

Редакционная коллегия

Баракина Т. В.
Бешенков С. А.
Босова Л. Л.
Воеводин Вл. В.
Дергачева Л. М.
Заславская О. Ю.
Захарова Т. Б.
Зенкина С. В.
Кириченко И. Б.
Кравцова А. Ю.
Кузнецов А. А.
Лаптев В. В.
Левченко И. В.
Рыбаков Д. С.
Слинкина И. Н.

Редакция

Босова Л. Л.
главный редактор
Кириченко И. Б.
*заместитель
главного редактора*
Дергачева Л. М.
научный редактор
Губкин В. А.
Коптева С. А.
Кузнецова Е. А.
Федотов Д. В.
Шарапкина Л. М.

Адрес редакции:

119261, г. Москва,
Ленинский пр-т, д. 82/2, комн. 6

Почтовый адрес:

119270, г. Москва, а/я 15

Телефон/факс: (495) 140-19-86

E-mail: readinfo@infojournal.ru

URL: <http://www.infojournal.ru>

Подписные индексы

в каталоге «Роспечать»:

для индивидуальных подписчиков — 81407
для предприятий и организаций — 81408

Подписано в печать 25.05.2018.

Формат 60×90¹/₈. Усл. печ. л. 8,0.

Тираж 2000 экз. Заказ № 408.

Отпечатано в типографии

ООО «Принт сервис групп»

105187, г. Москва, Борисовская ул.,
д. 14, стр. 6, тел./факс: (499) 785-05-18,
e-mail: 3565264@mail.ru

Свидетельство о регистрации

средства массовой информации

ПИ № 77-12068 от 11 марта 2002 г.

Все права защищены. Никакая часть журнала
не может быть воспроизведена в любой форме
или любыми средствами, электронными или
механическими, включая фотографирование,
сканирование, магнитную запись, размещение
в интернете или иные средства копирования
или сохранения информации, без письменного
разрешения издательства.

Журнал входит в Перечень российских
рецензируемых научных изданий ВАК,
в которых должны быть опубликованы
основные научные результаты диссертаций
на соискание ученых степеней доктора
и кандидата наук

© «Образование и Информатика», 2018

ИНФОРМАТИКА В ШКОЛЕ

№ 4 (137) май 2018

Содержание

УРОКИ ИНФОРМАТИКИ

Лапшева Е. Е., Огнева М. В. Образовательный квест по информатике
«Приключения принца Джавы» 4

МЕТОДИЧЕСКАЯ КОПИЛКА

Захарова Т. Б., Битько А. В. Формирование познавательных УУД
при изучении темы «Информация и информационные процессы»
в VII классе 9

Монахова Г. А., Монахов Н. В., Монахов Д. Н. Квесты в обучении
информатике 13

Филиппов В. И., Лабутинов В. Б. Развитие инженерной культуры
школьников 16

Кашей В. В. Релевантность алгоритмизации и программирования
в курсе информатики общеобразовательных организаций. Выбор языка
программирования 20

Савченкова М. В. Scratch-хакатон «Программируем в среде Scratch» 23

**Казарян М. Л., Шахраманьян М. А., Мурынин А. Б., Рихтер А. А.,
Давыдов А., Дементьев И., Игнатьев Д.** Методика построения
3D-моделей ригидных объектов по одному изображению
и ее применение в проектной деятельности школьников 28

Рединова А. А. Формирование метапредметных умений и навыков
на уроках информатики на основе структурирования учебного текста 35

Федосов А. Ю., Маркушевич М. В. Применение свободного
программного обеспечения в учебном процессе как фактор
духовно-нравственного воспитания учащихся 40

Оленев А. А., Малиатаки В. В. Исторические шифры и Maple 44

ЗАДАЧИ

Гаврилова И. В. Развитие алгоритмического мышления учащихся
на основе ментально-эмпирических трит-задач 50

ОЛИМПИАДЫ ПО ИНФОРМАТИКЕ

Путина А. С. Scratch-олимпиада по креативному программированию 56

ЮБИЛЕЙНЫЙ

Уважаемые коллеги!

В начале 2019 года редакцию ИНФО и всех наших читателей ожидают сразу два знаменательных события: во-первых, выйдет в свет юбилейный, 300-й, выпуск журнала «Информатика и образование», и, во-вторых, в нем будут подведены итоги юбилейного, пятнадцатого, конкурса ИНФО.

Первый выпуск научно-методического журнала «Информатика и образование» вышел в свет в августе 1986 года. Создание журнала было вызвано требованием времени — учителям была остро необходима методическая поддержка при решении актуальных задач внедрения в педагогическую практику нового общеобразовательного курса «Основы информатики и вычислительной техники».

Первым главным редактором ИНФО стал Владимир Андреевич Мельников — академик РАН (в то время — АН СССР), в первых номерах журнала были опубликованы статьи академиков Евгения Павловича Велихова и Андрея Петровича Ершова. Участие академии в жизни журнала подчеркивало то значение, которое придавала научная общественность новому школьному предмету, — уже в те годы было понимание того, что информатика будет играть важнейшую роль в развитии образования в XXI веке.

Статьи ведущих ученых, стоявших у истоков школьной информатики, — Александра Андреевича Кузнецова, Михаила Павловича Лапчика, Вадима Семеновича Леднева, Вадима Макарьевича Монахова, Алексея Львовича Семенова, Александра Юрьевича Уварова — не только оказывали существенную помощь учителям в их повседневной практической работе, но и раскрывали значимость и перспективы использования информационных технологий в образовании.

На страницах журнала информатика всегда рассматривалась как основа информатизации, поэтому в нем всегда было два равноправных направления: методика преподавания информатики и информатизация образования. Журнал не только задавал ориентиры в развитии методики преподавания информатики, давал образцы грамотного построения уроков по этому предмету, но и предлагал материалы, в которых известные ученые-теоретики и специалисты-практики формировали и развивали теоретические основы и практические аспекты информатизации образования, использования средств информационно-коммуникационных технологий в образовательном процессе.

За время своего существования школьная информатика претерпела колоссальные изменения, проделав огромный путь от «курса компьютерной грамотности» до полноценного общеобразовательного учебного предмета. И на всей этой длинной дистанции журнал не только отражал те перемены, которые происходили в учебном предмете «Информатика», но и принимал непосредственное участие в его успешном становлении. Авторы программ и школьных учебников по информатике, разработчики электронных образовательных ресурсов, учителя-практики постепенно пополняли авторский коллектив журнала.

Огромное число учителей-новаторов стали авторами журнала благодаря конкурсу ИНФО, который издательство «Образование и Информатика» проводит с ноября 2003 года. Многие педагоги, однажды решив прислать свои работы на конкурс ИНФО, в дальнейшем становились его постоянными участниками, а некоторые — и неоднократными лауреатами и дипломантами. Среди имен авторов нынешних выпусков журналов «Информатика и образование» и «Информатика в школе» (который в 2002 году начал издаваться как приложение к журналу «Информатика и образование», а затем вырос в самостоятельное издание) можно встретить немало тех, кто в свое время становился победителем конкурса.

В настоящее время перед школьной информатикой стоят новые задачи, определяемые новыми направлениями развития, модернизации российской школы. Новые стандарты образования и существенное расширение представлений о требованиях к образовательным результатам, совершенствование ЕГЭ и процедур оценивания учебных достижений школьников, создание новой информационно-образовательной среды и организация сетевого взаимодействия участников образовательного процесса — все эти аспекты обновления находят отражение на страницах журналов «Информатика и образование» и «Информатика в школе».

Наши журналы — верные помощники не только для тех, кто занимается непосредственно школьной информатикой. Современная школа немыслима без информационно-образовательной среды, современное образование невозможно представить без информационно-коммуникационных технологий, поэтому к страницам наших изданий обращаются учителя самых разных предметов, руководители системы образования, разработчики новых средств ИКТ, создатели цифровых образовательных ресурсов.

Мы уверены, что все разнообразные направления развития образовательной системы XXI века найдут отражение в работах нового конкурса ИНФО, и приглашаем вас к участию в нем.

КОНКУРС ИНФО-2018

Издательство «Образование и Информатика»,
Всероссийское научно-методическое общество педагогов
объявляют о проведении в 2018 году юбилейного

**XV Всероссийского конкурса
научно-практических работ
по методике обучения информатике
и информатизации образования
ИНФО-2018,
посвященного выходу в свет
300-го номера журнала «Информатика и образование»**

Руководит конкурсом **Организационный комитет** (далее — Оргкомитет), состоящий из представителей Российской академии образования, ведущих методистов, членов Всероссийского научно-методического общества педагогов, членов редакционных коллегий журналов «Информатика и образование» и «Информатика в школе», сотрудников объединенной редакции журналов.

Цели и задачи конкурса

1. Поддержка и распространение опыта педагогов и образовательных организаций по внедрению в образовательную практику современных методов и средств обучения и управления образованием.
2. Выявление и поддержка талантливых педагогов, методистов, руководителей образовательных организаций и органов управления образованием, заинтересованных в развитии инновационных образовательных технологий.
3. Включение педагогов, методистов, руководителей образовательных организаций и органов управления образованием в деятельность по разработке нового содержания образования, новых образовательных технологий, методик обучения и управления образованием.
4. Создание информационно-образовательного пространства на сайте издательства «Образование и Информатика», а также на страницах журналов «Информатика и образование» и «Информатика в школе» по обмену и распространению опыта внедрения инновационных образовательных технологий.
5. Повышение информационной культуры и информационно-коммуникационной компетентности всех участников образовательного процесса.

Условия участия в конкурсе

1. **К участию в конкурсе могут быть представлены любые работы по методике обучения информатике и информатизации образования.**
2. Участником конкурса может стать любой человек, связанный с работой в системе образования.
3. Возраст участников не ограничен.
4. Участником конкурса может быть индивидуальный заявитель или группа авторов.
5. Участниками конкурса могут быть как граждане России, так и граждане других стран, приславшие свои материалы на русском языке.
6. Заявки на участие в конкурсе принимаются только через заполнение формы на сайте издательства «Образование и Информатика».
7. Форма участия в конкурсе — заочная.
8. **В дополнение к основному конкурсу** каждая работа может быть представлена автором для онлайн-голосования на сайте издательства «Образование и Информатика». Победители онлайн-голосования будут отмечены **специальными дипломами**.

Сроки и этапы проведения конкурса

1. **Работы на конкурс принимаются** с 1 июня по 1 ноября 2018 года включительно. Работы, присланные позже 1 ноября 2018 года, к участию в конкурсе допускаться не будут.
2. **Голосование на сайте** за работы, представленные для онлайн-голосования, будет проходить с 1 декабря 2018 года по 1 января 2019 года включительно.
3. **Итоги конкурса** будут подведены до 1 февраля 2019 года и опубликованы на сайте издательства «Образование и Информатика», а также в журналах «Информатика и образование» № 1-2019 и «Информатика в школе» № 1-2019.
4. **Лучшие работы** будут опубликованы в журналах «Информатика и образование» и «Информатика в школе».

Победители конкурса получают (бесплатно):

1. Диплом от Всероссийского научно-методического общества педагогов и издательства «Образование и Информатика».
2. Электронную подписку на журналы «Информатика и образование» и «Информатика в школе» на 2019 год.
3. По одному печатному экземпляру журналов «Информатика и образование» № 1-2019 и «Информатика в школе» № 1-2019, в которых будут опубликованы итоги конкурса.
4. Авторский печатный экземпляр журнала с опубликованной работой.

**Подробную информацию о конкурсе
вы можете найти на сайте ИНФО:**
<http://infojournal.ru/competition/info-2018/>

Контакты Оргкомитета:
Телефон: +7 (495) 140-1986
E-mail: readinfo@infojournal.ru
<http://www.infojournal.ru/>



Е. Е. Лапшева, М. В. Огнева,

Саратовский национальный исследовательский государственный университет имени Н. Г. Чернышевского

ОБРАЗОВАТЕЛЬНЫЙ КВЕСТ ПО ИНФОРМАТИКЕ «ПРИКЛЮЧЕНИЯ ПРИНЦА ДЖАВЫ»

Аннотация

В статье представлен образовательный квест по математическим основам информатики и программирования, который содержит задания (с решениями и системой оценивания) по темам курса информатики: «Логика», «Системы счисления», «Графы», «Криптография».

Ключевые слова: образовательный квест, информатика, внеурочная деятельность.

Контактная информация

Лапшева Елена Евгеньевна, руководитель образовательного центра непрерывной подготовки IT-специалистов Саратовского национального исследовательского государственного университета имени Н. Г. Чернышевского; *адрес:* 410012, г. Саратов, ул. Астраханская, д. 83, СГУ, факультет КНИИТ; *телефон:* (8452) 21-36-23; *e-mail:* lapsheva@gmail.com

Огнева Марина Валентиновна, канд. физ.-мат. наук, зав. кафедрой информатики и программирования Саратовского национального исследовательского государственного университета имени Н. Г. Чернышевского; *адрес:* 410012, г. Саратов, ул. Астраханская, д. 83, СГУ, факультет КНИИТ; *телефон:* (8452) 21-36-23; *e-mail:* ognevamv@gmail.com

E. E. Lapsheva, M. V. Ogneva,
Saratov State University

EDUCATIONAL QUEST ON INFORMATICS AND PROGRAMMING "ADVENTURE OF PRINCE JAVA"

Abstract

The article describes an educational quest on the mathematical foundations of informatics and programming, that contains tasks (with solutions and an assessment system) on the themes of the informatics course: "Logic", "Number systems", "Graphs", "Cryptography".

Keywords: educational quest, informatics, extracurricular activities.

Активные и интерактивные технологии обучения являются наиболее востребованными и перспективными на современном этапе развития образования. К таким технологиям относятся различного вида игры, круглые столы, мастер-классы, проекты, мозговой штурм и т. д. Особую популярность в последнее время получили образовательные квесты.

Квест в переводе с английского означает «поиск». Обычно это командная игра с приключенческим или детективным сюжетом, в ходе которой для продвижения по сюжету необходимо выполнять различные задания. Квест можно проводить как внутри образовательного учреждения, так и вне его — в классе, в школе, в парке, в летнем лагере и т. д. Возможны квесты с поиском тайников, с элементами ориентирования на местности, а также компьютерные квесты. Задания могут быть различными: творческими, исследовательскими, на заданную тему, физическими и т. д.

Таким образом, квест объединяет в себе различные формы деятельности: исследование, мозговой штурм, работу в группах, дискуссии. Квест позволяет участникам полностью погрузиться в происходящее, проявить свои знания и способности, фантазию и творчество, развить коммуникативные навыки, повышает мотивацию к изучению данной дисциплины [3–5].

Такая форма работы активно применяется в образовательном центре непрерывной подготовки IT-специалистов Саратовского национального исследовательского государственного университета имени Н. Г. Чернышевского. Образовательные квесты по информатике проводятся со школьниками разного возраста, в очной и заочной формах.

В данной статье мы рассмотрим образовательный квест по информатике «Приключения принца Джавы» для школьников VI—VIII классов. Квест неоднократно проводился как в дистанционной, так и в очной формах [1, 2].

Рассмотрим подробнее организацию проведения данного мероприятия.

Вариант 1. Заочная (дистанционная) форма проведения квеста.

Квест проводится в четыре этапа.

1-й этап. Публикация объявления на сайте организаторов, рассылка информации по школам.

2-й этап. Формирование команд, оформление заявок, регистрация команд (со стороны участников квеста). Команда должна состоять из трех—пяти человек, среди которых выбирается капитан, он должен зарегистрироваться на сайте и впоследствии будет отправлять ответы на задания.

3-й этап. Публикация заданий и их выполнение командами.

Для проведения квеста на сайте организаторов удобно организовать отдельный курс. Задания открываются по очереди, на решение каждого дается от двух до пяти дней в зависимости от сложности. Авторы квеста для данной формы проведения применяют LMS Moodle, используемую в Саратовском го-

сударственном университете специально для обучения информатике школьников и студентов (<http://school.sgu.ru>). В течение выделенного срока команда должна отправить решение задания.

Каждое следующее задание открывается после того, как закрылось предыдущее. После прохождения каждого задания публикуются промежуточные итоги.

Критерии оценивания заданий:

- правильный ответ;
- логичность, связность, четкость описания решения;
- отсутствие небрежности в оформлении и грубых орфографических ошибок, мешающих правильному восприятию;
- умение увидеть и использовать для эффективного решения ключевой момент (для каждой задачи такой момент описан отдельно).

Максимальное количество баллов — разное для каждого задания и зависит от сложности задания.

4-й этап. Подведение итогов и награждение победителей.

На протяжении всех этапов квеста на сайте организаторов работает форум, где можно задать вопросы по заданиям (на них отвечают члены жюри) и где размещаются объявления организационного характера.

Вариант 2. Очная форма проведения квеста.

В очной форме мероприятие проводится на базе школы или летнего лагеря. Удачно, если школа может выделить несколько классов — пунктов проверки заданий или если в летнем лагере можно организовать такие пункты на территории. Но достаточно будет и двух помещений: одно (это обычно спортивный или актовый зал либо большой холл) — для размещения команд, другое — для размещения пунктов проверки заданий.

Разделение на команды можно осуществить заранее, по желанию школьников, а можно устроить жеребьевку накануне мероприятия. Так же как и в случае дистанционного проведения, команда состоит из трех—пяти участников, среди которых выбирается капитан — ответственный за сдачу заданий.

Для каждой команды организуется отдельное место, обычно это парта, на которой приготовлены ручки и бумага для черновиков, и необходимое количество стульев. В этом же помещении находятся вспомогательные материалы — плакаты с алфавитом шифровки или шифровальный диск, карта и т. д., а также в режиме онлайн (на переносной доске) заполняется таблица с итогами решения командами каждого задания.

Жюри разбивается на группы по количеству заданий. В каждой группе должно быть не менее двух человек и еще несколько членов жюри должны быть готовы помочь на каждом этапе, если одновременно будут сдавать задания больше двух команд. Обычно такое бывает на первых этапах. Так как каждое задание связано с каким-то пунктом на карте, то в комнате для проверки расставляются таблички с названиями этих пунктов, а также обманки — таблички с пунктами, которых нет в заданиях.

Мероприятие начинается с того, что каждая команда получает первое задание. Как только решение найдено, капитан выбирает представителя команды, который будет объяснять полученное решение жюри. Предположим, что команда решила задание и получила в ответе название города — Париж. Представитель команды должен найти табличку с надписью «Париж», подойти

к членам жюри, которые сидят около этой таблички, и отчитаться по решению задания. Если задание решено правильно, то жюри начисляет соответствующие баллы и выдает команде следующее задание.

Критерии оценивания заданий такие же, как и в случае проведения квеста в заочной форме.

Квест заканчивается, если все команды сдали задания или прошло определенное время, обычно это полтора часа.

Если команда не сдала задание с первой попытки, возможно сдать его повторно (всего три раза), за каждую следующую попытку часть баллов снимается. Можно также пропустить задание (и не получить за него баллы).

Обычно темами задания квеста являются следующие: «Логика», «Системы счисления», «Графы», «Криптография».

Далее предлагаются текст вступления и задания с решениями одного из таких квестов про приключения выдуманного принца по имени Джава.

Вступление.

Принц Джава скучал...

В этот послеобеденный час он обычно играл с наставником в шахматы или отгадывал головоломные задачи, которые старик придумывал сам. Но сейчас наставника не было — он уехал на несколько дней, и оказалось, что принцу его очень не хватает. Он уже было решил сыграть в шахматы сам с собой и даже направился к шкафу, как вдруг его осенила идея получше. На самой верхней полке в шкафу хранилась огромных размеров шахматная доска, которую старик никогда не разрешал брать для игры. «Мал еще играть в такие шахматы», — усмехался он. И вот теперь такой отличный шанс — хоть одним глазком посмотреть, что ж там за шахматы-то такие.

Джава решительно взобрался наверх по специальной лесенке, снял доску, положил ее на стол...

Задание 1.

К его удивлению, внутри не оказалось шахматных фигур, а только три маленьких шкатулочки с выбитыми на них надписями...

На первой было написано: «В этой шкатулке яд», на второй: «В этой шкатулке послание», на третьей: «Яд в шкатулке номер 2».

«Да... — подумал Джава, — и знать бы еще, что из этого правда...»

Какую шкатулку надо открыть принцу, если известно, что из трех утверждений в лучшем случае только одно истинно и что в каждой шкатулке обязательно что-то есть — или яд, или послание?

Помогите принцу Джава открыть нужную шкатулку.

Решение.

В соответствии с условием заполним следующую таблицу:

Шкатулка 1	Шкатулка 2	Шкатулка 3
В этой шкатулке яд	В этой шкатулке послание	Яд в шкатулке номер 2
Утверждение 1	Утверждение 2	Утверждение 3

Можно выделить два ключевых момента, которые сразу приводят нас к решению:

- 1) «В лучшем случае одно утверждение истинно» — это значит, что либо истинно одно утверждение из трех, либо ни одно утверждение не истинно, т. е. всего получаем четыре варианта;
- 2) второе и третье утверждения противоречат друг другу.

В соответствии с п. 2, из четырех вариантов, которые мы получили в п. 1, остаются только два: второе утверждение истинно, а третье ложно, и наоборот. И соответственно, первое утверждение в любом случае будет ложным. А так как в нем было сказано, что в шкатулке яд, значит, на самом деле в ней послание.

Так как в условии явно не было сказано, что послание только в одной шкатулке, можно более подробно рассмотреть, что будет в шкатулках 2 и 3.

Пусть истинно утверждение 2 «В этой шкатулке послание», тогда утверждение 3 «Яд в шкатулке номер 2» — ложно, т. е. в шкатулке 2 яда нет. То есть в этом случае во второй шкатулке послание, а что в третьей — неизвестно.

Пусть истинно утверждение 3 «Яд в шкатулке номер 2», тогда утверждение 2 «В этой шкатулке послание» — ложно. Утверждение 3 говорит о том, что яд в шкатулке номер 2, а из ложности утверждения 2 также вытекает, что в шкатулке 2 яд.

Таким образом, мы все равно получаем, что наверняка можно открывать только шкатулку номер 1, так как там послание будет находиться в любом случае.

Замечание. Есть иной подход к решению, когда рассматриваются варианты расположения яда и послания в шкатулках, а потом анализируется истинность или ложность соответствующих утверждений. На наш взгляд, это гораздо более трудоемкое решение, однако при соблюдении всех критериев оценка за такое решение не снижается.

Ответ. Послание в шкатулке 1.

Задание 2.

Принц Джава был уверен, что он угадал правильно, но все же открывал шкатулку с некоторым волнением в душе. В шкатулке лежали два листочка бумаги. «Неужели это послание можно прочитать? Или наставник просто посмеялся надо мной?» — думал Джава, с недоумением глядя на беспорядочно разбросанные буквы. Но постепенно у него стали появляться идеи...

Помогите принцу Джава прочитать послание.

Листок 1.

Т	П	П	Р	Ь	Я	И	Л
И	В	К	Т	Х	К	С	Н
О	А	Д	А	П	Д	И	Ф
Ц	Р	Е	В	А	А	У	Т
В	Ч	Ь	А	М	Е	Т	О
Т	О	Т	Р	В	С	В	Т
Х	И	Е	А	О	Д	Р	Ц
А	С	Д	Г	Т	А	О	В

Листок 2.

	Х		Х			Х	
					Х		
Х		Х					Х
				Х			
	Х					Х	
Х					Х		
		Х					Х
				Х			Х

Решение.

Даны два листка бумаги.

Первый листок представляет собой зашифрованное послание, второй — так называемую решетку. Крестиками обозначены «окошки», т. е. попросту дырочки. Если наложить такую решетку на послание, то в окошке будут видны какие-то буквы, эти буквы надо выписать.

Получится первая часть фразы:

приходитчетвергв

Или, если расставить пробелы:

Приходи в четверг в

Далее решетка поворачивается по часовой стрелке на 90 градусов и вновь накладывается на послание. Получается вторая часть фразы:

пятнадцатътридц

или

пятнадцать тридц

Еще раз повернем по часовой стрелке на 90 градусов, наложим на послание, выпишем буквы, которые видны в окошках:

тьвкафеумостасто

или

ть в кафе у моста сто

И, наконец, повернув последний раз, получим окончание фразы:

ликсправаотвхода

или

лик справа от входа

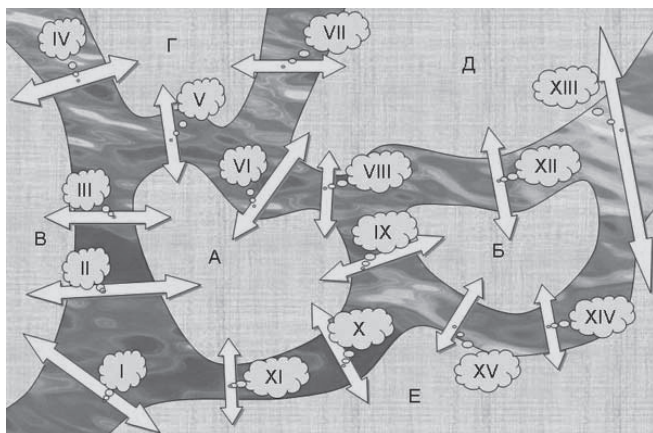
Ответ. Приходи в четверг в пятнадцать тридцать в кафе у моста, столик справа от входа.

