерами по числу правок стали статьи о Сталине и мультсериале “Смешарики”, а также о России и Украине. На каждую из них пришлось более 7 тыс. правок.

12. В ряде стран те или иные страницы онлайн-энциклопедии подвергались блокировке. Особенно непросто

складывается судьба Википедии в КНР: китайская версия сайта была заблокирована в Китае в июне 2004 года, в октябре–ноябре 2006 года. В июне 2015 года СМИ сообщили о полной блокировке китайской Википедии в КНР. Кроме того, Пекин не раз частично блокировал энциклопедию и на английском языке. Китайская версия энциклопедии содержит более 860 тысяч статей.

13. В августе прошлого года Роскомнадзор (Федеральная служба по надзору в сфере связи, информационных технологий и массовых коммуникаций) предъявил претензии к российской Википедии относительно статьи, связанной с наркотиками. Спустя какое-то время организация передумала блокировать энциклопедию, сославшись на то, что информация в “проблемной” статье отредактирована.

14. Энциклопедия старается быть самокритичной в своих статьях: в англоязычном разделе существуют статьи “Гендерные предрассудки Википедии” и “Список скандалов, связанных с Википедией”. Что до первой, то женщины регулярно жалуются на доминирование мужчин в коллективе редакторов и вытекающий из этого дисбаланс в освещении ряда тем. Джимми Уэльс признал, что предыдущие попытки сбалансировать коллектив по половому признаку провалились, и выразил намерение исправить такое положение дел в будущем.

15. В честь 15-летия энциклопедия запустила проект “Wikipedia 15”, где призывает пользователей рассказать, что она для них значит. Помимо многословных изречений о пользе для всего человечества, среди ответа на этот вопрос попадаются и “успешно сданные экзамены”...



ДЛЯ ЭРУДИТОВ

Новое наибольшее простое число

Американский профессор Кертис Купер из Центрального университета города Миссури получил новое наибольшее известное науке простое число. Оно равно $2^{74207281} - 1$ и содержит 22 338 618 цифр.

Открытие нового числа состоялось благодаря проекту GIMPS (*Great Internet Mersenne Prime Search*), использующему компьютеры пользователей Интернета.

Алгоритм обнаружения подобных чисел базируется на их поиске в форме так называемых “чисел Мерсенна”, которые имеют вид $2^p - 1$, где p также является простым числом. С помощью этого алгоритма и были найдены 15 последних самых больших простых чисел.

Поиск больших простых чисел имеет и практическое значение. Например, они используют-

ся в криптографии (чем больше простое число, которое используется для шифрования данных, тем сложнее этот код взломать). Расчеты чисел Мерсенна используются в тестах производительности компьютерного оборудования. Не так давно GIMPS помог обнаружить ошибку в процессорах Intel Skylake, работающих при высокой нагрузке.

Примечательно, что последний раз наибольшее простое число также открыл Купер в 2013 году — оно оказалось равным $2^{57885161} - 1$ и содержало больше 17 миллионов цифр. Тогда за это открытие математик (точнее — его группа) получил три тысячи долларов.

На данный момент наука знает о 49 простых числах Мерсенна. Общее же количество простых чисел — бесконечно.

Интересно, что награда за обнаружение первого простого числа длиной в 100 млн. цифр составляет уже 150 тыс. долларов...

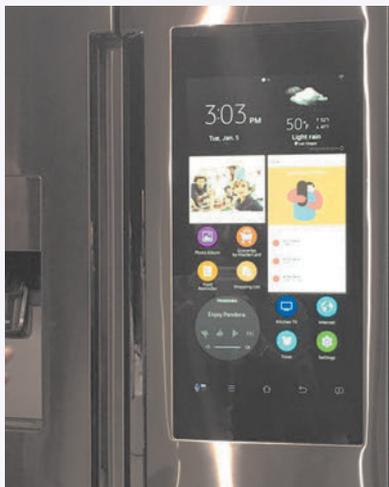
Пять наиболее интересных “устройств будущего” с выставки электроники CES-2016

В начале 2016 года в Лас-Вегасе (США) завершилась крупнейшая выставка потребительской электроники CES-2016. На ней был представлен ряд устройств, которые, по мнению специалистов, вот-вот войдут в повседневную жизнь.

“Айфон” в холодильнике

Южнокорейская компания Samsung привезла на выставку умный холодильник Family Hub Fridge с 21-дюймовой сенсорной панелью, встроенной в дверцу, а также расположенными внутри камерами.

С помощью внешнего экрана можно создавать различные напоминания, выводить изображение с телефона или планшета, управлять музыкой.