Задание 3.

Итак, некто приглашал Джаву в кафе у моста. Но в городе было 15 мостов и всего одно кафе. И все жители знали: чтобы попасть в это кафе, надо обойти все мосты точно по одному разу.

Решение должно содержать:

- 1) схему обхода мостов;
- 2) начальную и конечную точки обхода;
- 3) математическое объяснение вашего варианта решения задачи;



4) ответы на следующие вопросы:

- Единственная ли предложенная вами схема обхода мостов или есть еще другие варианты? Если они есть, то предложите еще несколько.
- Единственная ли пара точек начала и конца обхода мостов? Если не единственная, то перечислите все варианты.

Решение.

Если каждый остров обозначить точкой, а мост — линией, получится фигура, которая называется графом. Точки будем называть вершинами, линии — ребрами графа. Если в вершине сходится четное число ребер, то такую вершину назовем четной, если нечетное — то нечетной. Осуществить обход графа, при котором каждое ребро посещается в точности один раз, можно только лишь в двух случаях:

- 1) все вершины четные. Тогда мы возвращаемся в ту же вершину, из которой вышли;
- 2) существуют ровно две нечетные вершины. Тогда одна из них будет началом обхода, другая — концом.

В нашем случае имеются две нечетные вершины — это вершины, соответствующие островам Г и Д. Следовательно, обход можно начинать в одной из них, заканчивать в другой. Что касается остальных вершин, то, зайдя в любую из них, мы гарантированно выйдем из нее по еще не пройденному ребру, так как количество ребер, сходящихся в ней, — четное.

Замечание. При объяснении не обязательно использовать терминологию теории графов.

Ответ. Полностью приводить ответ мы не будем, так как маршрутов может быть много. Начальная и конечная точка обхода — это острова Г и Д (Г — начальная точка, тогда Д — конечная, и наоборот), других вариантов не существует.

Задание 4.

За крайним столиком уже сидел человек. Был он сгорблен и худ, с густой седой бородой. Несмотря на то что в кафе было жарко (или это Джава стало жарко после прогулки по мостам?), старик не снял капюшон, так что лицо его было почти неразличимо. «Ты все-таки пришел...» — проговорил он, и что-то знакомое почудилось Джава в его интонациях. «Ну что ж, присаживайся, принц Джава. Ты, наверное, хочешь знать, кто я и зачем пригласил тебя сюда. Ну что ж, я расскажу немного о себе...

В юности у меня была несчастная любовь. Было нас тогда четыре юноши и четыре девушки. Каждый был влю-

блен в кого-то одного, в каждого был влюблен кто-то один, но, увы, все в нашей компании любили без взаимности. Конечно, никто никому прямо не признавался в своих привязанностях, но кое-что все-таки было известно.

Андриан был равнодушен к девушке, которая была влюблена в юношу, которому страшно нравилась Теодора.

В Марселину был влюблен юноша, который нравился девушке, любимой Бенджамином. А вот в девушку, любящую Джереми, угораздило влюбиться Климента.

Еще я точно знал, что Зося не любит Бенджамина, а сама Зосю не любит тот, к кому равнодушна Геля. Но я не знал, кто же любит Андриана, а мне так хотелось это узнать...»

«Видно, старик — это и есть Андриан, — подумал Джава. — Вот интересно, узнал ли он, в конце концов, имя влюбленной в него девушки? Или это невозможно?»

За столом повисло молчание, но старик явно ждал, что ответит Джава...

Решение.

Будем обозначать неизвестного юношу буквой «ю», а неизвестную девушку буквой «д».

Из условия, что каждый влюблен в кого-то одного, в каждого влюблен кто-то один и все любят без взаимности, получается такая схема (обозначим ее *):

$$\begin{array}{ccc} д1 \rightarrow ю1 \rightarrow д2 & & \\ \uparrow & & \downarrow \\ ю4 & & ю2 \\ \uparrow & & \downarrow \\ д4 \leftarrow ю3 \leftarrow д3 \end{array}$$

в которой $A \rightarrow B$ означает, что А любит В.

Тогда условие «В Марселину был влюблен юноша, который нравился девушке, любимой Бенджамином», схематично записывается следующим образом:

$$B \rightarrow д \rightarrow ю \rightarrow M \quad (1)$$

Условие «Андриан был равнодушен к девушке, которая была влюблена в юношу, которому страшно нравилась Теодора», превращается в следующую цепочку:

$$A \rightarrow д \rightarrow ю \rightarrow T \quad (2)$$

Условие «в девушку, любящую Джереми, угораздило влюбиться Климента» запишется как:

$$K \rightarrow д \rightarrow D \quad (3)$$

И, наконец, «Зосю не любит тот, к кому равнодушна Геля»:

$$Г \rightarrow ю \rightarrow З \quad (4)$$

Кроме того, еще осталось условие «Зося не любит Бенджамина».

Цепочку (3) можно поставить на любое место схемы (*), например:

$$\begin{array}{ccc} д1 \rightarrow K \rightarrow д2 & & \\ \uparrow & & \downarrow \\ ю4 & & Д \\ \uparrow & & \downarrow \\ д4 \leftarrow ю3 \leftarrow д3 \end{array}$$

Тогда для цепочки (1) остаются два варианта:

$$\begin{array}{ccc} M \rightarrow K \rightarrow д2 & & \\ \uparrow & & \downarrow \\ ю4 & & Д \\ \uparrow & & \downarrow \\ д4 \leftarrow B \leftarrow д3 \end{array}$$

или

$$\begin{array}{ccc} d1 & \rightarrow & K \rightarrow M \\ \uparrow & & \downarrow \\ B & & D \\ \uparrow & & \downarrow \\ d4 & \leftarrow & y3 \leftarrow d3 \end{array}$$

Таким образом, Марселина — это или d1, или d2.

Аналогично, для цепочки (2) тоже существуют только два варианта:

$$\begin{array}{ccc} T & \rightarrow & K \rightarrow d2 \\ \uparrow & & \downarrow \\ y4 & & D \\ \uparrow & & \downarrow \\ d4 & \leftarrow & A \leftarrow d3 \end{array}$$

или

$$\begin{array}{ccc} d1 & \rightarrow & K \rightarrow T \\ \uparrow & & \downarrow \\ A & & D \\ \uparrow & & \downarrow \\ d4 & \leftarrow & y3 \leftarrow d3 \end{array}$$

Таким образом, Теодора — это тоже или d1, или d2.

Для Зоси и Гели остается d3 и d4, тогда из условия «Зосю не любит тот, к кому равнодушна Геля», получаем, что Зося — это d3, Геля — это d4:

$$\begin{array}{ccc} d1 & \rightarrow & K \rightarrow d2 \\ \uparrow & & \downarrow \\ y4 & & D \\ \uparrow & & \downarrow \\ \Gamma & \leftarrow & y3 \leftarrow 3 \end{array}$$

Так как Зося не любит Бенджамина, то для Бенджамина остается только y4:

$$\begin{array}{ccc} d1 & \rightarrow & K \rightarrow d2 \\ \uparrow & & \downarrow \\ B & & D \\ \uparrow & & \downarrow \\ \Gamma & \leftarrow & y3 \leftarrow 3 \end{array}$$

Тогда Андриан — это y3:

$$\begin{array}{ccc} d1 & \rightarrow & K \rightarrow d2 \\ \uparrow & & \downarrow \\ B & & D \\ \uparrow & & \downarrow \\ \Gamma & \leftarrow & A \leftarrow 3 \end{array}$$

Из цепочки (1) Марселина — это d2, из цепочки (2) Теодора — это d1:

$$\begin{array}{ccc} T & \rightarrow & K \rightarrow M \\ \uparrow & & \downarrow \\ B & & D \\ \uparrow & & \downarrow \\ \Gamma & \leftarrow & A \leftarrow 3 \end{array}$$

Ответ. Андриана любит Зося.

Задание 5.

«Ну что ж, — судя по голосу, старик был доволен ответом. — Ты правильно догадался. И я тоже смог тогда догадаться, но мне было это уже не очень важно, потому что в тот год я встретил свою будущую жену. Мы оба учились тогда в университете. Мне было 40 лет, а ей исполнилось 34 — она была на год младше. Спустя год после нашего знакомства мы поженились, и еще через год у нас родился сын. С тех пор прошло 100 лет... Сейчас у меня уже 10 внуков!» Старик помолчал. «Время летит слишком быстро, иногда я даже забываю, сколько же лет мне исполнилось в этом году. Ты не поможешь мне?»

Сколько лет сейчас герою этого рассказа?

Ответ должен содержать не только число лет героя, но и подробное решение задачи.

Решение.

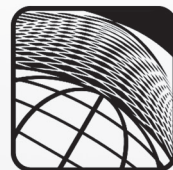
Из условия: «Мне было 40 лет, а ей исполнилось 34 — она была на год младше» ясно, что в той системе счисления, в которой считает старик, после 34 идет 40, т. е. цифра 4 — наибольшая в этой системе. Значит, это пятеричная система счисления. Старiku тогда было 20 лет, если считать в десятичной системе счисления.

В пятеричной системе счисления с тех пор прошел год, еще год, и еще 100 лет, т. е. всего 102 года. В десятичной системе счисления это 27 лет. То есть старику сейчас 47 лет.

«Ты опять ответил правильно, — судя по голосу, старик улыбался. — Значит, я был неправ, думая, что ты еще слишком мал». С этими словами старик откинул капюшон, и принц Джава узнал своего любимого наставника. Почему-то его это не очень удивило...

Список использованных источников

1. Кудрина Е. В., Лапшева Е. Е., Огнева М. В. Опыт организации дистанционной внеклассной работы школьников // Информационные технологии для Новой школы. Материалы конференции. Санкт-Петербург, 26–28 марта 2010 года. СПб.: ГОУ ДПО ЦПКС СПб «Региональный центр оценки качества образования и информационных технологий», 2010.
2. Лапшева Е. Е., Огнева М. В. Опыт проведения обучающей игры по информатике // Информационные технологии в образовании: Материалы VII Всероссийской научно-практической конференции. г. Саратов, 2–3 ноября 2015 года. Саратов: Издательский центр «Наука», 2015.
3. Магич Е. А., Скулачев А. А. Как создавать образовательный квест // Народное образование. 2015. № 1.
4. Николаева Н. В. Образовательные квест-проекты как метод и средство развития навыков информационной деятельности у учащихся // Вопросы Интернет-образования. 2002. № 7.
5. Осяк С. А., Султанбекова С. С., Захарова Т. В., Яковлева Е. Н., Лобанова О. Б., Плеханова Е. М. Образовательный квест — современная интерактивная технология // Современные проблемы науки и образования. 2015. № 1-2. <https://www.science-education.ru/ru/article/view?id=20247>



Т. Б. Захарова,
Академия социального управления, г. Москва,

А. В. Битько,
гимназия № 16, г. Химки, Московская область

ФОРМИРОВАНИЕ ПОЗНАВАТЕЛЬНЫХ УУД ПРИ ИЗУЧЕНИИ ТЕМЫ «ИНФОРМАЦИЯ И ИНФОРМАЦИОННЫЕ ПРОЦЕССЫ» В VII КЛАССЕ

Аннотация

В статье обсуждается накопленный авторами педагогический опыт организации образовательного процесса по информатике в VII классе общеобразовательной школы с акцентом на формировании познавательных универсальных учебных действий, в частности, при изучении темы «Информация и информационные процессы». Методический подход описан на примере формирования умений классификации.

Ключевые слова: универсальные учебные действия, информатика, информационные процессы.

Контактная информация

Захарова Татьяна Борисовна, доктор пед. наук, профессор, профессор кафедры информационно-коммуникационных технологий Академии социального управления, г. Москва; *адрес:* 129281, г. Москва, Староватутинский проезд, д. 8; *телефон:* (495) 472-32-08, доб. 149; *e-mail:* t_zakh@mail.ru

Битько Анна Владимировна, учитель информатики гимназии № 16, г. Химки, Московская область; *адрес:* 141411, Московская область, г. Химки, ул. Зеленая, д. 22; *телефон:* (495) 570-15-12; *e-mail:* kab16@mail.ru

T. B. Zakharova,
Academy of Public Administration, Moscow,
A. V. Bitco,
Gymnasium 16, Khimki, Moscow Region

THE FORMATION OF COGNITIVE UNIVERSAL EDUCATIONAL ACTIONS IN STUDYING THE THEME "INFORMATION AND INFORMATION PROCESSES" IN 7TH GRADE

Abstract

The article discusses the authors' pedagogical experience in the organization of educational process in Informatics in the 7th grade of secondary school with an emphasis on the formation of cognitive universal educational actions, in particular, in the study of the theme "Information and information processes". The methodical approach is described on the example of formation of abilities of classification.

Keywords: universal educational actions, informatics, information processes.

Основным требованием ФГОС основного общего образования [3] является формирование личностных, метапредметных и предметных образовательных результатов, адекватных актуальным потребностям личности, общества, государства. Важнейшим становится формирование универсальных учебных действий, среди которых особое место занимают познавательные УУД.

Как известно, познавательные универсальные учебные действия включают: общеучебные, знаково-символические, логические действия, а также действия постановки и решения учебных задач. Они формируются, безусловно, с учетом возрастных и психологических особенностей обучающихся при изучении всех предметов, в том числе информатики.

Рассмотрим кратко **методический подход к их формированию при изучении темы «Информация и информационные процессы» школьного курса информатики в VII классе**, определив, на какие соответствующие виды учебной деятельности надо обратить особое внимание (см. табл.).

Конечно, формируемые УУД и соответствующие им виды учебной деятельности тесно взаимосвязаны, фактически на всех учебных занятиях по информатике систематически формируются общеучебные, знаково-символические, логические универсальные действия и действия решения учебных задач. При этом следует особо отметить группу знаково-символических действий, которые играют немаловажную роль в познании, особенно в условиях развития современного информационного общества. Знаково-символические УУД связаны с представлениями об окружающей действительности, формированием образа изучаемых объектов, в связи с чем они являются основой для формирования в целом системы универсальных учебных действий.

Следует отметить, что основные аспекты изучения темы «Информация и информационные процессы» рассмотрены в целом ряде работ [2 и др.], при этом, на наш взгляд, основными из этих аспектов в VII классе являются следующие:

- расширение и систематизация представлений обучающихся о знаках и знаковых системах;
- классификация знаков по способу восприятия: пиктограммы и символы;
- систематизация представлений о языке как знаковой системе;
- установление общего и различий в естественных и формальных языках;
- обоснование универсальности двоичного кодирования.

Заметим, что понятие «знак» обучающиеся применяют еще на ступени начального общего образования. На ступени

Таблица

Формируемые познавательные УУД и соответствующие виды учебной деятельности при изучении темы «Информация и информационные процессы» школьного курса информатики в VII классе

№ п/п	Формируемые познавательные УУД	Соответствующие виды учебной деятельности
1	Владение информационнологическими умениями: определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинноследственные связи, строить логическое рассуждение, умозаключение (индуктивное, дедуктивное и по аналогии) и делать выводы	Определение понятия. Установление общего и различий. Установление соответствия между понятиями и их описаниями. Анализ объектов с целью выделения признаков. Выбор оснований классификации объектов. Классификация одного объекта по разным основаниям. Сравнение. Выдвижение гипотез и их обоснование. Синтез — составление целого из частей, в том числе самостоятельное достраивание с восполнением недостающих компонентов. Построение логического рассуждения. Вывод формулы
2	Владение основными универсальными умениями информационного характера, такими как: поиск и выделение необходимой информации, применение методов информационного поиска; структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий	Определение необходимых ключевых поисковых слов и запросов. Структурирование материала, умение показывать связи между отдельными частями. Решение задач с применением кругов Эйлера. Решение задач построением дерева
3	Владение информационным моделированием как основным методом приобретения знаний: умение преобразовывать объект из чувственной формы в пространственнографическую или знаково-символическую модель; умение строить разнообразные информационные структуры для описания объектов; умение «читать» таблицы, графики, диаграммы, схемы и т. д., самостоятельно перекодировать информацию из одной знаковой системы в другую; умение выбирать форму представления информации в зависимости от стоящей задачи	Построение схем состава. Чтение диаграмм. Работа с интерактивной схемой. Построение кластера. Построение ментальной карты

основного общего образования формируется понимание общепредметной сущности понятия «знак» (рис. 1), общеучебные умения анализа, сравнения, классификации.

Не останавливаясь подробно на всех видах учебной деятельности, обеспечивающих сформированность познавательных универсальных учебных действий при

изучении темы «Информация и информационные процессы» школьного курса информатики в VII классе, рассмотрим кратко **методический подход к формированию умения классификации изучаемых объектов.**

Относительно классификации объектов заметим, что она используется как в науке, так и во многих видах практической деятельности специалистов разного профиля. Понимается классификация как мыслительная операция, состоящая в разделении объектов на группы по определенным признакам.

Считаем, что одна из основных целей учителя — сформировать умение выделять основание классификации, а также умение проводить классификацию объектов по нескольким основаниям. Для реализации этой задачи с целью активизации мышления обучающихся можно начать обсуждение с групп объектов, хорошо знакомых ребятам из их жизненного опыта. Например, школьники в VII классе не затрудняются в приведении оснований для представления о разбиении класса (коллектива учащихся) на группы, например, по гендерному признаку, на группы занимающихся английским языком у разных преподавателей, по году рождения. При этом сначала обучающиеся могут не обратить внимания на некоторые аспекты при классификации, например, когда рассматривают в качестве основания для деления занятия



Рис. 1

в каких-либо организациях дополнительного образования. Но ведь один и тот же ребенок может заниматься и музыкой, и спортом. Поэтому уже в данном задании необходимо обратить внимание на возможность следующего уровня классификации в получившихся группах.

Один из приемов формирования понимания основания классификации — обсуждение неудачной классификации, например, такой классификации в магазине: женская, детская, домашняя и резиновая обувь. И вот примеры вопросов для обсуждения с обучающимися: «Можно ли считать это классификацией? Какие основания для классификации могли бы вы привести?»

Далее с точки зрения задач изучения темы «Информация и информационные процессы» можно предложить вопрос о классификациях в этом учебном материале. Например, уже на одном из первых уроков по теме «Информация и информационные процессы» происходит обсуждение, в результате которого формируется и понятие информации для человека, и понятие сигнала. В нашем случае мы берем определение из школьного учебника по информатике Л. Л. Босовой, А. Ю. Босовой «Информация для человека — это содержание сигналов (сообщения), воспринимаемых человеком непосредственно или с помощью специальных устройств, расширяющих его знания об окружающем мире и происходящих в нем процессах» [1].

Здесь важно отметить, что знакомой классификацией для обучающихся является классификация сигналов по физической природе: звуковые, световые, тепловые, механические, электрические. Совершенно новым является деление сигналов по множеству принимаемых ими значений на непрерывные и дискретные. «Непрерывный сигнал принимает бесконечное множество значений из некоторого диапазона. Между значениями, которые он принимает, нет разрывов. Дискретный сигнал принимает конечное число значений. Все значения дискретного сигнала можно пронумеровать целыми числами» [1]. Понимание понятия дискретности необходимо в дальнейшем для изучения понятия дискретизации при представлении информации.

Прежде чем начать рассматривать виды информации, следует организовать обсуждение следующих вопросов: «С какой целью разрабатывается классификация изучаемых объектов? Для чего в науке при изучении каких-либо объектов часто стремятся распределить эти объекты по группам в соответствии с определенными признаками?» Наряду с некоторыми общими свойствами информации и общими закономерностями работы с ней существуют особенности протекания информационных процессов для информации разных видов. И если мы знаем, к какому виду относится информация, то сможем подобрать более подходящие методы и средства для работы с ней.

Целесообразно остановиться на вопросах: «Какие же виды информации можно выделить? И что можно выбрать в качестве оснований деления?» Во многих школьных учебниках по информатике рассматриваются следующие виды информации: визуальная, аудиальная, обонятельная, вкусовая, тактильная.

Можно обратиться к учащимся с вопросом про связь информации с восприятием (и осознанием) человеком поступающих сигналов. Тогда ребята смогут сделать вывод, что информацию можно разделить по тому, как она воспринимается человеком, т. е. разделить по способу восприятия.

В игровой форме можно обсудить, как одну и ту же информацию можно по-разному представить, и предложить в качестве основания классификации выбрать форму представления.

Для того чтобы учащиеся смогли предложить еще одну классификацию, можно предложить рассмотреть рисунок (рис. 2).

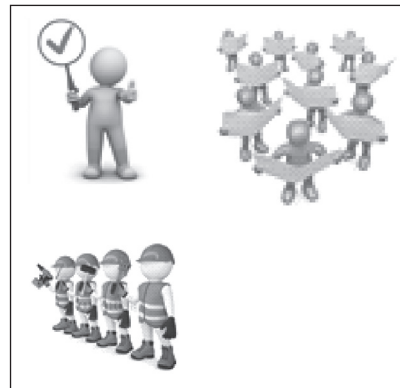


Рис. 2

Как правило, сделав несколько предположений, формулируются следующие виды по степени значимости: личная, массовая, специальная.

Обратив внимание учащихся на определение из учебника Л. Л. Босовой, А. Ю. Босовой, что «информация для человека — это содержание сигналов <...>, расширяющих его знания об окружающем мире и происходящих в нем процессах» [1], можно акцентировать внимание на выделении двух видов знаний. Для того чтобы догадаться, о каких двух видах знаний может идти речь, каждый ученик называет конкретный пример своих знаний, начиная предложение со слов: «Я знаю...» В разных классах это обсуждение проходит по-разному. В некоторых классах ребята приводят примеры только одного из видов знаний, тогда учитель может привести свой пример. На тот редкий случай, когда у обучающихся возникают затруднения с примерами, можно использовать интерактивную презентацию, в которой предлагается разделить приведенные примеры и сформулировать определение каждого вида (факты и правила, декларативные и процедурные знания). После этого в игровой форме ребята определяют, к какому виду относится тот или иной конкретный пример знаний.

Таким образом, рассматриваются виды информации по способу восприятия человеком, по форме представления, по степени значимости и информация как знания.

В ходе дальнейшего обсуждения обучающиеся выясняют, что в качестве критерия классификации могут быть выбраны и другие параметры, например, информацию можно классифицировать по характеру носителя информации, по характеру источника информации и т. п.

Так у обучающихся формируется умение классифицировать, рассматриваемое сегодня как познавательное универсальное учебное действие.

В заключение подчеркнем, что учебный материал темы «Информация и информационные процессы» общеобразовательного курса информатики в VII классе обладает огромным потенциалом для формирования целого ряда познавательных УУД. Нами разработана и успешно апробирована методика формирования таких познавательных УУД, как:

- анализ объектов с целью выделения признаков;

- выбор оснований для классификации объектов;
- классификации одного объекта по разным основаниям;
- сравнение;
- выдвижение гипотез и их обоснование.

Практика обучения показала целесообразность использования предлагаемого нами методического подхода формирования познавательных УУД при изучении темы «Информация и информационные процессы» школьного курса информатики в VII классе.

НОВОСТИ

«Цифровая школа» изменит роль педагогов в образовательных организациях

В выступлении на Петербургском международном экономическом форуме министр просвещения Российской Федерации О. Ю. Васильева прокомментировала, что изменит в сфере образования реализация проекта «Цифровая школа».

Министр уточнила, что в настоящее время ведомство дорабатывает паспорт проекта с учетом приоритетов, обозначенных в новых майских указах Президента Российской Федерации.

«Работа над проектом «Цифровая школа» важна для всех, кто сегодня задействован в системе образования. Проект позволит обеспечить обновление содержания образования и даст возможность нашим школьникам свободно и в то же время безопасно ориентироваться в цифровом пространстве. Благодаря проекту у родителей появится больше возможностей изучать интересы и способности своего ребенка. Для педагогов цифровизация снизит административную нагрузку, высвобождая время для повышения качества своих образовательных программ», — пояснила Ольга Юрьевна.

Министр просвещения подчеркнула, что реализация проекта «Цифровая школа» предусматривает также повышение квалификации педагогов: ««Цифровая школа» обеспечит повышение квалификации педагогов и оснащение школ необходимой инфраструктурой, что в свою очередь позволит максимально эффективно использовать возможности еще одного нашего масштабного ресурса — Российской электронной школы, в которой заключен значительный объем онлайн-материалов для учащихся и учителей».

О. Ю. Васильева подчеркнула, что запуск приоритетного проекта «Цифровая школа» — важнейший этап на пути реализации программы «Цифровая экономика в Российской Федерации». В рамках проекта будет создана цифровая экосистема, благодаря которой станет возможным переход к автоматизированному делопроизводству, работе с цифровыми инструментами, использованию широкого спектра современных методик и технологий обучения.

Реализация проекта повлечет за собой обновление содержания образования и изменение роли учителя, который станет куратором, ориентирующим ребенка в соответствии с его запросами и приоритетами, максимально индивидуализирует траектории обучения школьников.

Список использованных источников

1. Босова Л. Л., Босова А. Ю. Информатика: учебник для 7 класса. М.: БИНОМ. Лаборатория знаний, 2015.

2. Захарова Т. Б., Захаров А. С. Информатика как обязательный учебный предмет в системе общего образования // Наука и школа. 2015. № 5.

3. Федеральный государственный образовательный стандарт основного общего образования (в ред. Приказа Минобрнауки России от 29.12.2014 № 1644). http://минобрнауки.рф/документы/938/файл/749/приказ_Об_утверждении_1897.pdf

Появится возможность прохождения аттестации и сдачи государственной аттестации в цифровой форме. Родители смогут в режиме реального времени узнавать, находится ли ребенок в школе и какие результаты он показывает в образовательной деятельности.

В рамках проекта будут созданы цифровая платформа и информационный ресурс «Цифровая школа», которые обеспечат возможность использования и интеграции существующих на рынке решений (в том числе электронных журналов и дневников), будут обеспечены конкурентная среда и возможность взаимодействия государственного и частного сектора в сфере образования.

«Благодаря реализации проекта российские школьники смогут применять полученные компетенции в условиях стремительно развивающегося цифрового мира. Таким образом, будет сформирована преемственная линия подготовки кадров, способных решать глобальные задачи для нужд цифровой экономики», — резюмировала министр просвещения.

Реализация проекта рассчитана на период с 2018 по 2025 год. К 2025 году должны быть созданы институциональные и инфраструктурные условия, обеспечивающие равный доступ обучающихся к качественному общему образованию, уровень которого характеризуется положительной динамикой в значении показателей национальных и международных сравнительных исследований, а также возможностью использования цифровых образовательных ресурсов во всех школах.

Предложение о запуске проекта «Цифровая школа» в декабре на заседании президиума президентского совета по стратегическому развитию и приоритетным проектам озвучил премьер-министр Дмитрий Медведев. «Самое главное, что дети проще, чем родители, осваивают все современные технологии и глубже погружаются в цифровые принципы современной жизни. Поэтому предлагается запустить новый приоритетный проект «Цифровая школа». Он будет направлен на формирование у школьников навыков в цифровом мире, обучение работе и анализу данных, элементам программирования и, самое главное, умению создавать цифровые проекты для своей будущей профессии — в технике, медицине, искусстве — в общем, в любой сфере деятельности», — заявил тогда глава правительства.

(По материалам, предоставленным пресс-службой Минобрнауки России)

Г. А. Монахова, Н. В. Монахов,
Академия социального управления, г. Москва,

Д. Н. Монахов,
Московский государственный университет имени М. В. Ломоносова

КВЕСТЫ В ОБУЧЕНИИ ИНФОРМАТИКЕ

Аннотация

В статье рассматривается роль квестов в обучении информатике. Особое внимание уделяется веб-квестам. Подробно описываются инструменты, которые помогают в создании материалов для дидактических игр в форме приключений или путешествий.

Ключевые слова: квест, дидактическая игра, электронное обучение, онлайн-сервисы.

Контактная информация

Монахова Галина Анатольевна, доктор пед. наук, профессор, профессор кафедры информационно-коммуникационных технологий Академии социального управления, г. Москва; *адрес:* 129281, г. Москва, Староватутинский проезд, д. 8; *телефон:* (495) 472-32-08, доб. 149; *e-mail:* gamonahova@yandex.ru

Монахов Никита Вадимович, канд. пед. наук, доцент кафедры информационно-коммуникационных технологий Академии социального управления, г. Москва; *адрес:* 129281, г. Москва, Староватутинский проезд, д. 8; *телефон:* (495) 472-32-08, доб. 149; *e-mail:* distantmnv@yandex.ru

Монахов Данила Никитич, канд. пед. наук, ст. преподаватель кафедры методологии социологических исследований Московского государственного университета имени М. В. Ломоносова; *адрес:* 119991, г. Москва, Ленинские горы, МГУ, д. 1. стр. 33; *e-mail:* MonahovDN@yandex.ru

G. A. Monakhova, N. V. Monakhov,
Academy of Public Administration, Moscow,
D. N. Monakhov,
Lomonosov Moscow State University

THE QUESTS IN THE TEACHING OF INFORMATICS

Abstract

The article discusses the role and place of quests in teaching informatics. Special attention is paid to web quests. The authors describe in detail the tools that help in creating materials for didactic games in the form of adventure or travel.

Keywords: quest, didactic game, e-learning, online services.

Быстрое развитие технологий и активное внедрение их в жизнь провоцируют изменения в методической системе обучения информатике. Особенность федеральных государственных образовательных стандартов общего образования — их деятельностный характер. ФГОС основного общего образования [8, п. 5] и ФГОС среднего общего образования [9, п. 4] указывают на конкретные виды деятельности обучающегося, которыми он должен овладеть в процессе обучения.

Одними из направлений системно-деятельностного подхода, закрепленными в ФГОС, являются *развитие активной учебно-познавательной деятельности* обучающихся и *ориентация на результаты* образования, которые сформулированы в виде личностных, метапредметных и предметных результатов [8, п. 8, 10; 9, п. 8].

При этом практически все учителя сталкиваются со снижением интереса обучающихся к преподаваемой ими учебной дисциплине. Преодолеть такое противоречие — между «надо» (согласно ФГОС) и «есть» (согласно практике педагогической деятельности и реального образовательного процесса в школе) — возможно с использованием игровых технологий в процессе обучения всем школьным дисциплинам, в том числе информатике. «Игра — это искра, зажигающая огонек пытливости и любознательности», — писал В. А. Сухомлинский [7].

Хорошо известно, что игра побуждает обучающихся к учебной деятельности. При этом она помогает в формировании следующих универсальных учебных действий:

- **коммуникативных:** умений слышать, слушать, понимать партнера, выполнять согласованно совместные действия, распределять роли, взаимно контролировать действия друг друга, договариваться, правильно выражать свои мысли;
- **познавательных:** умений сравнивать, искать нестандартные решения, находить закономерности;
- **личностных:** умений фантазировать, проявлять интерес к окружающему миру;
- **регулятивных:** умений планировать, оценивать правильность выполнения действий.

Традиционно учителя информатики используют интерактивные игры, созданные в Microsoft PowerPoint. Примером может служить библиотека таких авторских разработок на сайте Pedsovet.su (<http://pedsovet.su/load/1184>).

Существует много сервисов интернета, которые помогают создавать учителям дидактические игры к учебным занятиям: генераторы ребусов (<http://rebus1.com/>) и кроссвордов (<https://childdevelop.ru/generator/>), ClassTools.net (<http://www.classtools.net>), LearningApps (<http://learningapps.org>), Study Stack (<http://www.studystack.com>), H5p (<https://h5p.org>), Educaplay (<https://www.educaplay.com/>) и др.

Все перечисленные онлайн-инструменты предназначены для создания интерактивных flash-ресурсов и учебно-методических пособий. Данные сервисы основаны на работе с шаблонами. Тематика их разнообразна: от работы с карточками до разгадывания кроссвордов и составления пазлов. Все они поддерживают кириллицу.

Дидактические игры, созданные с помощью этих сервисов, требуют от обучающихся решения разнообразных задач и головоломок. Но если совокупность разнообразных игр объединить единым сюжетом, то мы получим **квест**. Отличие квеста от дидактической игры заключается в заданиях проблемного характера и «погружении» в информационное пространство сети Интернет.

Под **квестом** обычно понимают разновидность развивающей игры — приключение (или путешествие). В основе квеста, как правило, лежит

сюжет фантастического или сказочного характера. «Для кого-то квест — это игра, граничащая с реальностью. Для кого-то — реальность, граничащая с игрой. Для кого-то — это образ и стиль жизни. Для кого-то — ощущение свободы и адреналина. Для кого-то — детектив и приключение. И несмотря на то, что на сегодняшний день в мире нет единственного и универсального значения квеста, квест — это все, вместе взятое» [6].

Под **квестом в компьютерных играх** принято понимать некий путь главного героя, где ему необходимо последовательно преодолевать все трудности и препятствия, которые выражаются в виде выполнения заданий различной сложности. Как правило, тут приходится изучать сложный игровой мир и решать непростые головоломки [10]. Примером такого образовательного квеста является «Информатикус» серии «Обучение с приключением» (издатель — «МедиаХауз»). Данная приключенческая игра содержит более 39 головоломок и 200 статей по информатике. Ее можно применять при изучении процессов передачи и хранения информации, двоичной системы счисления, графов, криптографии, основ программирования.

Обычно на учебных занятиях по информатике преподаватели используют веб-квесты, посвященные какой-либо теме учебной дисциплины. Примерами могут быть «Все о сайте» [3], «В мире кодов» [2], «История вычислительной техники» [4]. Роль такого рода веб-квестов, которые проводятся на одном или двух учебных занятиях по информатике, — углубление, расширение или интеграция знаний, получаемых из различных информационных источников, формирование или развитие компетенций.

«**Образовательный веб-квест** — это сайт в интернете, с которым работают учащиеся, выполняя ту или иную учебную задачу. Разрабатываются такие веб-квесты для максимальной интеграции интернета в различные учебные предметы на разных уровнях обучения в учебном процессе» [1].

В образовательном процессе по информатике можно использовать интеграцию квеста в реальной учебной обстановке и веб-квеста с использованием современных гаджетов и/или компьютера.

В такой интеграции помогут QR-коды, не только размещенные в компьютерном классе, но и расположенные на сайте. Примером может служить веб-квест «Неделя

информатики в Корифее» [5]. QR-коды могут содержать задания с гиперссылками на сайты с информацией по изучаемой теме или викторину, созданную с использованием сервиса Kahoot (<https://kahoot.com/>).

Сервис Kahoot ориентирован на разработку онлайн-викторин, тестов и опросов с возможностью включения в них фотографий и видеофрагментов. Обучающиеся могут отвечать на созданные учителем тесты с планшетов, смартфонов, т. е. с любого устройства, имеющего доступ к интернету. Учитель информатики может регулировать темп выполнения викторин путем введения временного предела для каждого вопроса. При желании педагог может ввести баллы за ответы на поставленные вопросы: за правильные ответы и за скорость. Для викторины учащиеся, просто считывая код с помощью собственного гаджета, открывают сервис и вводят PIN-код, который был зашифрован в виде ребуса или QR-кода. Это необычное задание, которое позволит в оригинальной форме изучить ту или иную тему по информатике.

Для создания такого рода квестов можно использовать генератор QR-кодов сервиса ClassTools.net. QR-коды можно использовать как в линейных, так и штурмовых образовательных квестах по информатике, в которых обучающиеся получают общее задание, но самостоятельно и произвольно продвигаются по маршруту от разных точек.

Объединить набор дидактических игр в квест можно с использованием лент (осей, линий) времени — Timeline. Это инструменты для создания учебного материала в хронологической последовательности. Они дают возможность визуализировать подачу учебного материала по информатике и превратить его в увлекательную историю на учебном занятии.

Передвигаясь по событиям ленты времени, школьники либо получают порцию материала по информатике, либо переходят по ссылкам на игровые задания, разработанные учителем информатики с помощью описанных ранее сервисов. Важно иллюстрировать прохождение по ленте сюжетными картинками, аудио- или видеофрагментами.

Интерес представляет MyHistro (<http://www.myhistro.com/explore>) — сервис для сторителлинга (рассказывания историй) с привязкой к географическим картам Google Maps. Его удобно использовать при изучении вопросов, связанных с развитием средств вычислитель-



Рис. Варианты шаблонов для создания квеста (<https://www.genial.ly/>)

ной техники, компьютеров, систем счисления. Сервис позволяет учителям информатики строить насыщенные мультимедийным содержанием ленты времени.

Другим инструментом, который дает возможность интегрировать дидактические игры и задания в квест, является Genial.ly (<https://www.genial.ly/>). Набор интерактивных шаблонов позволит совершить путешествие на воздушном шаре, найти пиратские сокровища или совершить космическое путешествие (см. рис.).

Данный ресурс позволяет в любую интерактивную область на рисунке включить текст, картинки, видео-, аудиоматериал или гиперссылку на внешний ресурс. Причем интерактивность может быть задана как в явном (значками, кнопками, указателями), так и в неявном виде. Эта неопределенность создает дополнительные моменты волнения при поиске такого рода интерактивных зон. Данный онлайн-инструмент можно использовать для реализации линейных образовательных квестов по информатике, в которых обучающиеся выполняют задания, продвигаясь по указанному маршруту.

Образовательный квест обладает большими возможностями: позволяет разнообразить процесс обучения информатике, сделать его увлекательным и запоминающимся. Он стимулирует познавательную деятельность школьников на занятиях по информатике, повышает интерес к учебной дисциплине, предоставляет комфортные условия для раскрытия личностного потенциала обучающихся, создавая ситуацию успеха на занятиях по информатике.

Квест является формой проведения занятия, нестандартным средством предложения учебных заданий, элементом в системе контроля за усвоением материала по информатике. Он помогает учителю в реализации требований ФГОС по информатике, так как направлен на персонализацию образовательной деятельности и развитие поисковой активности обучающихся.

Список использованных источников

1. Быховский Я. С. Образовательные веб-квесты. <http://ito.edu.ru/1999/III/1/30015.html>
2. В мире кодов. <http://cod66.blogspot.ru/>
3. Все о сайте. <https://sites.google.com/site/vebkvestvseosajte/>
4. История вычислительной техники. <http://istoriya-vt.narod.ru>
5. Неделя информатики в Копифее. <https://sites.google.com/site/nedinform2013/qv-kvest>
6. О квесте. <http://funquest.com.ua/o-kveste.html>
7. Сухомлинский В. А. Сердце отдаю детям. Киев: Радянська школа, 1974.
8. Федеральный государственный образовательный стандарт основного общего образования (в ред. Приказа Минобрнауки России от 29.12.2014 № 1644). http://минобрнауки.рф/документы/938/файл/749/приказ_Об_утверждении_1897.pdf
9. Федеральный государственный образовательный стандарт среднего общего образования (в ред. Приказа Минобрнауки России от 29.12.2014 № 1645). http://минобрнауки.рф/документы/2365/файл/736/12.05.17-Приказ_413.pdf
10. Что такое квест. <http://topkin.ru/voprosy/hobbi/chto-takoe-kvest/>

НОВОСТИ

Microsoft и HP меняют образовательную сферу с помощью технологий виртуальной реальности

24 мая 2018 года в ЦИТ «Умный город» Москвы прошла совместная практико-ориентированная образовательная конференция Microsoft и HP по применению современных технологий в учебном процессе. Мероприятие проводилось при поддержке городского проекта «Школа Новых Технологий».

В рамках конференции педагоги, методисты, ИТ-специалисты школ и другие представители сферы образования совместно с организаторами рассмотрели актуальные вопросы применения новейших информационных технологий в учебном процессе, в том числе возможности смешанной и виртуальной реальности. В частности, на мероприятии была представлена гарнитура HP Windows Mixed Reality Headset. Участники самостоятельно выполнили простую и быструю настройку устройства, после чего смогли отправиться в виртуальные экскурсии в другие страны, побывать в прошедших эпохах и познакомиться с обитателями морских глубин.

«Последние обновления Windows 10 и представленные на российском рынке гарнитуры Windows Mixed Reality значительно упростили возможность использования виртуальных миров. Мы придаем большое значение применению технологии Mixed Reality в области образования. Возможности виртуальной реальности со всеми шестью степенями свободы создают полный эффект погружения, позволяя совершенно по-новому подойти к подаче материала, сделать процесс обучения более наглядным и сильнее вовлечь учеников», — комментирует Марат Хайрулин, менеджер по партнерским коммуникациям Microsoft Россия.

Разработка специализированных продуктов и решений для образовательной сферы является важнейшим направлением деятельности компаний HP и Microsoft, ведь доступное и качественное образование открывает перед людьми широкие социальные и экономические перспективы. Продукты, разрабатываемые международными ИТ-лидерами, уже много лет помогают учителям применять персональный подход в обучении и вдохновляют учеников на изучение нового. Поддержке преподавателей организаторы уделяют особое внимание. HP и Microsoft не только предоставляют учителям и руководителям учебных заведений самые современные технологические решения, но и проводят практические мастер-классы, в рамках которых специалисты демонстрируют, как пользоваться специализированной техникой и программным обеспечением для модернизации учебного процесса.

«Для компании HP особая честь принимать участие в организации мероприятия городского проекта «Школа Новых Технологий», ведь инициативы по повышению уровня образования являются важнейшими инвестициями в повышение качества жизни — для всех и повсюду», — поделился своим видением Константин Кимельман, региональный директор по стратегическому планированию и развитию подразделения персональных систем и устройств печати компании HP.

По итогам конференции все участники получили сертификаты об обучении на мастер-классах HP и Microsoft. Все участники выразили согласие, что проведение мероприятий подобного формата должно стать ежегодной традицией.

(По материалам, предоставленным компанией Microsoft)

В. И. Филиппов, В. Б. Лабути́н,
Академия социального управления, г. Москва

РАЗВИТИЕ ИНЖЕНЕРНОЙ КУЛЬТУРЫ ШКОЛЬНИКОВ

Аннотация

В статье определены составляющие инженерной культуры, которые могут формироваться при обучении в основной школе, а также обобщен опыт использования робототехнического оборудования Moway в образовательных организациях Московской области.

Ключевые слова: инженерная культура, внеурочная деятельность, информатика, технология, образовательная робототехника.

Контактная информация

Филиппов Владимир Ильич, ст. преподаватель кафедры информационно-коммуникационных технологий Академии социального управления, г. Москва; *адрес:* 129281, г. Москва, Староватутинский проезд, д. 8; *телефон:* (495) 472-32-08, доб. 149; *e-mail:* vf95@rambler.ru

Лабути́н Василий Борисович, канд. пед. наук, доцент кафедры информационно-коммуникационных технологий Академии социального управления, г. Москва; *адрес:* 129281, г. Москва, Староватутинский проезд, д. 8; *телефон:* (499) 498-00-73; *e-mail:* labutin_vb@asou-mo.ru

V. I. Filippov, V. B. Labutin,
Academy of Public Administration, Moscow

DEVELOPMENT OF ENGINEERING CULTURE OF SCHOOLCHILDREN

Abstract

The article defines the components of the engineering culture that can be formed in school, and summarizes the experience of the use of robotic equipment Moway in educational organizations of the Moscow region.

Keywords: engineering culture, extracurricular activities, informatics, technology, educational robotics.

Современное мировое сообщество столкнулось с проблемами, вызванными ускорением социокультурных процессов, многократным ростом многообразия искусственного мира человека, интенсификацией социокультурных коммуникаций. Все это сопровождается социальной напряженностью и нестабильностью, усугубляется экологическими и гуманитарными проблемами, а нередко и катастрофами. Как кризисное оценивается состояние таких областей, как техника и технология, инженерные отрасли науки и производства. По всему миру наблюдаются рост дефицита инженерно-технических кадров, кризисные явления в инженерно-техническом образовании на всех уровнях.

В качестве ответа таким вызовам времени в России предпринимаются шаги по противодействию кризисным явлениям в области инженерного производства и науки и их предотвращению. Один из таких шагов — система мероприятий, направленных на преодоление кризиса инженерного образования. Эти мероприятия, проходящие на различных уровнях — от федерального и международного до муниципального, охватывают несколько крупных задач:

- модернизация высшего и среднего профессионального образования;
- модернизация содержания основного общего образования;
- создание условий для организации внеурочной деятельности.

Основными целями изменений можно назвать повышение общей культуры производства, создание условий для развития высокоточного и наукоемкого производства, повышение обороноспособности страны в условиях глобальной нестабильности, создание условий для развития цифровой экономики.

Остановимся подробнее на конкретных результатах, достижимых на уровне школьного образования. Именно в этом возрасте еще не поздно говорить о развитии фундаментальных качеств личности, вместе с тем можно вести речь о профессиональном самоопределении, профессиональных пробах и т. п. Именно в школе можно эффективно развивать **компетентности, важные как для будущего инженера, так и для профессионала в другой отрасли, востребованной в цифровом обществе.**

Если предельно сузить перечень компетентностей до наиболее важных для инженера, то среди таковых можно назвать проектную, конструкторскую, информационно-коммуникационную и графическую.

Проектная компетентность представляет собой совокупность умений, связанных с организацией проектной деятельности, владение методами и технологиями управления проектами.

Конструкторская компетентность связана с разработкой конструкции (прототипа или модели) технической системы, которая впоследствии будет реализована в производстве.

Информационно-коммуникационная компетентность включает в себя умения, связанные с обработкой, передачей и хранением информации, приемами применения в трудовой деятельности информационно-коммуникационных технологий.

Графическая компетентность отвечает за способность изложить свою мысль в виде эскиза, схемы, чертежа и т. п. Этот вид представления информации можно назвать классикой инженерной документации.

Современный инженер сталкивается с необходимостью эффективно обрабатывать, передавать, сохранять значительные объемы информации, проектировать, создавать информационные системы и управлять ими, встраивать информационно-коммуникационные технологии в производственный процесс. Тенденции развития науки и техники позволяют прогнозировать дальнейший рост значения информационно-коммуникационных технологий для проектирования, моделирования, конструирования и управления производством [2].

Обозначенные выше компетенции оптимально развивать, воздействуя на обучающегося комплексно: на уроке и в ходе внеурочной деятельности с использованием робототехнического оборудования.

В данной статье обобщен опыт применения робототехнического оборудования фирмы Moway в образовательных организациях Московской области. **Роботы Moway** можно использовать на уроках технологии, информатики и при организации внеурочной деятельности с учащимися V—VII классов, а также в профориентационных курсах по выбору в VIII—IX классах.

Роботы, выпускаемые фирмой Moway, относятся к готовым устройствам. Это инструмент, разработанный специально для обучения и исследований. Его цель заключается в том, чтобы обеспечить учебные центры возможностью изучения мира автономной робототехники. Робот Moway снабжен датчиками, которые помогают ему функционировать в окружающей среде. Он также оборудован двумя двигателями, которые позволяют ему передвигаться по поверхности. Все периферические устройства подключены к микроконтроллеру, который отвечает за управление роботом.

Робот может быть также оснащен рядом дополнительных опций через шину аппаратного расширения. Например, он может быть подключен к модулю беспроводной связи, видеокамере, опытному образцу электронной платы или любому другому устройству. Moway является инструментом, позволяющим понять принципы робототехники.

Робот взаимодействует с внешним миром благодаря различным датчикам и индикаторам, соединенным с микропроцессором Moway. Это следующие устройства:

- датчики: линии, препятствий, световой, температуры;
- разъем расширения;
- светодиоды;
- динамик;
- микрофон;
- акселерометр;
- индикатор батареи питания.

Программирование робота осуществляется с использованием блок-схем через программу, созданную

в MowayWorld или Scratch. После создания программы она в скомпилированном виде передается в устройство.

К основным преимуществам использования Moway в учебном процессе следует отнести то, что он не требует сборки, имеет небольшой размер, необходимое и достаточное количество датчиков для решения алгоритмических задач, а также напрямую подключается по USB к компьютеру. Кроме того, используемое программное обеспечение бесплатно и имеет интуитивно понятный интерфейс.

Мы исходим из того, что при обучении робототехнике обучающиеся должны познакомиться с основными алгоритмами, которые способны выполнять робототехнические системы, а также реализовать эти алгоритмы на практике с использованием одного из видов робототехнического оборудования.

Для изучения в курсе робототехники предлагаются следующие алгоритмы:

- движение по линии;
- движение внутри контура;
- движение в зависимости от уровня освещения;
- движение в лабиринте;
- парковка;
- движение внутри контура и вытеснение препятствий (борьба сумо);
- движение по заранее определенной траектории.

Учитель при составлении рабочей программы разрабатывает курс, ориентируясь на запросы участников образовательного процесса и имеющееся в наличии робототехническое оборудование.

При разработке цикла занятий использовались материалы, размещенные на официальном сайте фирмы, в частности, «Справочник учителя. Упражнения по робототехнике» [1].

В таблице 1 приведена краткая характеристика занятий по программированию роботов Moway в рамках курса «Технология».