Встроенные камеры фотографируют содержимое холодильника каждый раз, когда дверь открывается и закрывается. Снимки отправляются в “облако”, и владелец холодильника сможет посмотреть, какие продукты нужно купить, находясь непосредственно в магазине. В будущем еду можно будет заказывать домой прямо через холодильник.

Сверхтонкий ноутбук

Очередной рекорд по толщине может поставить новый ультрабук-трансформер Lenovo Yoga 900S. Его толщина составляет всего 12,8 мм, а весит устройство ровно 999 граммов.

Ноутбук Lenovo Yoga может раскладываться на 360 градусов. Ноутбук сохранил фирменную особенность всего семей-



ства Yoga — сенсорный экран и способность раскладываться на 360 градусов. Новинка получила 12,5-дюймовый экран, корпус из углеродного волокна и процессор Intel Core m7. Как обещают производители, ноутбук сможет проработать до 10,5 часов в режиме воспроизведения видео от одной подзарядки.

Умный самокат

Умные самокаты Segway уже известны всему миру. В новой версии отсутствует руль, управляется он наклоном коленей. На ее основе был создан Segway Advanced Personal Robot.

По команде из верхней части гаджета выдвигается экран, а по бокам появляются руки-манипуляторы. Робот может помочь владельцу донести сумки, открыть гостям дверь, и способен реагировать на различные голосовые команды.

Гаджет оснащен камерами и может вести видеосъемку. Позиционируется он как помощник по дому, но в будущем робот может превратиться в военного разведчика, полицейского или научиться выгуливать собак.

Кроссовки, способные заменить смартфон

Многие любители бега уже имеют фитнес-браслет, подключенный к смартфону, — чтобы считать количество шагов, километров и потраченных калорий. С кроссовками Gemini ничего из этого не нужно — у них уже есть аналогичное встроенное устройство, которое включается автоматически.

Носить с собой на пробежку смартфон и браслет необязательно, — вся информация хранится в памяти кроссовок и отправляется на телефон, когда бегун возвращается домой с занятий. Gemini 2 могут хранить информацию о пяти последних пробежках.

Создатель кроссовок, компания Under Armour, также обещает, что их обувь достаточно долговечна — спортсмен сможет комфортно пробежать в них более 700 километров. Что касается батареи, которая управляет встроенной электроникой, то она прослужит еще дольше.



Беспилотное воздушное такси будущего

Ehang 184 — это беспилотный летательный аппарат с закрытой кабиной, в которой может разместиться один человек. При этом, оказавшись внутри беспилотника, этот человек получает статус пассажира, а не пилота. Отметим, что среди интерьера кабины вы не найдете элементов управления летательным аппаратом или рычагов, или кнопок. Вместо большого количества неизвестных обывателю тумблеров и механизмов в единственном отсеке Ehang 184 нашлось место пассажирского кресла и планшета. Последний служит для того, чтобы, заняв место в воздушном такси, вы могли указать пункт назначения, а затем наслаждаться видами на землю.

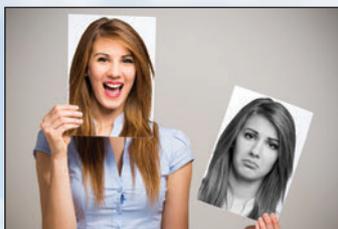


НОВЫЙ ПРОЕКТ

«Первого сентября»



СПЕЦИАЛИСТЫ-ПРАКТИКИ
СВИДЕТЕЛЬСТВО УЧАСТНИКА **О ВОСПИТАНИИ**
ОБ ОТНОШЕНИЯХ О РОДИТЕЛЬСКОЙ ПОЗИЦИИ
АБОНЕМЕНТЫ О САМООЦЕНКЕ **УДОБНОЕ** О ЦЕЛЯХ
О РАБОТЕ О МИРОВОСПРИЯТИИ **ВРЕМЯ** О ЧУВСТВЕ ВИНЫ
О КОММУНИКАЦИИ **О ДЕТЯХ**
ВЕБИНАРЫ
О ДЕТЯХ С ОВЗ **ВОСТРЕБОВАННЫЕ**
О СЕМЬЕ О КОНФЛИКТАХ **ТЕМЫ**
ДОСТУПНАЯ СТОИМОСТЬ **ВИДЕОЗАПИСИ**
О МЕТОДАХ ОБУЧЕНИЯ
О ВЫГОРАНИИ О КАРЬЕРЕ О ВЗАИМОПОНИМАНИИ О СТРЕССЕ
ВОПРОСЫ И ОТВЕТЫ ОНЛАЙН О ЦЕННОСТЯХ О ЛИЧНЫХ КРИЗИСАХ



Видеозаписи вебинаров на сайте
webinar.1september.ru