Таблица 1

Характеристика цикла занятий по робототехнике в рамках курса «Технология» с использованием роботов Moway

№ п/п	Тема занятия	Выполняемые работы	№ упражнения из «Справочника учителя»
1	Устройство робота Moway	Создание линейного алгоритма для робота	1.1–1.3
2	Датчики обнаружения линий и препятствий	Движение внутри контура. Движение до препятствия. Игра «Сбей препятствие»	2.1–2.5
3	Движение в соответствии с поставленной задачей	Движение по линии. Игра «Гонки». Игра «Защитник/Боец»	4.1–4.3, 4.9
4	Движение в соответствии с поставленной задачей	Движение в лабиринте	4.10
5	Использование переменных при программировании роботов	Использование светового датчика. Использование датчика препятствия	5.1–5.2*
6	Использование переменных при программировании роботов	Акселерометры	6.1
7	Использование радиочастотных модулей	Копирование движения. Удаленная станция обработки данных	7.1–7.3, 11.1–11.3

* Для данного задания рекомендуются также следующие упражнения:

1. Робот двигается в зависимости от освещенности. Если освещенность ниже 40 %, то он останавливается.
2. Робот двигается с переменной скоростью. Чем ближе препятствие, тем ниже скорость. Если до препятствия пройдено больше 90 % расстояния, то робот останавливается.

Таблица 2

Соответствие заданий, связанных с программированием роботов Moway, и понятий теоретического курса информатики

№ п/п	Задание	Умения, связанные с практической реализацией проекта	Основные теоретические понятия
1	Робот движется прямо на расстояние 15 см, поворачивается в течение 10 с, едет назад 10 с	Размещение блоков действий; настройка свойств для блоков; соединение блоков коннекторами	Алгоритм: исполнитель алгоритма; система команд исполнителя; графическая форма записи алгоритма; линейный алгоритм
2	Робот движется внутри черного контура	Размещение блоков датчиков линий; настройка свойств для датчиков	Ветвление; циклический алгоритм
3	Робот движется по черной линии		
4	Робот движется до препятствия и объезжает его	Размещение блоков датчиков линий; настройка свойств для датчиков	Ветвление; циклический алгоритм
5	Робот движется внутри черного контура и выталкивает препятствие	Размещение блоков датчиков линий и препятствия; настройка свойств для датчиков	Ветвление; циклический алгоритм
6	Робот движется в зависимости от степени освещенности. Если освещение менее 40 %, робот останавливается	Операции с данными: установка значений переменных, сравнение значений	Ветвление; циклический алгоритм; переменная; операции с переменными: объявление, присваивание значений, анализ значения
7	Робот движется до препятствия. Если препятствие близко, робот останавливается		
8	Робот движется до препятствия впереди. Затем едет назад до удара и останавливается		

В таблице 2 представлены понятия теоретического курса, которые могут быть рассмотрены при выполнении отдельных заданий по программированию роботов Moway.

Необходимо отметить, что разработка программ для роботов позволяет пройти с учащимися все этапы решения задач на компьютере, создав при этом информационные модели основных видов:

- 1) создание описательной модели;
- 2) создание формальной модели (разработка блок-схемы);
- 3) разработка программы в среде MowayWorld;
- 4) тестирование программы;
- 5) корректировка программы при необходимости.

Рекомендуется следующая структура занятия:

1. Знакомство с основными элементами конструкции роботов, их характеристиками.
2. Совместное проектирование алгоритма движения робота.
3. Реализация учащимися программы.
4. Тестирование программ.
5. Игровая пауза.

Рекомендуемое время для первого и второго занятия — по 40–45 минут. Для остальных занятий — от 60 до 80 минут.

Занятия могут быть расширены за счет использования дополнительных материалов, размещенных на сайте: <http://moway-robot.com>, а также за счет собственных разработок учителя.

В 2014/2015 учебном году в ходе преподавания курса технологии в седьмых классах МБОУ «Гимназия № 11» г. о. Железнодорожный были проведены уроки с использованием сред программирования Blockly, Scratch и роботов Moway. Учащиеся в рамках изучения раздела

«Алгоритм как модель деятельности исполнителя» на практических занятиях создали две игры в среде Blockly, а также игру в среде Scratch по предложенному сценарию и разрабатывали программы для роботов в соответствии с описанными выше алгоритмами. Анализ проведенных занятий показал высокую заинтересованность учащихся в изучении данной темы.

Данный цикл занятий лег в основу модуля «Реализация типовых алгоритмов роботами» курса внеурочной деятельности «Робототехника шаг за шагом» для учащихся V—VII классов, рассчитанного на 34 учебных часа.

Целью модуля является изучение учащимися основных алгоритмов, которые могут быть реализованы робототехническими системами.

Задачи модуля:

- формирование представлений о спектре задач, которые могут быть решены робототехническими системами;
- развитие у обучающихся навыков алгоритмического мышления;
- развитие умения творчески подходить к решению задачи.

Содержание и учебно-тематическое планирование модуля «Реализация типовых алгоритмов роботами» приведены в таблице 3.

В рамках деятельности академической площадки «Формирование основ инженерной культуры у обучающихся V—VIII классов на уроках и в процессе внеурочной деятельности с использованием робототехнического оборудования и сред программирования» кафедры информационно-коммуникационных технологий Академии социального управления на базе МБОУ «Лицей

Таблица 3

Содержание и учебно-тематическое планирование модуля «Реализация типовых алгоритмов роботами»

№ п/п	Тема занятия	Кол-во часов	Примерное содержание занятия
1	Понятие типовых алгоритмов для роботов. Датчики и их виды	2	Алгоритм для робота. Датчики, их виды, особенности использования
2	Реализация линейного алгоритма роботом	2	Разбор и реализация линейного алгоритма движения робота. Сборка и программирование робота
3	Движение в зависимости от уровня освещения	3	Использование переменных при программировании роботов. Использование светового датчика. Сборка и программирование робота
4	Датчики обнаружения линий и препятствий	5	Разбор и реализация алгоритма движения внутри контура и движения до препятствия. Сборка и программирование робота. Подготовка описания игры-проекта «Сбей препятствие». Защита проекта
5	Движение по линии	5	Разбор и реализация алгоритма движения по линии. Сборка и программирование робота. Подготовка описания игры-проекта «Гонки». Защита проекта
6	Движение в лабиринте	4	Разбор и реализация алгоритма движения в лабиринте. Сборка и программирование робота. Подготовка описания игры-проекта «Найди выход». Защита проекта
7	Парковка	3	Разбор и реализация алгоритма парковки. Использование датчика препятствия Акселерометры. Сборка и программирование робота
8	Движение внутри контура	5	Разбор и реализация алгоритма движения внутри контура. Вытеснение препятствий (борьба сумо). Сборка и программирование робота. Подготовка описания игры-проекта «Защитник/Боец». Защита проекта
9	Движение по заранее определенной траектории	5	Разбор и реализация алгоритма движения по заданной траектории. Сборка и программирование робота. Подготовка описания проекта «Робот-художник». Защита проекта
ИТОГО:		34	

№ 4 г. Дмитрова» учителем информатики О. В. Щецовой был разработан и прошел успешную апробацию цикл занятий с использованием роботов Moway в рамках курса информатики, а также в рамках курса по выбору для учащихся IX классов.

Роботы Moway можно использовать для реализации творческих и проектных заданий, например, таких [1]:

- маяк: роботы двигаются к источнику света;
- слалом: робот объезжает препятствия, расположенные на указанном расстоянии;
- кошка и мышка: робот двигается в сторону препятствия;
- робот-художник: робот рисует контур в соответствии с заданием.

Использование готовых роботов, не требующих сборки, обеспечивает:

- развитие алгоритмического мышления обучающихся;

- практические знания о роботах, их составных частях, эксплуатации и управлении ими;
- подготовку по теории и практике компьютерного управления;
- наглядную демонстрацию учащимся того, как действует формальный исполнитель;
- формирование представления об основных алгоритмических конструкциях, используемых в программировании.

Список использованных источников

1. Справочник учителя. Упражнения по робототехнике. Образовательная теория робототехники и практическое руководство. <https://ru.scribd.com/document/335693266/moway-spravochnik-prepodavatelya>

2. Шутикова М. И., Дзамыхов А. Х., Лабутин В. Б. Факторы, влияющие на состав и структуру понятия «инженерно-технологическая культура» // Мир науки, культуры, образования. 2016. № 3.

В. В. Кащей,

Академия социального управления, г. Москва

РЕЛЕВАНТНОСТЬ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ В КУРСЕ ИНФОРМАТИКИ ОБЩЕОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЙ. ВЫБОР ЯЗЫКА ПРОГРАММИРОВАНИЯ

Аннотация

В статье анализируется роль алгоритмизации и программирования в курсе информатики. Рассматриваются критерии и предлагаются рекомендации по выбору языка программирования при изучении информатики.

Ключевые слова: школьная информатика, базовое образование по информатике, программирование.

Контактная информация

Кащей Владимир Васильевич, канд. пед. наук, доцент кафедры информационно-коммуникационных технологий Академии социального управления, г. Москва; *адрес:* 129281, г. Москва, Староватутинский проезд, д. 8; *телефон:* (495) 472-32-08, доб. 149; *e-mail:* www54@mail.ru

V. V. Kaschei,

Academy of Public Administration, Moscow

THE RELEVANCE OF ALGORITHMIZATION AND PROGRAMMING IN THE INFORMATICS COURSE OF SECONDARY EDUCATIONAL ESTABLISHMENTS. THE CHOICE OF PROGRAMMING LANGUAGE

Abstract

The article analyzes the role of algorithmization and programming in the course of informatics. The criteria and recommendations for the choice of programming language in the study of informatics are considered.

Keywords: school informatics, basic informatics education, programming.

Необходимость и глубина изучения алгоритмизации и программирования в курсе информатики обсуждаются с момента введения данного курса в школьную программу. При этом высказываются самые различные мнения: от безусловной необходимости их изучения до полного исключения программирования из курса информатики. Вторая точка зрения обосновывается тем, что сегодня при использовании цифровых ресурсов нет необходимости уметь программировать в традиционном понимании этого термина. Наличие удобного интерфейса позволяет управлять ресурсами, не создавая программ. Попробуем выяснить, что дает изучение алгоритмизации и программирования.

Когда в школьный курс вводился новый предмет, авторы представляли его как «вторую грамотность». Введение в школьный курс дисциплины «Программирование» имело большое значение для обучения алгоритмическому образу мышления. Так же как изобретение языка и письменности обеспечило переход на качественно новый уровень развития человеческого мышления, овладение «второй грамотностью» развивает алгоритмическое мышление и вместе с ним абстрактное, логическое, критическое, т. е. научное мышление.

Первоначально предмет «Информатика» назывался «Основы программирования и вычислительной техники» и задумывался как обучение программированию. Целью было сделать программирование таким же понятным для всех, как и школьная математика. При этом предполагалось, что в будущем без знания основ программирования общаться с компьютером будет невозможно.

Программирование в то время было отраслью, в которой наиболее востребовано умение мыслить, творить. Здесь создавались новые знания, которые выражались в создании новых алгоритмов и кодировке их в программы. Проверенные алгоритмы записывались либо с помощью схем алгоритмов, либо на алгоритмическом языке высокого уровня типа Алгол. Сборники отобранных и проверенных алгоритмов периодически публиковались [1].

Разработанные алгоритмы реализуются в программах, которые не требуют материальной реализации, поэтому скорость реализации алгоритма в основном определяется только скоростью работы разработчика. Разработка алгоритма начинается с анализа условий задачи, выбора модели и метода численного решения. Все это обеспечивает выработку навыков аналитического, абстрактного, логического и алгоритмического мышления. В итоге обеспечивается выработка навыков критического мышления, которое является высшей формой мышления и относится к научному виду мышления. Программирование в принципе невозможно без творчества, поэтому научить программировать означает то же самое, что научить творчески мыслить. Выработка и развитие научного стиля мышления проявились в программировании в наиболее «чистом» виде.

Федеральные государственные образовательные стандарты определяют результаты изучения информатики. Согласно ФГОС среднего общего образования [6], изучение предметной области «Математика и информатика» должно обеспечить в том числе сформированность основ логического, алгоритмического и математического мышления; сформированность умений применять полученные знания при решении различных задач.

Есть две точки зрения на определение понятия «алгоритмическое мышление». Первая определяет алгоритмическое мышление как систему

навыков, ориентированную на заранее установленные правила, общепринятую последовательность действий, необходимых для решения типовых задач. Вторая рассматривает алгоритмическое мышление более широко. Рассмотрим более подробно эту вторую точку зрения.

Становление человеческого мышления происходило в процессе решения жизненно важных для существования человека задач. При решении некоей задачи необходимо было проанализировать сложившуюся ситуацию и либо подобрать уже существующий и опробованный алгоритм действий, либо создать новый на базе существующих путем их преобразования для использования в изменившейся ситуации.

Чем больше набор алгоритмов и чем лучше они систематизированы, тем лучше происходит адаптация объекта в системе. Необходимость анализа ситуации породила абстрактное мышление (выработка понятий), классификацию и систематизацию алгоритмов (выделение базовых конструкций) и синтез нового алгоритма на основе имеющихся абстрактных понятий и набора уже использованных алгоритмов. В этом смысле можно определять алгоритмическое мышление как систему мыслительных приемов, направленных на решение задач. Можно выделить две стадии разработки алгоритма: поиск и выделение уже известного (ранее опробованного) алгоритма и создание нового. При таком определении алгоритмического мышления мы не ограничиваемся разработкой, записью и кодированием алгоритма в программу, а рассматриваем весь процесс решения задачи, начиная с постановки задачи (анализа условий) и заканчивая получением результата и дальнейшим использованием разработанного алгоритма и программного продукта с учетом возможностей его дальнейшей модификации.

Изучение программирования способствует решению одной из главных задач образования — развитию научного (критического) мышления, без которого невозможно объективное понимание картины окружающего мира. Именно поэтому обучение алгоритмизации и программированию наиболее способствует формированию навыков такого мышления. Из этого следует, что алгоритмизация и программирование остаются одним из главных разделов информатики, и если мы хотим обучить и воспитать полноценного гражданина информационного общества, нам необходимо обеспечить их изучение.

Характерно, что представители коммерческих структур, заинтересованные в получении готовых специалистов, тоже выступают за обучение информатике в школе с большей теоретической составляющей и формированием алгоритмического мышления. Выступая на Десятой открытой Всероссийской конференции «Преподавание информационных технологий в Российской Федерации», Б. Г. Нуралиев, директор фирмы «1С», отметил: «Сегодня очевидно, что обучение современным технологиям нужно начинать не в вузе, а гораздо раньше. Необходимо правильное позиционирование школьной дисциплины “Информатика”, преподавание которой должно быть нацелено на освоение, прежде всего, ее фундаментальной составляющей, на то, чтобы привить учащимся необходимое современному человеку алгоритмическое мышление (полезное, независимо от будущей специальности), на сохранение преемственности между курсами информатики для разных ступеней образования» [5].

Президент и генеральный директор компании АБВУУ С. Г. Андреев, выступая на той же конференции, сказал: «Только человек, обладающий алгоритмическим мышлением, способен создавать что-либо новое, оригинальное, уникальное. Те же, кто имеет навыки пользователя, пусть и уверенного, могут лишь потреблять уже созданное кем-то. Поэтому, если мы действительно хотим выйти в мировые лидеры в сфере ИТ, нужно весь процесс обучения со школьной скамьи выстраивать с соответствующих позиций» [5].

При обучении программированию и выработке алгоритмического мышления возникает вопрос выбора языка высокого уровня, на примере которого проводится обучение. Этот вопрос постоянно обсуждается сообществом учителей информатики и заинтересованными лицами.

Определим требования, которым должен удовлетворять этот язык:

- Соответствие языка парадигме структурного программирования.
- Простота и компактность синтаксиса. Синтаксис должен быть «очевиден»: зная логику построения языка, как правило, можно «угадать» синтаксис операторов и описаний. Компактность синтаксиса обеспечивает простоту изучения языка.
- Строгость языка. Все объекты должны быть описаны и «приписаны» к соответствующим типам. Преобразования типов могут выполняться очень редко специальными функциями преобразования типов.
- Наличие интегрированной среды разработки и отладки программы с хорошей встроенной справочной системой и возможностями отладки программы.
- Высокая надежность программы за счет контроля использования типов данных и объектов программы, как на этапе трансляции программы, так и при ее выполнении.

Большое количество языков программирования, разработанных в последнее время (Java, C/C++, PHP, Python и др.), допускают более компактное и простое описание исполнения алгоритма. Однако выбор их как первого языка для обучения спорен. Их синтаксис достаточно сложен и, главное, неочевиден, есть много непонятных обучающимся конструкций. Процесс программирования облегчается за счет использования большого количества библиотек, но их использование в то же время затрудняет понимание в процессе изучения языка и методов программирования, поскольку действия, которые выполняются библиотечной программой, «невидны», и поэтому непонятно, что и, главное, как делает компьютер.

Например, на языке Python легко поменять местами значения переменных a и b . Это делается одним оператором $a, b = b, a$. Но у обучаемого не будет понимания того, как это делается на самом деле, он не узнает, что для такого обмена необходима еще одна переменная, в которой будет временно храниться промежуточное значение одной из переменных.

Еще сильнее можно запутаться при изучении структур данных в Python. Рассмотрим пример такой структуры, как списки. Списки похожи на массивы, но могут одновременно содержать данные любого типа, т. е. в одном массиве могут находиться числовые, строковые данные и данные других типов. Попробуйте это объ-

яснить обучаемым, когда большинству из них непросто понять, что такое обычный одномерный массив.

Python, конечно, мощный язык, но для начального изучения он все-таки слишком сложен. Аналогично обстоит дело и с некоторыми другими недавно разработанными языками программирования. Энтузиазм по выбору «нового и перспективного» языка для первичного обучения, проявляемый некоторыми учителями, вероятно, основан на том, что они успели забыть, как сложно разобраться с некоторыми базовыми понятиями программирования, когда встречаешься с ними впервые.

Представляется целесообразным, что при наличии времени необходимо дать возможность обучаемым основательно изучить основы программирования на базовом языке, включив сюда объектно-ориентированное программирование и базовые структуры данных (например, файлы, множества, записи, динамические переменные и структуры, отображаемые на их основе, — списки, таблицы, графы и т. д.). Лучше хорошо изучить один язык, чем бегло ознакомиться с несколькими. В этом случае при освоении нового языка можно по аналогии, в сравнении понять средства этого языка.

Изучив объектно-ориентированное программирование, можно перейти к созданию программ на основе визуальной объектно-ориентированной RAD* среды программирования, которая является развитием изученного перед этим языка. Например, Delphi или Lazarus для языка Pascal.

При последовательном изучении нескольких языков обучаемые начинают путать конструкции этих языков. В 1980-х годах в некоторых московских вузах студенты на протяжении двух—четырех семестров изучали два—четыре языка программирования, и у них постоянно возникали проблемы с синтаксисом и структурами данных изученных языков. Это неоднократно отмечалось преподавателями вузов, проводившими занятия по программированию.

При изучении информатики не так важно, какой именно язык учить: основа всех языков идентична, а надстройка над основой изучается позже, в профильных учебных заведениях. Достаточно разобраться в способе реализации языком базового набора понятий математической логики, условий, циклов и т. п., имеющихся

практически в любом языке программирования. Задача школы состоит в ознакомлении с основами программирования и формировании приблизительного представления о методах его реализации.

Разработка программ предполагает до начала написания программы разработку алгоритма. Именно на этом этапе должны решаться основные проблемы разработки и реализации решения поставленной задачи. Написание текста программы — это перевод алгоритма на язык программирования, который является достаточно формальной задачей. Это еще раз подтверждает мысль, что выбор языка не так уж важен, если мы правильно разработаем алгоритм. При этом при выборе формы записи алгоритма рекомендуется учесть особенности языка, на который будет переведен алгоритм. В этом случае предпочтительнее язык с простым синтаксисом, максимально приближенным к естественному английскому языку.

Подведем итоги. Изучение алгоритмизации и программирования, несмотря на кажущуюся узкой область применения программирования, помогает развивать научное мышление. Поэтому изучение алгоритмизации и программирования является одной из ключевых составляющих обучения информатике. При выборе языка для обучения программированию необходимо руководствоваться пригодностью его для обучения, в первую очередь, основам программирования.

При правильной разработке и записи алгоритма выбор языка программирования не так уж и критичен, так как хорошо разработанный и правильно записанный алгоритм не сложно перевести на любой язык программирования.

Список использованных источников

1. Библиотека алгоритмов 16-506: справочное пособие. М.: Сов. радио, 1975.
2. Бобров А. Н. Проблемы выбора языка программирования в школьном курсе информатики // Молодой ученый. 2015. № 24. <https://moluch.ru/archive/104/24471/>
3. Ворожцов А. В. Языки программирования в школе // Потенциал. 2016. № 5. https://ru.wikibooks.org/wiki/Языки_программирования_в_школе
4. Мелешко В. Алгоритмическое мышление. Без него невозможно создать что-либо новое // Учительская газета. № 22 от 31.05.2011. <http://www.ug.ru/archive/40532>
5. Преподавание информационных технологий в Российской Федерации: Материалы Десятой открытой Всероссийской конференции (16—18 мая 2012 года). М.: МГУ имени М. В. Ломоносова, 2012.
6. Федеральный государственный образовательный стандарт среднего общего образования (в ред. Приказа Минобрнауки России от 29.12.2014 № 1645). http://минобрнауки.рф/документы/2365/файл/736/12.05.17-Приказ_413.pdf

* RAD (от *англ.* rapid application development — быстрая разработка приложений) — концепция создания средств разработки программных продуктов, уделяющая особое внимание скорости и удобству программирования, созданию технологического процесса, позволяющего программисту максимально быстро создавать компьютерные программы.

М. В. Савченкова,

*Академия социального управления, г. Москва;
Лицей г. Протвино, Московская область*

SCRATCH-ХАКАТОН «ПРОГРАММИРУЕМ В СРЕДЕ SCRATCH»

Аннотация

В статье обсуждается опыт проведения Scratch-хакатона, в частности, использование формата хакатона как средства мотивации учащихся к программированию в среде Scratch. Рассматриваются развитие российского Scratch-сообщества и банка полезных проектов в среде Scratch, подготовка участников Scratch-хакатона к конкурсу по визуальному программированию на языке Scratch.

Ключевые слова: образовательный хакатон, Scratch, проекты.

Контактная информация

Савченкова Мира Викторовна, канд. тех. наук,

ст. преподаватель кафедры информационно-коммуникационных технологий Академии социального управления, г. Москва; *адрес:* 129281, г. Москва, Староватутинский проезд, д. 8; *телефон:* (495) 472-32-08, доб. 149;

учитель информатики Лицея г. Протвино, Московская область; *адрес:* 142280, Московская область, г. Протвино, ул. Школьная, д. 12; *телефон:* (496) 744-69-26; *e-mail:* mira1965@mail.ru

M. V. Savchenkova,

Academy of Public Administration, Moscow; Lyceum, Protvino, Moscow Region

SCRATCH HACKATHON "CODING IN SCRATCH"

Abstract

The article discusses the experience of the Scratch hackathon, in particular, the use of the hackathon format as a means of motivating students to program in the Scratch environment. The development of the Russian Scratch community and the bank of useful projects in the Scratch environment are considered, the training of the participants of the Scratch hackathon for the contest on visual programming in the Scratch language is described.

Keywords: educational hackathon, Scratch, project activity.

Количество проходящих в нашей стране хакатонов увеличивается с каждым днем, даже существует специальный сайт с актуальным и регулярно обновляемым списком хакатонов, которые проводятся в России: <https://8d9.ru/it-events/hackathons>

Название «хакатон» происходит от двух слов — «хакер» и «марафон», т. е. по сути **хакатон** — это программистский марафон, мероприятие по программированию, которое продолжается в течение нескольких часов. Впервые термин был использован корпорацией Sun Microsystems и разработчиками OpenBSD в 1999 году, когда обе компании совместно трудились над созданием шифровальных алгоритмов. Хакатоном был назван один из проектов по созданию программного обеспечения на языке Java [2].

Формат мероприятия оказался настолько эффективным, что проведение хакатонов стало популярным не только среди сотрудников корпораций, но и в студенческой среде. А в последнее время в это движение вливаются все больше и больше школьников.

Конечно, изменение состава участников приводит к определенной корректировке формата проведения мероприятия: если в «классическом» варианте группа разработчиков «запиралась» на 48 часов и устраивала мозговой штурм по решению некоторой проблемы, то в студенческой среде продолжительность хакатона ограничивается сутками, а хакатоны с участием школьников обычно длятся от четырех до восьми часов. Кроме того, изменилась и тематика проводимых мероприятий: теперь это марафон не только для программистов, но и для робототехников, во время которого небольшие команды (обычно два-три человека) соревнуются в разработке электронного устройства и одновременно в создании программного обеспечения для него.

Кроме того, в последнее время становятся популярными и чисто технические хакатоны. На технических хакатонах разрабатываются и тут же воплощаются в жизнь технические устройства, в которых запрограммировать просто нечего. Классический пример такого технического хакатона, когда команда получает задание создать из подручных средств воздушный шар и победитель определяется в зависимости от того, чей шар поднимет больший груз.

Но все же большая часть хакатонов по-прежнему связана с программированием. Так, на «Атомном Нейротроне», который проводился в этом году в г. Обнинске Калужской области, нужно было построить биодинамический светильник «Технолаб» и запрограммировать устройство [4]. Это был настоящий марафон технических специалистов (в нем участвовали программисты, электронщики, конструкторы, дизайнеры, биологи, медики), где перед участниками была поставлена задача: управлять силой мысли подсветкой светильника. Управление предметами силой мысли не фантастика, а одно из перспективных направлений в сфере информационных технологий, которое позволит вернуть к полноценной жизни людей с различными проблемами со здоровьем, дав им возможность как получать информацию из внешнего мира (искусственные зрение и слух), так и активно на него воздействовать (бионические протезы). Разработки таких устройств велись с 60-х годов прошлого века, а в 2014 году на чемпионате мира по футболу удар по мячу нанес человек с парализованными ногами. Протезом он управлял при помощи нейроинтерфейса [2].

Таким образом, мы видим, что хакатоны могут проводиться по различным направлениям — от медицины до проектирования технического устройства.

Если в хакатон добавить элемент соревновательности и поставить перед участниками цель — создание игры, то получится Game Jam (джем-хакатон) [3].

Существуют определенные **этапы проведения хакатона**:

- открытие хакатона;
- постановка задач, предложенных для разработки проектов;
- формирование команд;
- командная работа над проектами (основной этап);
- презентация работ;
- если хакатон носит соревновательный характер, жюри оценивает работы и определяет победителя.

Результат проведения хакатона — это разработка некоторого проектного продукта: мобильного приложения, веб-приложения, сайта, робота, технического устройства и т. д. Такой продукт в случае, когда хакатон проводят коммерческие компании, может перейти из категории «проектный продукт» в категорию «коммерческий продукт», причем «коммерческий» (как следует из истории проведения первого хакатона [2]) не обязательно означает продукт, приносящий прибыль, — это может быть и общественно значимый продукт, распространяемый на безвозмездной основе или на основе добровольных пожертвований.

Но вернемся к основной теме, которую хотелось бы обсудить и осмыслить, — **проведение хакатонов среди учащихся средних образовательных учреждений**. В этом случае цель создания инновационного продукта не ставится. Основная цель хакатона среди школьников — разбудить в участнике творца, показать ему, что он может не только читать, смотреть и повторять, но и создавать нечто свое, оригинальное.

В хакатоне заложен большой образовательный потенциал, поскольку в данном формате *творческий* момент имеет приоритет над умением использовать шаблоны. На образовательном хакатоне от школьника требуется подойти к решению поставленной задачи самостоятельно, творчески и максимально продуктивно. Хакатон должен закончиться созданием определенного продукта, который будет оцениваться судьями и будет представлен другим участникам, — все максимально открыто.

Образовательный хакатон по программированию — мероприятие, на котором предлагается набор проблемных заданий с элементами ролевой игры, для выполнения которых требуются какие-либо ресурсы. Одна из особенностей подобных хакатонов — их междисциплинарный характер, когда информационные технологии применяются для решения задач из разных школьных дисциплин и на разных уровнях обучения.

Рассмотрим пример такого **хакатона, разработанного в рамках проекта «Твой курс: ИТ для молодежи» в Лицее г. Протвино Московской области**.

В процессе подготовки хакатона **перед его организаторами были поставлены следующие задачи**:

- разработать хакатон, вовлекающий в проект учащихся и учителей не только самого лицея, но и других общеобразовательных учреждений города;
- предложить задания, способные увлечь участников и в то же время имеющие образовательный характер;
- показать возможности дальнейшей работы участников над проектами в среде Scratch, в том числе возможность участия в конкурсе команд-

ного программирования на языке Scratch «Collab Challenge», проводимом Всемирным банком и Рыбаков Фондом при поддержке АНО ЦРСИ «Проектория» и АО «РОББО»: <http://notoproject.org/cc> [1].

Площадкой, удобной для проведения хакатона и размещения на нем заданий, стал **сайт хакатона**: <https://sites.google.com/view/scratchhakaton2018/>

Цель проведения хакатона: популяризация среды креативного программирования Scratch как эффективного инструмента для развития навыков XXI века.

Задачи хакатона:

дидактические задачи:

- развитие навыков программирования в среде Scratch;
- развитие творческих способностей учащихся;

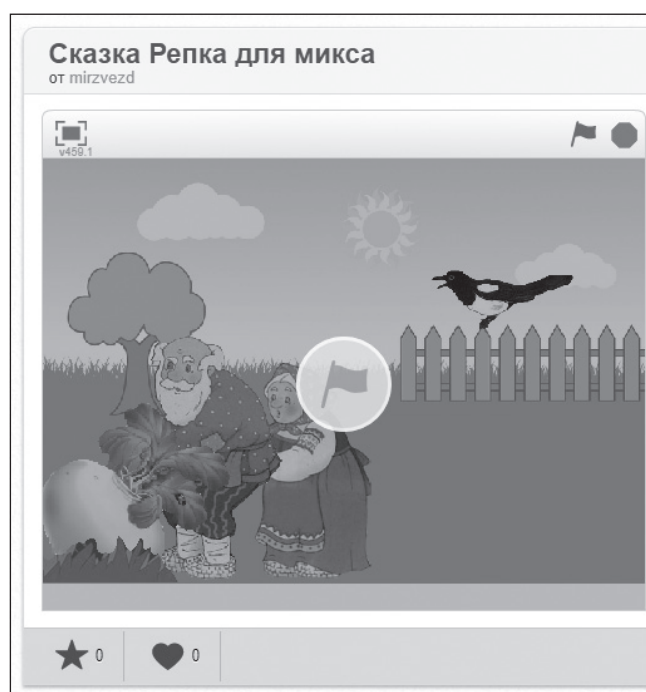
деятельностные задачи:

- развитие способности к коммуникации и самоорганизации действий;
- создание дополнительных возможностей для молодежи в изучении программирования и развития цифровых навыков;
- развитие российского сообщества и банка полезных проектов в среде Scratch;
- подготовка школьников к участию в конкурсе по визуальному программированию на языке Scratch «Collab Challenge»;
- развитие способности к осознанию полученной информации, рефлексии собственных действий.

В качестве **заданий** были предложены три файла-заготовки, которые нужно было изменять и дополнять в среде Scratch (для этого существует специальный термин — «миксовать»).

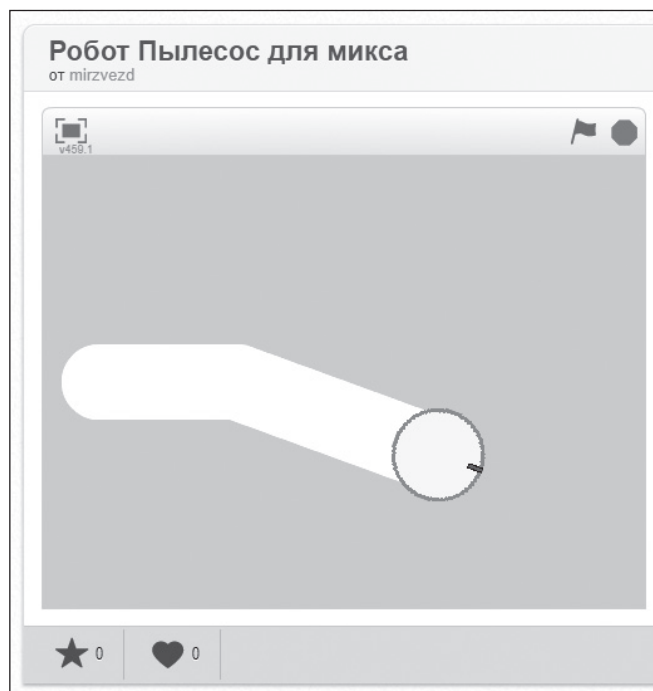
Вариант 1. Сказка ложь, да в ней намек.

<https://scratch.mit.edu/projects/212288670/>

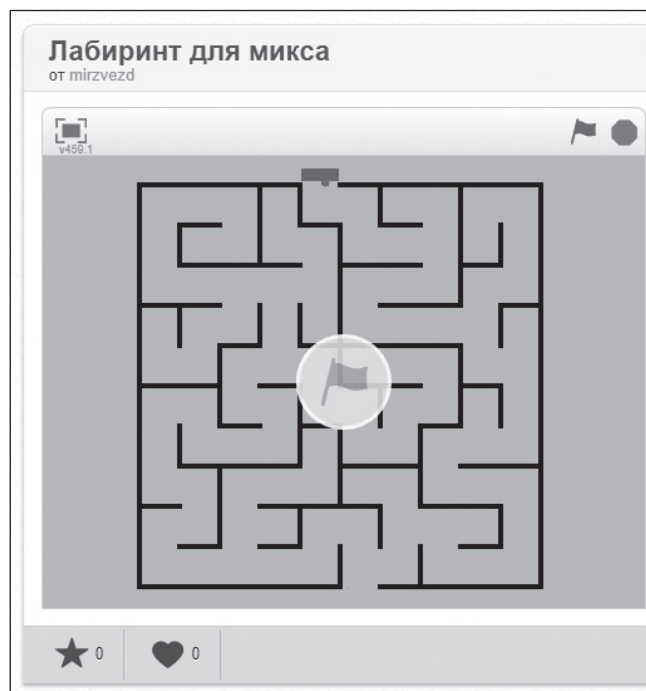


Вариант 2. Пусть всегда будет чисто.

<https://scratch.mit.edu/projects/212288472/>

**Вариант 3. Ищем выход.**

<https://scratch.mit.edu/projects/212289272/>



Файлы-заготовки для предлагаемых на выбор проектов были заранее подготовлены организаторами, и в дальнейшем участники могли либо развивать предложенные проекты, либо «снимать кино» не про Репку, а про Курочку Рябу.

Ход хакатона

№ п/п	Этап	Время, мин	Действия учителя	Действия учащихся
1	Актуализация знаний, повторение ранее изученного материала	40		
1.1	Фронтальный опрос	8	С помощью проектора проводит опрос для актуализации знаний, полученных на предыдущих занятиях по Scratch. По ситуации дополняет ответы, дает советы по использованию программных средств.	Отвечают на вопросы, настраиваясь на тему урока. Пытаются дать ответ на поставленные вопросы.
1.2	Знакомство с новыми возможностями Scratch	15	Рассказывает о возможностях Scratch, которые до работы на хакатоне, возможно, были мало востребованы участниками: «рюкзак», студия, куратор.	Слушают объяснение. Задают вопросы.
1.3	Разделение на команды	5	Оказывает консультационную помощь.	В каждой команде выбирают капитана. Занимают места за компьютерами, входят в аккаунты, созданные на сайте: https://scratch.mit.edu Капитан заводит студию (например, pro-karitoska), приглашает кураторами членов своей команды.
1.4	Объяснение правил проведения хакатона	12	Предлагает задание на выбор: 1. «Сказка ложь, да в ней намек» — интерактивный мультфильм-сказка. 2. «Пусть всегда будет чисто» — моделирование поведения робота. 3. «Ищем выход» — программируемая игра-лабиринт.	

№ п/п	Этап	Время, мин	Действия учителя	Действия учащихся
			Объясняет общий алгоритм работы команды: 1. Разделить сюжет (сделать раскадровку) на всех участников команды. 2. Предложить каждому написать свой кусок кода. 3. Публикация своей части работы каждым участником команды, участники нажимают кнопку «поделиться» (необходимо опубликовать свою работу, иначе она будет видна только автору). 4. Собираем проект целиком. Для сборки проекта воедино капитан команды заходит в студию, где кураторы уже разместили свои сделанные фрагменты общей работы, заходит в работу куратора, забирает к себе в рюкзак героя куратора, его сцену, делает сборку и отладку общей работы. 5. По окончании работы добавляют в студию руководителя хакатона в качестве куратора по его никнейму, после чего жюри оценивает работы. Знакомит учащихся с критериями оценивания работы: 1. Наличие заставок: «название», «конец» (2 балла). 2. Смена экранов; дизайн фона соответствует жанру истории, не мешает восприятию; есть смена фона (1 балл). 3. Интерактивность сказки (вопрос/ответ) (1 балл). 4. Управление персонажем с помощью клавиатуры/мышки (1 балл). 5. Персонажи общаются между собой, произносят фразы, связанные с игровыми событиями/событиями истории, или вступают в диалог с другими персонажами/объектами (1 балл). 6. В пояснении к проекту-сказке даются инструкции, которые объясняют игроку задачу, возможности управления главным героем игры и цель игры (1 балл). 7. Оригинальность сюжета: • оригинальный сюжет (2 балла); • известный сюжет (1 балл). 8. Наличие ошибок в коде персонажей программы, которые мешают просмотру сказки (0 баллов, отсутствие — 1 балл). 9. Оценка пользователей игры: было ли им интересно, комфортно играть, понравилась ли идея, сюжет игры/истории, персонажи. (Судья добавляет 1 балл.) 10. Игра носит образовательный характер, несет в себе общечеловеческие ценности и моральные нормы, поднимает важные проблемы экологии или мирного существования (1 балл). Возможно автоматическое оценивание проекта: http://www.drscratch.org/	Выбирают задание.
2	Перерыв	5		
3	Основная часть — Scratch-хакатон	65		
3.1	Формулирование заданий	5	Вариант 1. Сказка ложь, да в ней намек. Задание: • Изменить героя. • Добавить сцену, откуда приходят Баба и прочие персонажи. • Закончить сказку. Ссылка на исходный файл: https://scratch.mit.edu/projects/212288670/ Вариант 2. Пусть всегда будет чисто. Задание: • Изменить героя. • Добавить свои лабиринты.	Слушают.

№ п/п	Этап	Время, мин	Действия учителя	Действия учащихся
			• Добавить ловушки — статические или гоняющиеся за персонажем. • Добавить счетчик времени прохождения лабиринта и таблицу рекордов. Ссылка на исходный файл: https://scratch.mit.edu/projects/212288472/ Вариант 3. Ищем выход. Задание: • Изменить героя. • Добавить в робота-пылесос автоматический режим. • Научить робота максимально хорошо убирать комнату в автоматическом режиме. • Прорезать дверь и добавить вторую комнату, в которую робот переходит, закончив с первой. • Расположить во второй комнате мебель и обучить робота максимально хорошо убирать комнату с мебелью. • Научить робота различать грязь и сильную грязь. • Грязь чистится простым проходом, а сильная — проходом с вращением. Ссылка на исходный файл: https://scratch.mit.edu/projects/212289272/	
3.2	Выполнение заданий	60	Вместе с координаторами (тьюторами) оказывает консультативную помощь участникам. Проводит физкультминутки.	Непосредственная работа учащихся с программой. Обсуждают проблему, ищут пути ее решения. Планируют работу группы. Ставят частные задачи для решения общей проблемы. Каждый участник группы решает свою задачу. Работают над заданием. Каждый участник работает над своей частью задания. Сверяются с критериями. Собирают работу в единое целое. По окончании работы добавляют в студию руководителя хакатона по его никнейму (необходимо для оценивания работ).
4	Обеденный перерыв	30		
5	Подведение итогов хакатона	25		
5.1	Презентация проектов	20	Проводит общий просмотр полученных от учащихся материалов.	Просматривают получившиеся материалы. Оценивают вклад в общую работу каждого учащегося. Оценивают качество получившихся продуктов.
5.2	Подсчет баллов. Выявление победителей. Награждение	5	Подводит общий итог мероприятия. Выставляет баллы в соответствии с заранее сформулированными критериями. Награждает участников грамотами и сертификатами.	Слушают.

Список использованных источников

1. *Патаракин Е. Д.* Конкурс российских скретчеров. Презентация в рамках мастер-класса «Всемирное Scratch-братство» (15.02.2018, ВДНХ) // Letopisi.ru. <http://letopisi.org/index.php/BrScratch/Presentation>

2. *Солозобов О.* Хакатон: что это и как работает? // IT тренды от Олега Солозобова. <https://8d9.ru/xakaton-chto-eto-i-kak-rabotaet>

3. *Толкачева Е.* Хакатон для детей — игры с пользой // Newtonew — медиа о современном образовании. <https://newtonew.com/tech/digital-banana-hackathon>

4. *Юрьева З.* Сегодня в Обнинске завершился «Атомный нейротрон» // Портал НГ-РЕГИОН. <http://ngregion.ru/novosti/segodnya-v-obninske-zavershitsya-atomnyj-nejrotron>

М. Л. Казарян, М. А. Шахраманьян,
Владикавказский филиал Финансового университета
при Правительстве Российской Федерации,
Республика Северная Осетия — Алания,

А. Б. Мурнин,
Научно-исследовательский институт аэрокосмического
мониторинга «АЭРОКОСМОС», г. Москва;
Вычислительный центр им. А. А. Дородницына Федерального
исследовательского центра «Информатика и управление»
Российской академии наук, г. Москва,

А. А. Рихтер,
Научно-исследовательский институт аэрокосмического
мониторинга «АЭРОКОСМОС», г. Москва;
школа № 97, г. Москва,

Артём Давыдов, Илья Дементьев, Данила Игнатьев,
учащиеся школы № 97, г. Москва

МЕТОДИКА ПОСТРОЕНИЯ 3D-МОДЕЛЕЙ РИГИДНЫХ ОБЪЕКТОВ ПО ОДНОМУ ИЗОБРАЖЕНИЮ И ЕЕ ПРИМЕНЕНИЕ В ПРОЕКТНОЙ ДЕЯТЕЛЬНОСТИ ШКОЛЬНИКОВ*

Аннотация

Предлагается простая методика построения 3D-моделей объектов поверхности Земли по данным одного космического изображения. Методика дает оценку модели по объекту-эталоноу, для которого известно значение высоты, соответствующей заданному отрезку на изображении, дающему в пространственной окрестности объекта высотный коэффициент, т. е. масштабирующий коэффициент по оси аппликата. Высотный коэффициент меняется во времени, т. е. от снимка к снимку, что позволяет оценить модели объектов разной четкости, от низкой до высокой, используя пространственно-временную окрестность объекта-эталона. С расстоянием от эталона высотный коэффициент непрерывно меняется, что позволяет оценить высоту отдаленного объекта «перемещением» расчетов через цепочку более близких объектов, соединяющих исходный и конечный объекты. Использование альтернативного «расчета по тени» позволяет дать оценку модели при фотографировании местности в надири, т. е. при отсутствии видимости боковых частей расчетного объекта. Ввиду доступности практического применения данной методики в системе общего образования по данной методике выполнен ряд проектных работ школьников в области 3D-моделирования по данным космических изображений.

Ключевые слова: полигон ТБО, объект захоронения отходов, свалка, 3D-модель, 3D-моделирование, хоззона, хозяйственная зона, здание, ригидный объект, Sketch Up, высота, коэффициент масштаба, Google Earth, проектная работа.

Контактная информация

Казарян Маретта Леоновна, канд. физ.-мат. наук, доцент, зав. кафедрой «Математика и информатика» Владикавказского филиала Финансового университета при Правительстве Российской Федерации, Республика Северная Осетия — Алания; *адрес:* 362001, Республика Северная Осетия — Алания, г. Владикавказ, ул. Молодежная, д. 7; *телефон:* (8672) 51-96-67; *e-mail:* marettak@bk.ru

Шахраманьян Михаил Андраникович, доктор тех. наук, профессор, профессор кафедры «Математика и информатика» Владикавказского филиала Финансового университета при Правительстве Российской Федерации, Республика Северная Осетия — Алания; *адрес:* 362001, Республика Северная Осетия — Алания, г. Владикавказ, ул. Молодежная, д. 7; *телефон:* (8672) 51-96-67; *e-mail:* 7283963@mail.ru

Мурнин Александр Борисович, канд. тех. наук; ст. научный сотрудник Научно-исследовательского института аэрокосмического мониторинга «АЭРОКОСМОС»; *адрес:* 105064, г. Москва, Горюховский пер., д. 4; ст. научный сотрудник Вычислительного центра им. А. А. Дородницына Федерального исследовательского центра «Информатика и управление» Российской академии наук, г. Москва; *адрес:* 119333, г. Москва, ул. Вавилова, д. 40; *телефон:* +7 (915) 355-45-64; *e-mail:* AMurynin@bk.ru

Рихтер Андрей Александрович, инженер Научно-исследовательского института аэрокосмического мониторинга «АЭРОКОСМОС»; *адрес:* 105064, г. Москва, Горюховский пер., д. 4; педагог дополнительного образования школы № 97, г. Москва; *адрес:* 121357, г. Москва, Кременчугская ул., д. 46; *телефон:* +7 (915) 355-45-64; *e-mail:* urfin17@yandex.ru

M. L. Kazaryan, M. A. Shakhramanyan,

The Financial University under the Government of the Russian Federation, Vladikavkaz branch,

A. B. Murynin,

Institute for Scientific Research of Aerospace Monitoring "AEROCOSMOS", Moscow;

Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow,

A. A. Richter,

Institute for Scientific Research of Aerospace Monitoring "AEROCOSMOS", Moscow;

School 97, Moscow,

A. Davidov, I. Dementiev, D. Ignatiev,

School 97, Moscow

METHOD OF CONSTRUCTION OF 3D MODELS OF RIGID OBJECTS ON ONE IMAGE AND ITS APPLICATION IN PROJECT ACTIVITY OF SCHOOLCHILDREN

Abstract

A simple technique for constructing 3D models of ground surface objects from a single space image is proposed. The method gives an estimation of the model from the reference object, for which the value of the height corresponding to the given segment on the image giving the altitude coefficient in the spatial neighborhood of the object is known, i.e. scaling coefficient along the axis of the applicator. The altitude coefficient varies with time, i.e. from snapshot to snapshot, which allows to evaluate models of objects of different clarity, from low to high, using the space-time neighborhood of the reference object. With a distance from the standard, the altitude coefficient is continuously changing, which makes it possible to estimate the height of a distant object by "moving" the calculations through a chain of closer objects connecting the source and the final. Using an alternative "calculation by shadow" allows us to estimate the model when photographing the terrain in the nadir, i.e. in the absence of visibility of the side parts of the design object. In view of the availability of practical application of this methodology in the general education system, a number of project works of schoolchildren in the field of 3D modeling based on space images data.

Keywords: landfill, dump, 3D model, 3D modeling, economic zone, building, rigid object, Sketch Up, height, scale factor, Google Earth, project work.

* Дополнительные материалы к статье, в том числе представленные в статье рисунки в цветном изображении и высоком разрешении, можно скачать на сайте ИНФО: http://infojournal.ru/journals/school/school_04-2018/

Введение

3D-моделирование — одно из самых актуальных и перспективных направлений научно-исследовательской деятельности. Одна из областей 3D-моделирования — построение моделей ригидных объектов по данным космического мониторинга. «Ригидных» означает «твердых», имеющих определенную, установленную форму. Это, прежде всего, антропогенные объекты, такие как здания, сооружения и оборудование. При этом объекты могут быть подвижными или неподвижными, т. е. могут занимать переменное или постоянное, фиксированное положение относительно поверхности Земли. Автоматизированный подход к задаче 3D-моделирования поверхности дает грубую визуализацию объекта. Моделирование каждого объекта в отдельности в специальных программных средствах для 3D-моделирования (например, 3Ds-max, Sketch Up и др.) дает более точную визуализацию объекта и его компонентов: положение, окраску, форму и др. (хотя, возможно, с более высокими ошибками оценки самой 3D-модели).

Цель и задачи

Целью проведенного нами исследования была разработка методики 3D-моделирования ригидных объектов поверхности Земли по данным одного космического изображения, простой с точки зрения ее практического применения как специалистами в области аэрокосмического мониторинга, так и обычными пользователями.

Задачами, решаемыми в данной работе, были следующие:

- разработка и описание методики 3D-моделирования;
- подключение методики к такому направлению космического мониторинга, как мониторинг объектов захоронения отходов (построение 3D-моделей хозяйственных зон полигонов твердых бытовых отходов (ТБО)) [1, 8, 9, 11];
- применение методики в проектной деятельности общеобразовательных школ, в частности, в проектных работах школьников на примере конкретных полигонов ТБО.

Описание методики

Опишем методику построения 3D-моделей объектов правильной и неправильной форм посредством визуального детектирования космических изображений высокого пространственного разрешения в видимом диапазоне спектра.

В основе методики лежит выбор эталонного объекта, для которого известно значение высоты h , на базе которого пересчитываются значения высот объектов, детектируемых визуально на заданном исходном снимке (космическом изображении, аэрофотоснимке, фотографии и др.).

Будем считать, что исходное изображение ортотрансформировано, т. е. каждому пикселю соответствует постоянное пространственное разрешение в окрестности всего изображения, а также устранены искажения, вызванные дисторсией объектива и перспективой. При этом каждая точка на местности наблюдается строго «в надир», а по осям X и Y на «плоскости» Земли метрические длины l рассчитываются пропорционально разрешению k : $l = kN$, где N — длина в пикселях.

При известном h эталона (в частности, прямоугольного здания) для данного изображения I рассчитываем высотный коэффициент: $m = h/n$, где n — длина высотного отрезка, физически соответствующего данному значению высоты (например, на несущем ребре здания) на изображении. Предполагаем, что высотный коэффициент имеют здания в окрестности эталона. Тогда пересчет значений длин в пикселях n' для этих зданий на том же изображении I в соответствующие линейные (метрические) значения h' проводится по обратной формуле: $h' = mn'$.

На рисунке 1, а (изображение жилой зоны к востоку от полигона ТБО Кучино Московского региона, широта — $55^{\circ}43'45''\text{C}$, долгота — $37^{\circ}57'59''\text{B}$, Google Earth) для эталонного объекта A известно значение высоты $h = 3q = 15$ м (предполагаем, что это «пятиэтажка», т. е. число этажей $q = 5$, высота одного этажа — 3 м). В программе Google Earth измерения длин — в метрических единицах l , а не в пикселях N . Измерение n дает $n = 6,2$ м. Тогда высотный коэффициент $m = 2,42$. Измерение расстояния на том же изображении для соседнего расчетного объекта B (на его ребре) дает $n' = 3,6$ м. Тогда пересчет высоты расчетного объекта B : $h' = 8,7$ м — строение высотой в двух-трехэтажный дом (предположительно — торговый центр напротив «хрущевки»).

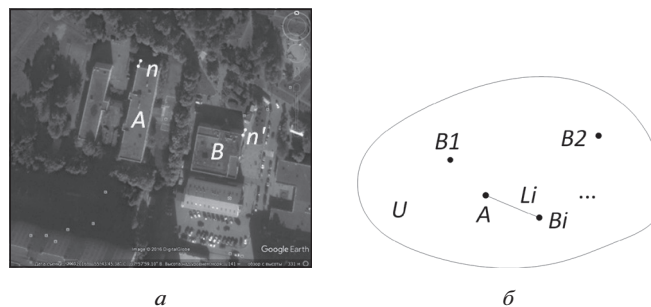


Рис. 1. Пространственные соотношения высотных объектов: а — пример изображения, б — схема соотношений

По известной высоте одного эталонного объекта A можно рассчитать застройку $\{B_i\}$ в его окрестности некоторого радиуса $r \sim 10^1$ км, точнее, объектов B_i застройки на расстояниях $L_i \leq r$, четко заданных на изображении, главным образом, несущих ребрах зданий (рис. 1, б). Непрерывное изменение высотного коэффициента по поверхности Земли может быть обусловлено непрерывностью изменения проективной геометрии объектов с изменением угла фотографирования поверхности, прерывистое (резкое) изменение коэффициента — геопривязкой снимков поверхности и их фрагментов на одной географической проекции.

Расчет объектов проводится в окрестности эталонного объекта по данным одного изображения. Часть ригидных объектов области наблюдения на изображении задана четко, а часть — нечетко. Чем четче задан объект, тем точнее может быть получена его 3D-модель.

Для объектов с низкой четкостью видимости на изображении I проводится пересчет значений высот по другому изображению J , на котором данный объект имеет высокую четкость. На рисунке 2:

- объект C — четкий на обоих изображениях — I и J ;
- объект A — объект-эталон, четкий на I ;
- объект B — расчетный объект, нечеткий на I , но четкий на J .

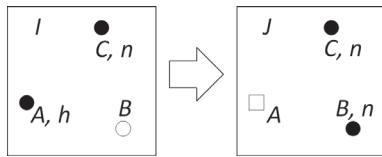


Рис. 2. Расчет нечеткого объекта по двум изображениям

Высотный коэффициент для изображения I :

$$m_I = \frac{h_A}{n_{IA}}$$

(h_A — известная высота эталона, n_{IA} — длина отрезка высоты объекта A на I).

Высота объекта C :

$$h_C = m_I n_{IC} = h_A \frac{n_{IC}}{n_{IA}}$$

(n_{IC} — длина отрезка высоты объекта C на изображении I).

Высотный коэффициент для изображения J :

$$m_J = \frac{h_C}{n_{JC}}$$

(n_{JC} — длина отрезка высоты объекта C на изображении J).

Расчет высоты объекта B :

$$h_B = m_J n_{JB} = h_C \frac{n_{JB}}{n_{JC}} = h_A \frac{n_{IC}}{n_{IA}} \frac{n_{JB}}{n_{JC}}$$

В общем случае проводится цепочка таких «элементарных» расчетов, позволяющая «дойти» от любого объекта A к любому объекту B в пространстве-времени на поверхности Земли (при условии, что все промежуточные объекты C на «своих» изображениях I и J должны быть интерпретируемы, A — интерпретируемо на первом изображении, B — на последнем):

$$h_q = h \cdot \prod_{i=1}^{q-1} \frac{n_{i+1}^i}{n_i^i},$$

где:

n_i^i — длина отрезка высоты объекта A_i на изображении I_j ;

$h = h_1$ — высота объекта-эталона;

h_j — высота j -го объекта;

q — число промежуточных объектов «на пути» от A до B включительно.

Объект A_i цепочки объектов $\{A_j\}$ интерпретируем на изображениях I_{i-1} и I_i , $A = A_1$, $B = A_q$.

Высотные коэффициенты изображений:

$$m_q = \frac{h}{n_q^{q-1}} \cdot \prod_{i=1}^{q-1} \frac{n_{i+1}^i}{n_i^i}.$$

Расчет цепочки высотных коэффициентов по пространственно-временной серии покрывающих изображений позволяет, в частности, рассчитать отдаленные и текущие объекты в пространстве или во времени относительно базового эталонного объекта. Например, в случае возникновения построек и пристроек в некоторый момент времени t_2 , которых не было (не будет) в момент t_1 прошлого: $t_1 < t_2$ (будущего: $t_1 > t_2$), для построения более полной 3D-модели инфраструктуры не в «момент» времени, а на некоторой протяженности во времени. Заметим, что в программе Google Earth моменты времени $t = t_i$ — неопределенные, но заданные на интервале $T_a < t < T_b$, где T_a — более ранний, T_b — более поздний моменты времени съемки, ближайшие к моменту t , для которого получена оценка модели объекта (a, b — номера отчетов моментов времени съемки, покрывающей данный объект).

Для трудно интерпретируемых, мелких или неправильных объектов и их деталей либо для альтернативного к «расчету по ребру» подхода можно применить «расчет по тени». Такой расчет может также быть проведен как по одному базовому изображению, так и «по цепочке». При этом расчет можно как комбинировать с «расчетом по ребру», так и нет. Тени также соответствует высотный коэффициент m_T , постоянный для данного изображения I (либо в некоторой своей окрестности для большой территории), также переменный во времени. Детали объекта могут быть визуализированы методиками улучшения качества визуализации.

Для объектов более сложных форм, чем прямоугольные, характерные части (для определения их длин на изображении) можно найти геометрическими построениями на самом космическом изображении. Для этого сначала интерпретируется форма объекта, затем проводятся геометрические построения по правилам планиметрии и фотограмметрии, в некоторых случаях интуитивные (рис. 3).

Программа по 3D-моделированию (Sketch Up) может быть применена не только для построения модели объекта поверхности Земли по рассчитанным линейным параметрам, но и на самом аэрокосмическом изображении. Для оси системы координат привязка делается к ортого-

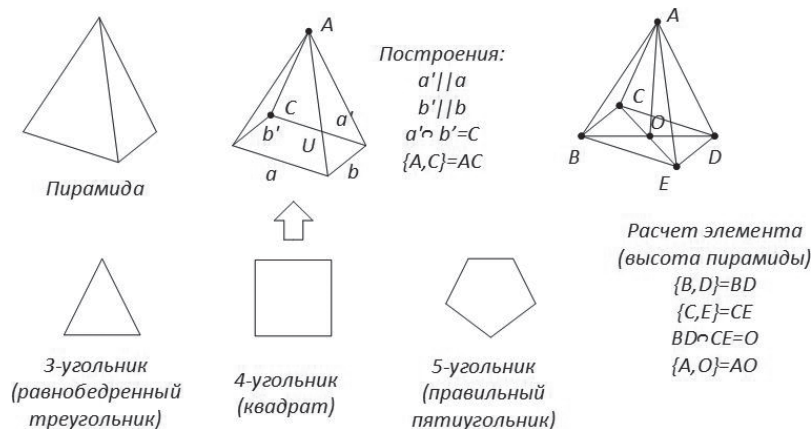


Рис. 3. Геометрические построения на космическом изображении (предположение и восстановление невидимой части объекта на примере пирамиды)

нальным направляющим объекта (например, по ребрам здания) и далее проводятся необходимые построения на изображении, но в пространстве привязанной системы координат. В случае, если ортогональные направляющие неизвестны, их можно найти соответствующими геометрическими построениями (см. рис. 3).

Примеры применения методики в проектной деятельности школьников

Простота и наглядность описанного подхода позволяет применить его в образовательном процессе. Для практического применения методики в образовательном процессе для получения входных изображений можно использовать программу Google Earth. Построение макетов объектов и их систем осуществляется в специальных программных средствах по 3D-моделированию (Sketch Up) [5, 7, 10].

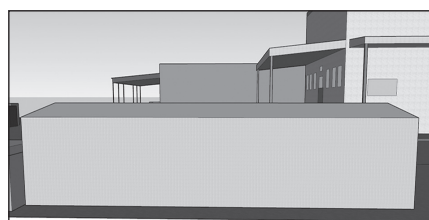
На рисунке 4 приведены примеры школьных проектов, выполненных в Sketch Up учениками ГБОУ «Школа № 97» г. Москвы Данилой Игнатьевым (VIII класс) (рис. 4, а), Артёмом Давыдовым (VIII класс) (рис. 4, б),

Ильей Дементьевым (X класс) (рис. 4, в) («Школьные проекты на цифровой карте Московского региона, апрель 2017 года»).

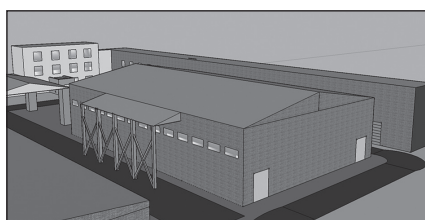
Для трех значимых полигонов ТБО Московского региона (Химки, Кучино, Электросталь) были построены 3D-модели их хозяйственных зон (рис. 5). Каждый учащийся на базе рассмотренной методики выработал и применил собственный подход к получению 3D-модели с высокой детализацией по данным одного изображения.

Д. Игнатьев просчитывал модель по цепочке изображений «по тени» с использованием дополнительных фотографий (вид спереди) меток программы Google Earth, в связи с низким качеством изображений поверхности (вид сверху), снятых за ограждением. Во все моменты времени поверхность заснята примерно в надир, а Солнце расположено под некоторым углом, что позволило оценивать высоту по тени. Несмотря на небольшое количество зданий в составе модели ($K=5$), потребовалось применение дополнительных наземных фотоизображений, так как одно из зданий имеет сложную структуру (рис. 6).

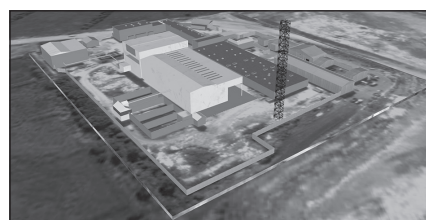
А. Давыдов строил многослойную модель по изображениям Google Earth с разных ракурсов для видимо-



а



б



в

Рис. 4. Схематические изображения 3D-моделей хоззон полигонов ТБО (ракурс):
а — Химки, б — Кучино, в — Электросталь



а



б



в



г



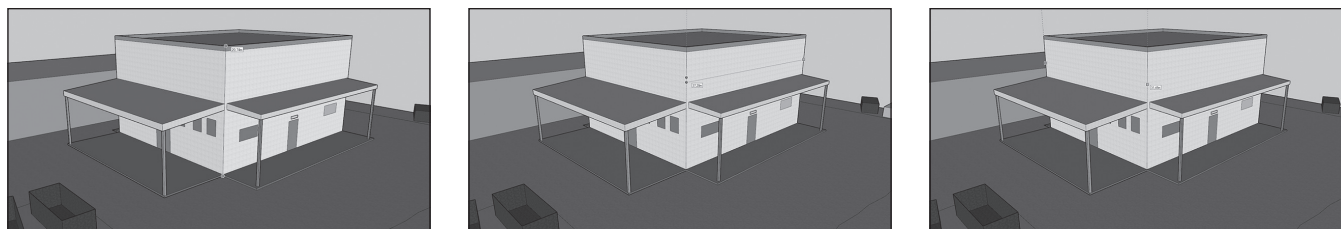
д



е

Рис. 5. Фрагменты космических изображений хозяйственных зон полигонов ТБО:

а, г — Химки; б, д — Кучино, в, е — Электросталь. а–в — выделение хоззоны, г–е — положение хоззоны на территории полигона



а

б

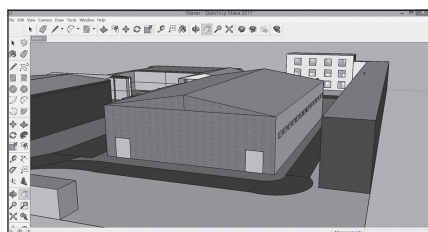
в

Рис. 6. Основные размеры здания (хоззона Химкинского полигона ТБО):
а — высота (20,74 м), б — длина (37,28 м), в — ширина (31,45 м)

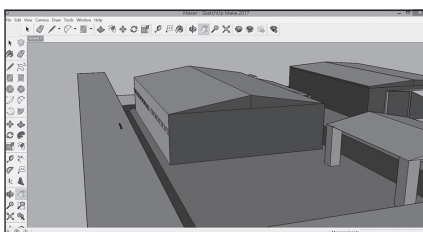
сти объектов (зданий) модели со всех сторон (в разные моменты времени углы фотографирования отличаются) для определения основных частей объектов и их пространственных соотношений. Затем для определения частей и деталей, невидимых на космических изображениях, использовал фотографии данной хоззоны, снятые с воздушного средства, которые он нашел в интернете (обнаружил две такие фотографии). В результате (рис. 7) неидентифицированные (информационность равна нулю) части объектов были обозначены характерным цветом (красным), а видимые части закрашены текстурой того материала, который эта часть имеет (металлические конструкции, кирпичные стены, стеклянные окна и т. д.). Учащийся построил и качественно сравнил две модели хоззоны: с применением дополнительных

фотографий и без применения (по данным космического снимка). Как оказалось, вторая модель является дополнением и уточнением первой и они не противоречат друг другу. Помимо данного полигона учащимся построена 3D-модель хоззоны полигона ТБО Шельбутово (близ аэропорта Внуково).

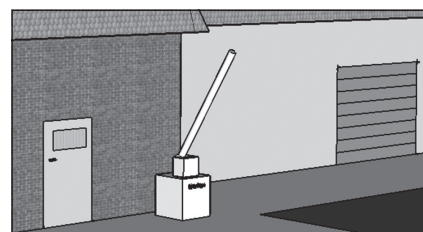
И. Дементьев строил модель (рис. 8), рисуя геометрические примитивы и текстуры полигонов, близкие к натуральным (см. рис. 8, д—ж), непосредственно по накладываемому космическому изображению. Для этого он расположил пространственные оси так, чтобы на привязанном изображении с некоторого ракурса центр координат совпадал с углом здания, а оси шли вдоль его ребер от этого угла. С данного ракурса были проведены основные построения, а достроения — по общим прави-



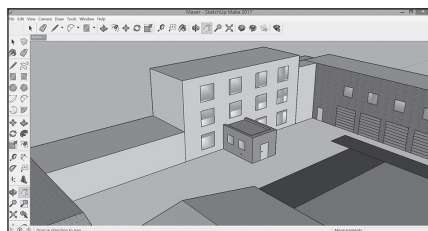
а



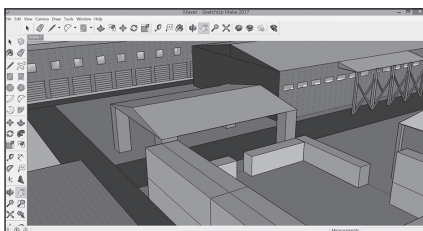
б



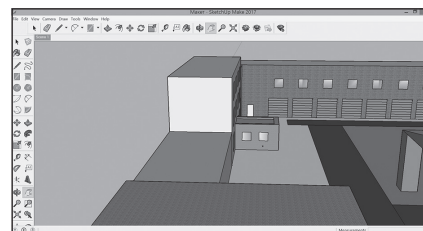
в



г



д



е

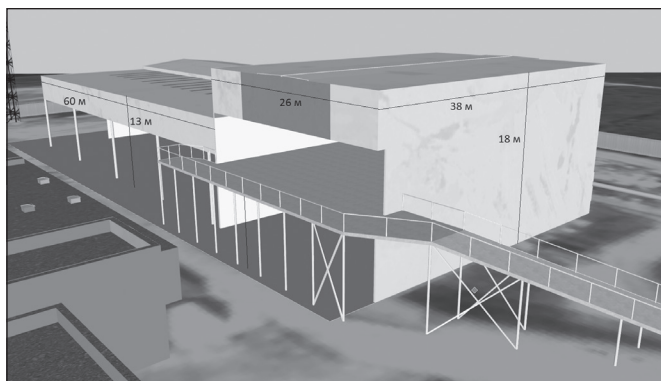


ж



з

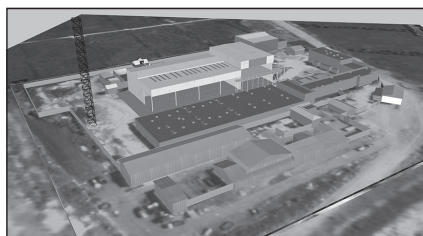
Рис. 7. Хоззона полигона ТБО Кучино:
а — видимая часть здания, б — невидимая часть здания; в — детали модели; г—е — скриншоты модели с разных мест и ракурсов; ж—з — дополнительные фотографии (интернет-ресурс)



а



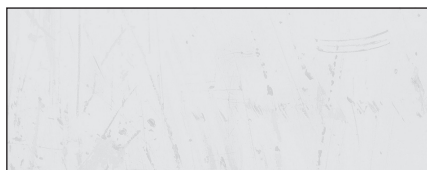
б



в



г



д



е



ж



з



и

Рис. 8. Хоззона полигона ТБО Электростальский:
 а — выделение объекта и обозначение основных размеров;
 б–г — скриншоты модели с разных ракурсов; д–ж — текстуры объектов (примеры);
 з–и — дополнительные данные (интернет-ресурс): з — фото, и — видео

лам (нанесение линий, контуров и т. д. с применением инструментов навигации в пространстве). Также он построил две модели хозяйственной зоны: с учетом и без учета фото с перспективой (см. рис. 8, з–и). Помимо данной хоззоны учащийся построил 3D-модель хоззоны труднодоступного полигона ТБО Торбеево.

Выводы

Представлена простая методика построения 3D-моделей объектов поверхности Земли по данным одного космического изображения (с возможностью

применения дополнительных информационных ресурсов). Предложенная методика с успехом применена для моделирования реальных объектов окружающего мира в проектной деятельности школьников. Такое применение развивает инженерные навыки учащихся и привлекает детей к науке и инженерии. Методика может быть также полезной в научной и прикладной деятельности, например, в построении пространственных и пространственно-временных карт обширных территорий, в построении 3D-моделей скрытых (экранированных другими объектами, плохого качества визуализации) объектов на аэрокосмических или фотоизображениях.

Список использованных источников

1. Бондур В. Г., Мурынин А. Б., Рихтер А. А., Шахрамьян М. А. Разработка алгоритма оценки степени деградации почвы по мультиспектральным изображениям // Известия ЮФУ. Технические науки. 2012. № 6 (131).
2. Визильтер Ю. В., Желтов С. Ю. Использование проективных морфологий в задачах обнаружения и идентификации объектов на изображениях // Известия РАН. Теория и системы управления. 2009. № 2.
3. Выголов О. В., Желтов С. Ю., Визильтер Ю. В. Mobile Robot Stereovision System For Real-Time Obstacle Detection // Мехатроника, управление и информатика. 2011. № 6.
4. Желтов С. Ю., Выголов О. В. Обнаружение препятствий на пути следования мобильного робота на основе метода радиального ортотрансформирования // Известия РАН. Теория и системы управления. 2005. № 5.
5. Киричук В. С., Шакинов А. К. Двухпороговый алгоритм обнаружения точечных объектов по стереоизображениям // Автоматизация. 2014. Т. 50. № 6.
6. Мурынин А. Б., Матвеев И. А. Принципы построения системы стереоскопического зрения для управления движением робота // Известия РАН. Теория и системы управления. 2003. № 3.
7. Попело В. Д., Ванеева М. В. Автоматизированная корреляционная процедура выбора соответственных точек на цифровых изображениях в стереопаре // Модели и технологии природообустройства (региональный аспект). 2015. Т. 1.
8. Рихтер А. А., Казарян М. Л., Шахрамьян М. А. Метод автоматизации оценки индексов подстилающей поверхности и их изменения во времени по космическим изображениям и его применение при оценке состояния окружающей среды в окрестности полигонов твердых бытовых отходов // Известия Томского политехнического университета. Инжиниринг ресурсов. 2016. № 8.
9. Рихтер А. А., Казарян М. Л., Шахрамьян М. А. Методика автоматического детектирования компонент объектов захоронения отходов по космическим изображениям // Известия Томского политехнического университета. Инжиниринг георесурсов. 2017. Т. 328. № 3.
10. Суцеский Д. В. Выявление и 3d визуализация искусственных объектов земной поверхности с использованием лидарных и оптических спутниковых изображений // Вестник Херсонского национального технического университета. 2014. № 3 (50).
11. Richter A., Kazaryan M., Shakhramanyan M., Borisova D., Stankova N., Ivanova I. Information modeling of waste disposal sites // Ecological Engineering and Environment Protection. 2017. No. 1.

НОВОСТИ

Российские школьники завоевали девять наград в международном научном конкурсе

С 12 по 18 мая в Питтсбурге (США) прошел международный конкурс научных проектов школьников Intel ISEF (International Science and Engineering Fair). В нем приняли участие более 1800 школьников из более чем 75 стран мира. По итогам конкурса российская команда заняла девять призовых мест в различных номинациях от химии до робототехники.

Наград удостоены:

- Р. Николаенко, Предуниверситарий НИЯУ МИФИ — призер по секции «Physics and Astronomy» («Физика и астрономия»);
- И. Белоусова, Московский химический лицей — призер по секции «Chemistry» («Химия»);
- В. Лебедева, гимназия № 5, г. Королев, Московская область — призер по секции «Biomedical Engineering» («Биомедицинская инженерия»);
- Д. Казанцев и Д. Семенов, лицей № 12, Екатеринбург — команда-призер по секции «Robotics and Intelligent Machines» («Робототехника и умные машины»);
- А. Сердюков, школа № 564, Санкт-Петербург — призер по секции «Mathematics» («Математика»);
- Ю. Маслова и А. Андрианова, Московский химический лицей — команда-призер по секции «Chemistry» (Химия);
- А. Савельева, школа № 57, г. Москва — призер по секции «Mathematics» («Математика»);
- П. Хакимов и М. Шароян, лицей г. Кемерово — команда-призер по секции «Robotics and Intelligent Machines» («Робототехника и умные машины»);

- И. Захаров, лицей № 1533, г. Москва — призер по секции «Systems Software» («Системное программное обеспечение»).

Также ребята получили шесть специальных призов от NASA, Американского математического общества и Химического общества.

Intel ISEF — международный конкурс научных проектов для подростков от 13 до 18 лет. Цель конкурса — выявление и поддержка талантливых молодых ученых, развитие исследований в области прикладных и фундаментальных наук, а также технического творчества.

Конкурс Intel ISEF называют «Малой Нобелевской премией» — более 20 будущих нобелевских лауреатов становились призерами конкурса в школьном возрасте. Работы участников конкурса затрагивают важные области жизнедеятельности человека и современного мира: поиск альтернативных источников энергии, лечение онкологических заболеваний, решение острых климатических проблем и другие. Более 30 процентов участников, представляющих работы, получают патент на свое изобретение.

Отбор команды от России проходил в рамках пяти российских научно-технических соревнований: Приволжского конкурса исследовательских работ школьников РОСТ (Нижний Новгород), Всероссийского конкурса научных работ школьников «Юниор», Всероссийской научно-практической конференции одаренных школьников «Авангард», Научно-инженерного конкурса «Ученые будущего» (Москва) и Балтийского научно-инженерного конкурса «БалтикSEF» (Санкт-Петербург).

(По материалам, предоставленным пресс-службой Минобрнауки России)

А. А. Рединова,
Смоленский государственный университет

ФОРМИРОВАНИЕ МЕТАПРЕДМЕТНЫХ УМЕНИЙ И НАВЫКОВ НА УРОКАХ ИНФОРМАТИКИ НА ОСНОВЕ СТРУКТУРИРОВАНИЯ УЧЕБНОГО ТЕКСТА*

Аннотация

В статье рассматриваются проблема формирования метапредметных умений и навыков и возможность применения для этого структурирования учебных текстов. Приведен пример урока по информатике на тему «Управление и кибернетика» для девятого класса. В рамках урока учащимся предлагается использовать адаптированный учебный текст и выполнить упражнения по его структурированию и составлению модели.

Ключевые слова: метапредметный подход, структурирование, учебный текст, модель текста.

Контактная информация

Рединова Александра Андреевна,
аспирант кафедры информационных и образовательных технологий Смоленского государственного университета; адрес: 214000, г. Смоленск, ул. Пржевальского, д. 4; телефон: (4812) 70-02-45; e-mail: a-redinova@yandex.ru

A. A. Redinova,
Smolensk State University

FORMING METASUBJECT SKILLS AT INFORMATICS LESSONS ON THE BASE OF THE STRUCTURING OF THE EDUCATIONAL TEXT

Abstract

The possibility of using the structuring of the educational text for metasubject skills forming is considered in the article. The example of the informatics lesson on the theme "Control and cybernetics" for the 9th grade is given. At the lesson students are encouraged to use the adapted educational text, structure it and compile its model.

Keywords: metasubject approach, structuring, educational text, text model.

В сегодняшнем мире человек должен быть как никогда гибким, мобильным и готовым к постоянным изменениям. В связи с этими требованиями меняется и образование: оно должно готовить учащихся к обучению на протяжении всей жизни. Поэтому особую роль приобретает *умение учиться*. С ним связано и достижение метапредметных результатов обучения.

Приставка «мета» означает «за», «следующее после» и используется для обозначения явлений, стоящих за или над другими явлениями. В рамках педагогической науки метапредметной, или метадеятельностью, называется универсальная деятельность, которая осуществляется в рамках других видов деятельности. Соответственно, метапредметными называются умения и навыки универсальные, применимые во всех сферах жизни. Федеральные государственные образовательные стандарты конкретизируют метапредметные результаты освоения учащимися основной общеобразовательной программы. Обобщая содержание статьи 10 ФГОС основного общего образования [3], **можно выделить следующие метапредметные результаты обучения, которые должна обеспечить школа:**

- умение находить межпредметные понятия и устанавливать межпредметные связи;
- систематичность и интегрированность знаний — умение переносить знания из одной предметной области в другую;
- умение самостоятельно планировать и осуществлять учебную и познавательную деятельность:
 - целеполагание;
 - прогнозирование;
 - выбор наиболее эффективных способов решения задач;
 - оценка правильности выполнения задания и собственных возможностей ее решения;
 - самоконтроль;
 - самокоррекция (регулятивные универсальные учебные действия);
- познавательные универсальные учебные действия:
 - анализ;
 - классификация;
 - формулировка определений;
 - сравнение;
 - логические умозаключения;
 - умение создавать и использовать модели и схемы в учебной и познавательной деятельности;
- коммуникативные универсальные учебные действия;
- ИКТ-компетенция;
- экологическое мышление.

Нами разработана **методика формирования метапредметных умений и навыков с помощью структурирования учебного текста**. По мнению С. Г. Антоновой, Л. Г. Тюриной [1, с. 14], учебные тексты, содержащиеся в учебниках с готовыми выводами в конце параграфа, выделением основных идей и определений и другими видами маркировки, избавляют учащихся от необходимости самим выполнять указанные действия, хотя именно в результате их выполнения и формируются метапредметные умения и навыки. В дальнейшем будем называть такой текст *базовым*, а текст без маркировки — *начальным*.

Мы предлагаем ввести в учебный процесс **упражнения по маркировке начального текста**.

* Материалы к статье можно скачать на сайте ИНФО:
http://infojournal.ru/journals/school/school_04-2018/

Приведем **пример урока по информатике с использованием предложенной методики.** В самой информатике заложен большой метапредметный потенциал, так как она используется в прикладном качестве в различных научных дисциплинах.

Тема урока: Управление и кибернетика.

Класс: девятый.

Цели урока:

воспитательные (для достижения личностных результатов):

- развитие навыков самоорганизации;
- воспитание вежливого отношения к оппоненту;
- воспитание активной позиции в обучении;

образовательные (для достижения предметных результатов):

- изучение основных понятий управления: управление, исполнитель, алгоритм управления;

развивающие (для достижения метапредметных результатов):

- развитие навыков смыслового чтения;
- развитие навыков создания и использования знаков, символов и моделей в учебной и познавательной деятельности.

Ход урока.

Рассмотрим ход урока с двух позиций:

- традиционная работа с учебником, где учащемуся предлагается прочитать текст из учебника и выполнить задания;
- занятие с помощью предложенной методики, в которой учащемуся дается начальный текст (*Приложение 1*).

В рамках данного урока предлагается использовать учебник И. Г. Семакина и др. «Информатика и ИКТ» [2]. Возможные варианты ответов даны в *Приложении 4*.

№ п/п	Этап урока	Деятельность на традиционном занятии	Деятельность на занятии на основе предложенной методики	Формируемые метапредметные умения и навыки
1	Организационный момент (2 мин)	Учащиеся занимают свои места. Проверка присутствующих.		
2	Актуализация знаний (10 мин)	Обсуждение темы урока и возможностей компьютера в управлении.	Мини-эссе «Компьютер в нашей жизни» (предложения, идеи, словосочетания). Обсуждение мини-эссе.	Систематичность знаний. Возможность применения своего опыта в учебной деятельности
3	Получение новых знаний (10 мин)	Прочитайте текст.	Задание 1. Просмотрите предложенный текст (<i>Приложение 1</i>). О чем он? Дайте название параграфу и выделите основные его темы. Задание 2. Выделите в предложенном тексте основные мысли и определения (жирным шрифтом или курсивом, цветом, подчеркиванием — на компьютере; цветом, подчеркиванием или любыми другими значками — на бумаге).	Навыки смыслового чтения. Универсальные учебные действия: анализ, классификация, ранжирование, знаково-символические действия
4	Закрепление (10 мин)	Что такое управление? кибернетика? алгоритм управления? Приведите свои примеры управляющих и управляемых объектов.	Задание 3. Нарисуйте схему взаимодействия управляющего и управляемого объектов. Задание 4. Заполните таблицу (<i>Приложение 2</i>) двумя-тремя примерами.	Умение составлять и использовать на практике модели и схемы
5	Подведение итогов урока (10 мин)	Что нового вы узнали на уроке? Как можно применять компьютер в управлении?	Задание 5. Составьте модель предложенного текста: выпишите основные понятия и свяжите их по смыслу между собой (можно использовать схему из задания 3). Задание 6. Сделайте выводы по параграфу. Задание 7. Встречались ли на сегодняшнем уроке понятия, общие для нескольких наук? Задание 8. Какие вопросы возникли у вас по ходу урока?	Универсальные учебные действия: логическое умозаключение, умение делать выводы. Умение находить межпредметные понятия
6	Домашнее задание (3 мин)	Прочитайте параграф учебника. Ответьте на вопросы	Выберите из задания 8 один-два вопроса и найдите на них ответ. Заполните бланк (<i>Приложение 3</i>).	Мотивация для дальнейшего развития. Самостоятельное планирование и осуществление познавательной деятельности

На традиционном занятии учащемуся предлагается прочитать текст учебника. Уже с первых строк учащийся знает, о чем этот текст. А в конце параграфа предлагаются выводы. Однако такой подход лишает учащегося осознанности получения знаний.

В предлагаемой методике структурирования текста в результате выполнения упражнений учащийся не «получает» готовые знания, а «добывает» их сам.

Задания на составление модели текста можно применять на любых этапах урока. В таких заданиях правильных ответов может быть несколько. Поэтому чрезвычайно важно организовать обсуждение полученных моделей, выявить типичные ошибки и составить единую модель, которая будет объединять наиболее удачные решения.

В результате урока, проведенного по предложенной методике, у учащегося получится собственный конспект учебного материала. Если создавать модели материала по каждой теме и объединять их между собой, получится единая модель предметной области, которую можно использовать для повторения, а также для нахождения межпредметных понятий и установления метапредметных связей.

Безусловно, и традиционную работу с учебником можно сделать интересной для учащихся, но нашей

целью было показать возможности учебного текста в формировании метапредметных умений и навыков. Предложенный вариант работы с текстом распространяется не только на информатику, но и на другие науки гуманитарного и естественнонаучного циклов. Также формирование метапредметных умений и навыков осуществляется «без отрыва» от основной задачи — изучения необходимого материала, т. е. формирования предметных умений и навыков. Таким образом, предлагаемая методика может использоваться для формирования метапредметных умений и навыков как целиком, так и с помощью отдельных упражнений.

Список использованных источников

1. Антонова С. Г., Тюрина Л. Г. Новое поколение учебной литературы: теоретические и методические предпосылки // Университетская книга. 2000. № 10.
2. Семакин И. Г., Залогова Л. А., Русаков С. В., Шестакова Л. В. Информатика и ИКТ: учебник для 9 класса. М.: БИНОМ. Лаборатория знаний, 2012.
3. Федеральный государственный образовательный стандарт основного общего образования (в ред. Приказа Минобрнауки России от 29.12.2014 № 1644). [http://минобрнауки.рф/документы/938/файл/749/приказ Об утверждении 1897.pdf](http://минобрнауки.рф/документы/938/файл/749/приказ%20Об%20утверждении%201897.pdf)

Приложения*

Приложение 1

Начальный текст темы

В 1948 году в США и в Европе вышла книга американского математика Норберта Винера «Кибернетика, или управление и связь в животном и машине». Эта книга провозгласила рождение новой науки — кибернетики. Не случайно время появления этого научного направления совпало с созданием первых ЭВМ. Н. Винер предвидел, что использование ЭВМ для управления станет одним из важнейших их приложений, а для этого потребуются глубокий теоретический анализ самого процесса управления. Этому и посвящена кибернетика. Управление есть целенаправленное воздействие одних объектов, которые являются управляющими, на другие объекты — управляемые. Простейшая ситуация — два объекта: один — управляющий, второй — управляемый. С кибернетической точки зрения все варианты управляющих воздействий следует рассматривать как управляющую информацию, передаваемую в форме команд. В данном выше определении сказано, что управление есть целенаправленный процесс, т. е. команды отдаются не случайным образом, а с вполне определенной целью. В простейшем случае цель может быть достигнута после выполнения одной команды. Для достижения более сложной цели бывает необходимо выполнить последовательность (серию) команд. Последовательность команд по управлению объектом, выполнение которой приводит к достижению заранее поставленной цели, называется алгоритмом управления. В таком случае объект управления можно назвать исполнителем управляющего алгоритма. С точки зрения кибернетики взаимодействие между управляющим и управляемым объектами рассматривается как информационный процесс. С этой позиции оказалось, что самые разнообразные процессы управления в природе, технике, обществе происходят сходным образом, подчиняются одним и тем же принципам [2, с. 140—142].

Приложение 2

Примеры

Управляющий объект	Управляемый объект	Управляющее воздействие	Цель воздействия	Исполнитель управляющего алгоритма

* Приложения можно скачать на сайте ИНФО: http://infojournal.ru/journals/school/school_04-2018/

Бланк структурирования материала параграфа

Глава ____.	
§ 1.	
Основные темы параграфа: • •	
Основные понятия параграфа	Примеры
Выводы	
Дополнительный материал/вопросы/идеи	

Ответы на задания

Пример выполнения задания 2.

В 1948 году в США и в Европе вышла книга американского математика *Норберта Винера* «Кибернетика, или управление и связь в животном и машине». Эта книга провозгласила рождение новой науки — **кибернетики**. Не случайно время появления этого научного направления совпало с созданием первых ЭВМ. Н. Винер предвидел, что использование ЭВМ для управления станет одним из важнейших их приложений, а для этого потребуются глубокий теоретический анализ самого процесса управления. Этому и посвящена кибернетика.

Управление есть целенаправленное воздействие одних объектов, которые являются управляющими, на другие объекты — управляемые.

Простейшая ситуация — два объекта: один — управляющий, второй — управляемый. С кибернетической точки зрения все варианты управляющих воздействий следует рассматривать как управляющую информацию, передаваемую в форме команд. В данном выше определении сказано, что *управление есть целенаправленный* процесс, т. е. команды отдаются не случайным образом, а с вполне определенной целью. В простейшем случае цель может быть достигнута после выполнения одной команды. Для достижения более сложной цели бывает необходимо выполнить последовательность (серию) команд.

Последовательность команд по управлению объектом, выполнение которой приводит к достижению заранее поставленной цели, называется **алгоритмом управления**.

В таком случае объект управления можно назвать *исполнителем управляющего алгоритма*. С точки зрения кибернетики взаимодействие между управляющим и управляемым объектами рассматривается как *информационный процесс*. С этой позиции оказалось, что самые разнообразные процессы управления в природе, технике, обществе происходят сходным образом, подчиняются одним и тем же принципам.

Пример выполнения задания 4.

Управляющий объект	Управляемый объект	Управляющее воздействие	Цель воздействия	Исполнитель управляющего алгоритма
Человек	Телевизор	Человек нажимает кнопку	Найти интересующую передачу	Телевизор
Хозяин	Собака	Хозяин голосом подает команду	Выполнить определенное задание хозяина	Собака
Светофор	Автомобиль	Светофор разными цветами управляет движением автомобилей и пешеходов на перекрестке	Благополучно проехать перекресток	Автомобиль

Пример выполнения задания 5.**Пример выполнения домашнего задания.**

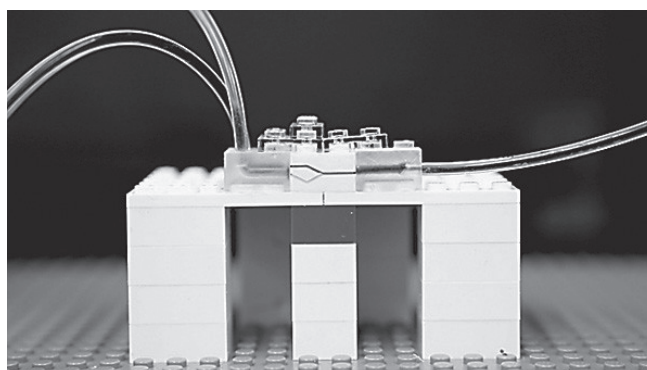
Глава 5. Управление и алгоритмы	
§ 1. Кибернетика как наука управления	
Основные темы параграфа: • возникновение кибернетики; • управление как целенаправленный процесс; • алгоритм управления.	
Основные понятия параграфа Управление — целенаправленное воздействие управляющих объектов на управляемые. Последовательность команд по управлению объектом, выполнение которой приводит к достижению заранее поставленной цели, называется алгоритмом управления.	<i>Примеры представлены в таблице «Пример выполнения задания 4» (Приложение 4)</i>
Выводы Управление — процесс целенаправленный. Управляющий объект воздействует на управляемый с помощью команд. Серия команд называется алгоритмом управления.	
Дополнительный материал/вопросы/идеи Может ли исполнитель каким-то образом реагировать на управляющее воздействие?	

НОВОСТИ

Микрожидкостные устройства научились собирать из кирпичиков LEGO

Современные микрожидкостные устройства, как правило, имеют форму пластин с микроскопическими каналами, входами, выходами и пересечениями, выполняющими смешивание и разделение, перемещение и хранение жидкостей. Исследователи из Массачусетского технологического института нашли необычную платформу для изготовления микрожидкостных устройств — кирпичики LEGO. Возможность сборки по принципу конструктора из готовых модулей может значительно облегчить разработку прототипов новых устройств. По размеру кирпичики в наборах LEGO соответствуют обычным микрожидкостным схемам. Они дешевы и изготавливаются с высокой точностью. Каналы для жидкости в них можно нарезать с помощью микрофрезерования и герметизировать пленкой. Сейчас устройства на основе кирпичиков LEGO уже могут работать с биологическими

образцами, выполняя сортировку клеток, фильтрацию жидкостей и захват молекул в отдельных каплях.



(По материалам «Computerworld Россия»)

А. Ю. Федосов,

Российский государственный социальный университет, г. Москва,

М. В. Маркушевич,

школа № 1352, г. Москва

ПРИМЕНЕНИЕ СВОБОДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В УЧЕБНОМ ПРОЦЕССЕ КАК ФАКТОР ДУХОВНО-НРАВСТВЕННОГО ВОСПИТАНИЯ УЧАЩИХСЯ

Аннотация

В статье рассматривается влияние выбора типа программного обеспечения, используемого в учебно-воспитательной деятельности образовательной организации, на духовно-нравственное воспитание учащихся. Особое внимание уделяется необходимости формирования свободной личности, умеющей сделать самостоятельный компетентный, квалифицированный выбор в области программного обеспечения и нести ответственность за последствия принятого решения.

Ключевые слова: свободное программное обеспечение, духовно-нравственное воспитание, свобода.

Контактная информация

Федосов Александр Юрьевич, доктор пед. наук, доцент, профессор кафедры информатики и прикладной математики Российского государственного социального университета, г. Москва; *адрес:* 129226, г. Москва, ул. Вильгельма Пика, д. 4, стр. 1; *телефон:* (495) 255-67-67, доб. 1992; *e-mail:* alex_fedosov@mail.ru

Маркушевич Михаил Владимирович, учитель информатики и черчения школы № 1352, г. Москва; *адрес:* 107497, г. Москва, ул. Байкальская, д. 44; *телефон:* (495) 467-77-55; *e-mail:* mihael11@yandex.ru

A. Yu. Fedosov,

Russian State Social University, Moscow,

M. V. Markushevich,

School 1352, Moscow

USING FREWARE IN THE EDUCATIONAL PROCESS AS A FACTOR OF SPIRITUAL AND MORAL EDUCATION OF STUDENTS

Abstract

The article examines the influence of the choice of the type of software used in the educational activities of the educational organization, on the spiritual and moral education of students. Particular attention is paid to the need for the formation of a free individual who can make an independent, competent, qualified choice in the field of software and be responsible for the consequences of the decision.

Keywords: freeware, spiritual and moral education, freedom.

Чтобы жить, нужно солнце, свобода и маленький цветок.

Ганс Христиан Андерсен

Проблеме выбора между свободным и проприетарным программным обеспечением в целях его дальнейшего использования в учебно-воспитательном процессе среднеобразовательной школы было посвящено большое количество научных работ [см. например, 3, 9, 10, 11, 13, 16], но в них в центре внимания находились в основном методические и технические аспекты рассматриваемой проблемы. Ранее авторы уже постарались проследить связь типа используемого программного обеспечения с духовно-нравственным становлением младших школьников [11]. В настоящей работе предлагается обратить внимание также на духовно-нравственную сторону рассматриваемой проблемы, наиболее важную даже в таком «техническом» вопросе, как выбор типа программного обеспечения, так как речь идет о воспитательном воздействии на формирующуюся личность ребенка.

Как следует из самого названия одного из рассматриваемых нами типов ПО, проблема выбора между свободным и проприетарным программным обеспечением напрямую связана в том числе и с такой общефилософской и духовной категорией, как свобода. В частности, немецкий философ начала XIX века Георг Вильгельм Фридрих Гегель замечал по этому поводу следующее: «Воспитание имеет целью сделать человека самостоятельным существом, т. е. существом со свободной волей» [4].

Попытаемся понять: о каком виде свободы идет речь в данном случае, о свободе кого и от чего? Ведь свобода — это категория, допускающая различные трактовки.

Например, свободу можно понимать как состояние субъекта, в котором он является определяющей причиной своих действий, т. е. они не обусловлены непосредственно иными факторами, в том числе природными, социальными, межличностно-коммуникативными и индивидуально-родовыми. Отсутствие возможности выбора, вариантов исхода события равносильно отсутствию свободы [14].

В области права свобода — это закрепленная в конституции или ином законодательном акте возможность определенного поведения человека (например, свобода слова, свобода вероисповедания и т. д.) [12].

Еще во Франции в 1789 году в «Декларации прав человека и гражданина» свобода человека трактовалась как возможность «делать все, что не наносит вреда другому: таким образом, осуществление естественных прав каждого человека ограничено лишь теми пределами, которые обеспечивают другим членам общества пользование теми же правами. Пределы эти могут быть определены только законом» [14].

Обратим внимание на тот факт, что в настоящее время в России свобода человека признается высшей ценностью, а также гарантируется ее соблюдение и защита Конституцией Российской Федерации. В статье 2 Конституции РФ, в частности, говорится:

«1. В Российской Федерации признаются и гарантируются права и свободы человека и гражданина согласно общепризнанным принципам и нормам международного права и в соответствии с настоящей Конституцией.

2. Основные права и свободы человека неотчуждаемы и принадлежат каждому от рождения» [8].

В разделе права, касающемся получения и использования информации, свобода — это неприкосновенность частной жизни при использовании информационных технологий, что также гарантируется статьей 24 Конституции РФ [8].

В авторском праве под свободой применительно к программному обеспечению понимаются права («свободы») на его неограниченную установку, запуск, свободное использование, изучение, распространение и изменение (совершенствование), а также распространение копий и результатов изменения [17]. Именно программное обеспечение с вышеуказанными правами называется свободным программным обеспечением (СПО). Надо отметить, что свободное программное обеспечение чаще всего является ПО с открытым кодом, ибо иначе невозможно было бы его изменять, совершенствовать.

Таким образом, на основании вышеприведенного краткого обзора будем *относить свободу к высшим духовным ценностям, которая позволяет человеку реализовать свою субъективную позицию, осуществлять поступки, исходя только из понимания их значимости для себя, без внешнего принуждения* [12].

Далее попытаемся определить, как связаны освоение учащимися такой общечеловеческой духовной ценности, как свобода, в процессе выбора и дальнейшего использования ими в своей учебной деятельности типа программного обеспечения и их духовно-нравственное воспитание.

Воспитание мы рассматриваем как сопровождение целенаправленного процесса развития человека, основанного на гуманистических, нравственных ценностях и реализующегося в процессе взаимодействия воспитателей и воспитанников [12]. Под духовно-нравственным воспитанием будем понимать один из аспектов воспитания, направленный на усвоение подрастающими поколениями и претворение в практическое действие и поведение высших духовных ценностей.

Применительно к духовно-нравственному воспитанию выделяют следующие высшие духовные ценности:

- индивидуально-личностные (жизнь человека, права ребенка, честь, свобода, достоинство);
- семейные (отчий дом, родители, семейный лад, родословная семьи, ее традиции);
- национальные (образ жизни, поведения, общения; Родина, святыни страны, национальная геральдика, родной язык, родная земля, народная культура, единство нации);
- общечеловеческие (биосфера как среда обитания человека, экологическая культура, мировая наука и культура, мир на земле и т. д.) [6].

В своей работе «Свобода и воспитание» [12] один из основоположников экзистенциальной педагогики М. И. Рожков рассматривает воспитательный процесс в трех различных аспектах в зависимости от поставленных целей. Для нашей работы важность представляют первые два аспекта.

Во-первых, воспитание — это педагогический компонент процесса социализации, ориентированный на интеграцию человека в социальное общество. В этой связи невозможно отрицать, что одной из существенных составляющих успешной социализации является умение подростка сознательно и квалифицированно противостоять различным видам угроз, присутствующим в современном информационном обществе.

Попытаемся понять, с какими информационными угрозами может столкнуться школьник при выборе программного обеспечения и каковы преимущества и недостатки свободного программного обеспечения перед альтернативными вариантами использования проприетарного ПО.

Так, нравственный аспект принципа свободы в отношении программного обеспечения прекрасно был сформулирован Ричардом Столлманом, основателем движения Open Source. По этому поводу он говорит следующее: «Закрытое программное обеспечение неэтично, потому что оно лишает пользователя базовых свобод — контроля своего собственного компьютера и сотрудничества с другими пользователями. Оно может быть также низкокачественным и небезопасным, но это вторичные вопросы. Я откажусь от него, даже если оно лучшее в мире, просто потому, что я слишком высоко ценю свободу, чтобы отказаться от нее ради такого ПО» [15].

Что же подразумевает Ричард Столлман, когда говорит о том, что закрытое ПО лишает пользователя контроля над собственным компьютером? Дело в том, что программное обеспечение с закрытым кодом вполне может включать в себя скрытые программные модули, функция которых заключается в сборе и отправке информации о пользователе данного ПО и его действиях на локальном компьютере и в сети Интернет. Иными словами, закрытое ПО является для его владельца неким «черным ящиком», управлять которым он может лишь отчасти. Программное обеспечение с открытым кодом, которым чаще всего и является *свободное программное обеспечение*, напротив, вполне «прозрачно».

Таким образом, **первое преимущество использования свободного программного обеспечения в учебном процессе в духовно-нравственном отношении заключается в формировании у учащихся ментальности квалифицированного члена современного информационного общества, различающего наличие потенциальных информационных угроз и готового противостоять им.**

Вышеуказанная проблема обозначена в Доктрине информационной безопасности Российской Федерации, утвержденной 5 декабря 2016 года Президентом Российской Федерации [5]. В ней говорится о том, что ряд зарубежных стран наращивает возможности информационно-технического воздействия на российскую информационную инфраструктуру. Очевидно, что проприетарное закрытое программное обеспечение иностранного производства, используемое в учебном процессе общеобразовательных школ России, вполне может стать инструментом такого воздействия в обозримом будущем.

Во-вторых, воспитанием является само развитие личности человека, оказание ему максимальной помощи в реализации заложенных в него потенциальных возможностей, осуществление того, чтобы он достиг собственной вершины.

В свою очередь, достижение вершины, раскрытие всех своих потенциалов возможно, по глубокому убеждению авторов, только для внутренне свободного человека. В нашей ситуации речь идет о свободе и умении человека совершить квалифицированный, компетентный выбор того типа программного обеспечения, которое оптимально для решения пользовательских задач, стоящих перед ним. Под фактором, косвенно ограничивающим

данную свободу, надо понимать навязанный в процессе обучения в начальной школе и, возможно, закреплённый в основной и старшей школе стереотип поведения в случае осуществления такого выбора.

Попытаемся понять, что подвело авторов настоящей работы к формулированию вышеуказанной проблемы.

Дело в том, что, к сожалению, авторы различных существующих в настоящее время методик преподавания информатики в школе не обращают внимания на столь важный аспект, как правильная ориентация учащихся в нравственном плане при выборе типа программного обеспечения для разрабатываемых ими методик. Чаше всего авторы предлагают определенное программное обеспечение в качестве базового для своих методик, но никаким образом не обосновывают выбор именного данного ПО, не приводят никаких доводов в его пользу, также не отмечают недостатков предлагаемого к использованию программного обеспечения.

Насколько оправдан данный «молчаливый» выбор программного обеспечения и чем его можно объяснить?

Если связь типа используемого программного обеспечения с качеством обученности школьников является до настоящего момента весьма неоднозначной и требующей дальнейших педагогических исследований, то факт влияния выбора между свободным и проприетарным ПО на нравственную сферу школьников является совершенно очевидным, с точки зрения авторов, по следующим причинам:

- учащиеся, особенно учащиеся начальной школы, весьма восприимчивы к демонстрируемым им моделям поведения, в том числе к предлагаемым им для использования компьютерным программам;
- воспринятый ими в ходе обучения информатике стереотип в области «принятых на вооружение» программ фиксируется на длительное время;
- принцип свободы имеет непосредственное отношение не только к правовой, но и к духовно-нравственной категории, так как влияет на формирование основы личности ребенка, становление его системы ценностей.

В этой связи, в случае рассмотрения принципа свободы в области информационных технологий применительно к общеобразовательной школе, с точки зрения авторов, надо понимать следующее: если в течение срока обучения в начальной школе учащийся привыкнет работать на проприетарном закрытом ПО, то и в дальнейшей взрослой жизни освоение им альтернативных вариантов решения проблемы выбора программного обеспечения будет крайне затруднено. Вполне вероятно, что бывший школьник усвоит некий шаблон, навязанный ему еще в школе, и будет действовать на основе этого шаблона. В пользу данной гипотезы говорит тот факт, что даже некоторые школьные преподаватели информатики, являющиеся априори профессионалами в области информационных технологий или хотя бы высококвалифицированными пользователями, вполне приняли выбор проприетарного закрытого программного обеспечения «по умолчанию».

Характерным «шаблонным» поведением учащихся старших классов, не знакомых или мало знакомых со свободным программным обеспечением, в отношении использования проприетарного ПО является желание найти в сети Интернет «взломанную» версию интересующей их программы, установить ее на свой компьютер и использовать для решения стоящих перед ними поль-

зовательских задач. К сожалению, данный сомнительный подход часто не вызывает у них даже колебаний в плане его правовой и нравственной допустимости, что, в свою очередь, говорит о невысоком уровне как информационной, так и духовно-нравственной культуры.

Возникает вопрос: а можно ли рассматривать стремление к соблюдению закона в области авторского права действием внутренне свободного человека? На этот вопрос прекрасно отвечает известный советский педагог Василий Александрович Сухомлинский: «Умей подчиняться закону, дисциплине, порядку... Умение подчиняться <...> законам — высшее выражение свободы» [1].

Таким образом, **вторым очевидным преимуществом использования принципа свободы является, как производная, воспитание информационной культуры и этики учащихся, одна из основных составляющих которых — компетенция квалифицированного выбора типа ПО, используемого для решения конкретных пользовательских задач.**

В подтверждение правильности вышеприведенного вывода можно привести цитату известного немецкого философа и математика Готфрида Вильгельма Лейбница, который писал: «Моральная необходимость мало мешает свободе, ибо кто выбирает наилучшее, то он от этого не становится менее свободен; наоборот, самая совершенная свобода скорее состоит именно в том, чтобы ничто не мешало действовать наилучшим образом» [7].

Кроме того, здесь, с точки зрения авторов, вполне уместно вспомнить известную формулировку нидерландского философа Бенедикта Спинозы, определившего свободу как осознанную необходимость. В нашем случае мы можем говорить об осознанной учащимися необходимости сделать тот или иной оптимальный выбор в плане применяемого ими в своей учебной и досуговой деятельности свободного программного обеспечения как проявление сформированной у них ментальной свободы.

Второе преимущество приобретает особую актуальность в ракурсе заявленного в федеральных государственных образовательных стандартах системно-деятельностного подхода. Системно-деятельностный подход пришел на смену ранее применявшемуся в отечественном образовании знаниевому подходу и качественно отличается от последнего тем, что во главу угла ставится цель научить школьников учиться, самостоятельно добывать и усваивать знания, а также сформировать у них ряд компетенций, необходимых для успешной социализации в современном динамично развивающемся информационном обществе.

В ряду формируемых у школьников в процессе их обучения в школе ИКТ-компетенций надо выделить компетенцию квалифицированного выбора оптимального для решения определенных пользовательских задач типа программного обеспечения, что подразумевает понимание принципиальной разницы между свободным и проприетарным ПО и начальные умения ориентироваться на рынке программного обеспечения.

Причем, с точки зрения авторов, вышеуказанная ИКТ-компетенция является первичной по отношению к пользовательским компетенциям, таким как, например, набор, редактирование и форматирование текста в текстовом редакторе. Школьник, владеющий навыками редактирования и форматирования текста в некотором определенном текстовом редакторе, но не знающий альтернативных типов текстовых редакторов,

их недостатков и преимуществ, не понимающий причин, по которым учителем ему предложен для работы именно данный программный инструмент, вряд ли на следующих ступенях школьного образования сможет стать действительно компетентным в области информационно-коммуникационных технологий.

Можно сказать, что данный негативный результат будет обусловлен применением именно явно устаревшего знаниевого подхода, предполагающего передачу учащимся некоторого объема знаний в области определенного текстового редактора и технологии работы в нем и не учитывающего тот факт, что рынок программного обеспечения является очень быстро развивающимся и к моменту окончания школьником обучения в общеобразовательной организации изучаемый им редактор может или вообще исчезнуть, или измениться до неузнаваемости, или же стать совершенно неприемлемым для него в ценовом плане.

Последний фактор, с точки зрения авторов, приобрел крайнюю актуальность в последние годы и выражается в следующем: учащийся привыкает в образовательной организации работать на лицензионном проприетарном программном обеспечении (в случае его применения в учебном процессе данной образовательной организации). В данном случае лицензия на использование программного обеспечения оплачивается учредителем образовательной организации или ею самой (в случае, если в ее бюджете предусмотрены средства на приобретение лицензий на проприетарное ПО). После окончания учащимся школы у него сформирована устойчивая привычка работать на том ПО, которое использовалось в ней, но платить за лицензию теперь он должен уже самостоятельно. Таким образом, в случае несформированности компетенции квалифицированного выбора типа ПО у школьников мы создаем потенциальных потребителей проприетарного ПО (причем чаще всего иностранного производства) или же нарушителей авторского права, что явно не отвечает национальным интересам России в информационной сфере и противоречит целям общего образования.

Для решения рассматриваемой проблемы важно сформировать у учащихся ясное понимание зависимости достижения целей от собственного выбора способа решения данной задачи и научить совершать этот выбор компетентно, квалифицированно. Учащийся должен видеть все плюсы и минусы выбираемого им решения, осознавать правовую и нравственную ответственность. Когда свободный человек совершает социальный выбор, то он принимает и всю ответственность за этот выбор на себя, но остается свободным в нравственно-духовном плане.

В этом смысле свобода и ответственность всегда являются двумя сторонами одной медали и не могут существовать отдельно друг от друга. Безусловно, выбор требует от учащегося определенной смелости, решимости, квалификации, но, как писал великий русский философ Николай Александрович Бердяев: «Человек раб потому, что свобода трудна, рабство же легко» [2].

В противном случае в духовно-нравственной сфере учащийся получает некую скрытую деформацию или повреждение, связанное с навязанным ему стереотипом действий. Как справедливо отмечает М. И. Рожков, «свободный человек — это человек, обладающий внутренней свободой, который осознает себя хозяином своей судьбы и поступает в соответствии с принятыми им принципами жизни» [12].

Указанная деформация говорит о наличии у индивидуума внутренней несвободы, что может привести впоследствии не к формированию творческой, самостоятельной личности, готовой к дальнейшей социализации в условиях современного, динамично развивающегося информационного общества, а к формированию человека, способного лишь к механическому копированию некоего усвоенного им ранее алгоритма, причем в совершенно различных сферах жизни.

Суммируя все вышесказанное, можно сформулировать **третье преимущество использования свободного программного обеспечения в учебном процессе в духовно-нравственном плане следующим образом: данный подход раскрывает творческие способности личности учащегося путем освобождения его от скрытых духовных деформаций, обусловленных наличием навязанных и многократно закрепленных неоптимальных алгоритмов действий в тех или иных ситуациях.**

Список использованных источников

1. Афоризмы Василия Сухомлинского. пишем.рф/author/Vasilii_Suhomlinskii/?curPos=40
2. Бердяев Н. О рабстве и свободе человека. М.: АСТ; Хранитель, 2000.
3. Богомолова О. Б. Преподавание ИКТ на базе свободного программного обеспечения: методическое пособие. М.: БИНОМ. Лаборатория знаний, 2013.
4. Георг Гегель. Викицитатник. https://ru.wikiquote.org/wiki/Георг_Гегель
5. Доктрина информационной безопасности Российской Федерации (Утверждена Указом Президента Российской Федерации от 5 декабря 2016 года № 646). <https://rg.ru/2016/12/06/doktrina-infobezobasnost-site-dok.html>
6. Духовно-нравственное воспитание // Википедия. [https://ru.wikipedia.org/wiki/Духовно-нравственное воспитание](https://ru.wikipedia.org/wiki/Духовно-нравственное_воспитание)
7. Ермишин О. Афоризмы. <https://info.wikireading.ru/101837>
8. Конституция Российской Федерации. <http://constitution.garant.ru/rf/chapter/741609f9002bd54a24e5c49cb5af953b/>;
9. Маркушевич М. В. Проблема выбора между платным и свободным программным обеспечением для использования его в учебном процессе // Информатика и образование. 2016. № 10.
10. Маркушевич М. В., Низаметдинова И. А. Преимущества применения свободного кроссплатформенного программного обеспечения в учебном процессе начальной школы // Герценовские чтения. Начальное образование. 2016. Т. 7. № 2.
11. Маркушевич М. В., Погосова О. В. Духовно-нравственное воспитание младших школьников через использование свободного программного обеспечения в учебно-воспитательном процессе начальной школы // Психология и педагогика XXI века: теория, практика и перспективы: Материалы VI Международной научно-практической конференции (г. Чебоксары, 17 сентября 2017 года). Чебоксары: ЦНС «Интерактив плюс», 2017.
12. Рожков М. И. Свобода и воспитание // Ярославский педагогический вестник. 2016. № 4.
13. Сабитова Г. М. Проблемы внедрения пакета свободного программного обеспечения в общеобразовательных учреждениях // Информатика и образование. 2011. № 4.
14. Свобода // Википедия. <https://ru.wikipedia.org/wiki/Свобода>
15. Свобода, инновации и удобство: интервью с Ричардом Столлманом. <http://tigran.am/stallman.htm>
16. Федосов А. Ю., Маркушевич М. В. Актуальные вопросы применения учителем информатики свободного программного обеспечения в учебно-воспитательном процессе общеобразовательной школы // Современные информационные технологии и ИТ-образование: Сборник научных трудов XI Международной научно-практической конференции. М.: МГУ, 2016.
17. Что такое свободная программа? <http://www.gnu.org/philosophy/free-sw.ru.html>

А. А. Оленев, В. В. Малиатаки,
 Ставропольский государственный педагогический институт

ИСТОРИЧЕСКИЕ ШИФРЫ И MAPLE*

Аннотация

В статье рассмотрены особенности использования системы компьютерной алгебры Maple для демонстрации работы алгоритмов шифрования в процессе изучения защиты информации в школьном курсе информатики. Приведены программы, позволяющие производить шифрование и расшифрование текстов с использованием различных исторических шифров.

Ключевые слова: шифрование, расшифрование текста, Maple.

Контактная информация

Оленев Александр Анатольевич,
 канд. тех. наук, доцент, доцент кафедры математики и информатики Ставропольского государственного педагогического института;
 адрес: 355029, г. Ставрополь, ул. Ленина, д. 417 «А»; телефон: (8652) 56-08-21; e-mail: olenevalexandr@gmail.com

Малиатаки Виктория Викторовна,
 канд. пед. наук, ст. преподаватель кафедры математики и информатики Ставропольского государственного педагогического института;
 адрес: 355029, г. Ставрополь, ул. Ленина, д. 417 «А»; телефон: (8652) 56-08-21; e-mail: maliataki@yandex.ru

A. A. Olenov, V. V. Maliataki,
 Stavropol State Pedagogical Institute

THE DEVELOPMENT OF ALGORITHMIC THINKING OF STUDENTS ON THE BASIS OF MENTAL-EXPERIENTIAL TRIT-TASK

Abstract

The article discusses the features of using the computer algebra system Maple for demonstrating the operation of encryption algorithms in the process of studying the protection of information in the school informatics course. The programs that allow encryption and decryption of texts using various historical ciphers are given.

Keywords: encryption, decryption of text, Maple.

Использование криптографии в современных цифровых технологиях актуально для многих сфер жизни нашего общества. Человечество применяет шифрование с того момента, как появилась конфиденциальная информация, содержание которой нужно было скрыть от злоумышленников. В настоящее время этот процесс становится все более и более масштабным, обуславливая актуальность вопросов сохранности данных, передаваемых с помощью информационных технологий. В первую очередь для передачи конфиденциальной информации по незащищенным каналам связи применяется шифрование, с использованием которого можно шифровать любую информацию — тексты, рисунки, звук, базы данных и т. д. [6, 7].

В школьном курсе информатики защита информации рассматривается преимущественно с этической и правовой сторон, что находит свое отражение и в федеральных государственных образовательных стандартах общего образования. Так, в ФГОС среднего общего образования в составе метапредметных результатов выделено умение «использовать средства информационных и коммуникационных технологий с соблюдением требований <...> правовых и этических норм, норм информационной безопасности» [12], а в предметных результатах по информатике — «принятие этических аспектов информационных технологий; осознание ответственности людей, вовлеченных в создание и использование информационных систем, распространение информации» [там же].

В школьных учебниках вопросы из области криптографической защиты информации представлены тоже довольно поверхностно. На основе анализа учебников можно сделать вывод о том, что данная тема относительно полно представлена только в учебнике К. Ю. Полякова, Е. А. Еремина «Информатика. 10 класс. Базовый и углубленный уровни» [10] в составе темы «Информационная безопасность» (шифрование; хэширование и пароли; современные алгоритмы шифрования; алгоритм RSA; электронная цифровая подпись; стеганография; шифрование данных). В остальных учебниках изучение данной темы происходит преимущественно в рамках раздела «Социальная информатика», что, на наш взгляд, не позволяет обучающимся получить достаточный объем сведений по криптографическим основам защиты информации. Сложность изучения рассматриваемой темы обусловлена также ее достаточно абстрактным представлением в учебниках, сведением в основном к теоретическим знаниям. Тогда как успешность и глубина ее изучения возросла бы, будь у школьников возможность поработать с некоторыми алгоритмами шифрования [1, 11].

Достаточно удобным инструментом для демонстрации работы алгоритмов криптографического преобразования является система компьютерной математики Maple, в частности, использование ее библиотеки StringTools [6, 7, 9].

В данной статье проиллюстрировано применение Maple на уроках информатики для демонстрации работы двух видов шифров, которые использовались еще в древние времена, — шифра замены и шифра перестановки.

В связи с тем что составными элементами шифра являются:

- алфавиты для записи исходных сообщений (защищаемой информации, открытого текста) и зашифрованных сообщений (шифротекстов, шифрограмм, криптограмм);
- алгоритмы криптографического преобразования (зашифрования и дешифрования);
- множество ключей,

нам необходимо обеспечить реализацию указанных компонентов средствами Maple.

* Материалы к статье можно скачать на сайте ИНФО:
http://infojournal.ru/journals/school/school_04-2018/

Шифр замены

Сущность шифрования методом замены заключается в следующем. Пусть шифруются сообщения на русском языке и замене подлежит каждая буква этих сообщений. Тогда букве А исходного алфавита сопоставляется некоторое множество символов (шифрозамен) M_A , букве Б — M_B , ..., букве Я — M_Y . Шифрозамены выбираются таким образом, чтобы любые два множества (M_i и M_j , $i \neq j$) не содержали одинаковых элементов ($M_i \cap M_j = \emptyset$).

Таблица, приведенная ниже, является ключом шифра замены. Зная ее, можно осуществить как шифрование, так и расшифрование:

А	Б	...	Я
M_A	M_B	...	M_Y

При шифровании каждая буква А открытого сообщения заменяется любым символом из множества M_A . Если в сообщении содержится несколько букв А, то каждая из них заменяется на любой символ из M_A . За счет этого с помощью одного ключа можно получить различные варианты шифрограммы для одного и того же открытого сообщения. Так как множества M_A , M_B , ..., M_Y попарно не пересекаются, то по каждому символу шифрограммы можно однозначно определить, какому множеству он принадлежит и, следовательно, какую букву открытого сообщения он заменяет. Поэтому расшифрование возможно и открытое сообщение определяется единственным образом [4].

В системе компьютерной алгебры Maple эту задачу можно решить, используя возможности библиотеки StringTools.

Зададим исходный и криптоалфавит:

```
> restart;
> with(StringTools):
```

Процедуры шифрования и дешифрования тогда можно представить следующим образом:

```
> Алфавит := "абвгдеёжзийклмнопрстуфхцчшщъыьэюя";
Криптоалфавит := "АБВГДЕЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ";
```

```
Алфавит := "абвгдеёжзийклмнопрстуфхцчшщъыьэюя"
Криптоалфавит :=
"АБВГДЕЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"
```

```
> Шифрованиезаменой := сообщение ->
CharacterMap(Алфавит, Криптоалфавит, сообщение):
Дешифрованиезаменой := код ->
CharacterMap(Криптоалфавит, Алфавит, код):
```

Пример использования:

```
> Шифрованиезаменой("пожалуйста, не читайте это
сообщение");
```

"ПОЖАЛУЙСТА, НЕ ЧИТАЙТЕ ЭТО СООБЩЕНИЕ"

```
> Дешифрованиезаменой("ПОЖАЛУЙСТА, НЕ ЧИТАЙТЕ
ЭТО СООБЩЕНИЕ");
```

"пожалуйста, не читайте это сообщение"

Итог выполнения шифрования и дешифрования — замена строчных букв кириллицы на прописные и обратно.

Для более наглядного изображения можно использовать интерактивное представление процедуры замены [1, 2] — см. рисунки 1, 2.

```
> restart;
> with(StringTools):
> with(Maplets[Elements]):
> заменаmaplet := Maplet(
>   Window('title' = "Шифр Замены",
>   BoxColumn('halign' = 'none', 'background' = 'red',
>   BoxRow('halign' = 'none',
>     Label("Введите исходный алфавит", Font("arial", 17)),
>     TextBox['TF1']('font' = Font("arial", bold, 16), 2..35)
>   ),
>   BoxRow('halign' = 'none',
>     Label("Введите шифроалфавит: m = ", Font("arial", 17)),
>     TextBox['TF2']('background' = 'white', 2..35, 'font' = Font("arial", bold, 16))
>   ),
>   BoxRow('halign' = 'none',
>     Label("Введите сообщение: m = ", Font("arial", 17)),
>     TextBox['TF3']('background' = 'white', 2..35, 'font' = Font("arial", bold, 16))
>   ),
>   BoxRow('halign' = 'none',
>     Button['B1']("Зашифровать", 'foreground' = 'black', 'font' = Font("arial", bold, 15),
>       Evaluate('TF4' = 'CharacterMap(TF1, TF2, TF3)'),
>       TextBox['TF4']('editable' = 'false', 'background' = 'white', 'font' = Font("arial", bold, 16),
>         2..35),
>     Button['B2']("Расшифровать", 'foreground' = 'black', 'font' = Font("arial", bold, 15),
>       Evaluate('TF4' = 'CharacterMap(TF2, TF1, TF3)'),
>   HorizontalGlue()
>   ),
>   BoxRow('halign' = 'center', 'background' = 'navy',
>     Button("Очистить все", 'font' = Font("arial", bold, 15),
>       Action(SetOption('TF1' = ""), SetOption('TF2' = ""),
>       SetOption('TF3' = ""), SetOption('TF4' = ""))),
>     Button("OK", 'font' = Font("arial", bold, 15), Shutdown())
>   ))):
> Maplets[Display](заменаmaplet):
```

Рис. 1. Maplet-код для шифра замены

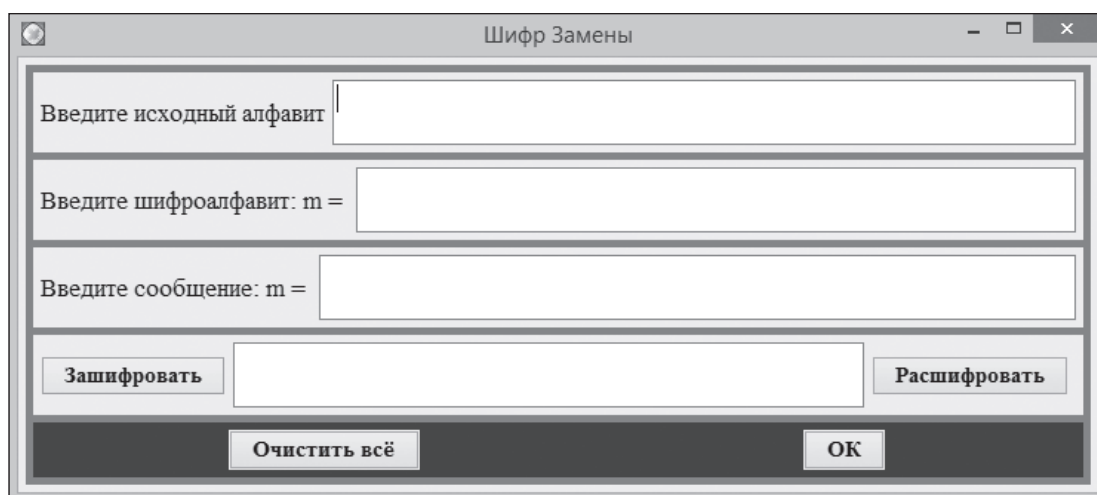


Рис. 2. Вид Maplets для шифра замены

Шифр Цезаря

Наиболее древним и распространенным примером шифра замены является шифр Цезаря. Гай Юлий Цезарь использовал в своей переписке шифр собственного

изобретения. Суть шифра Цезаря состоит в том, что буквы исходного алфавита заменяются буквами того же алфавита, но со сдвигом вправо на три позиции. Пример ключа применительно к русскому языку можно представить в следующем виде:

```
> restart;
> with(Maplets[Elements]):with(StringTools):
АЛФАВИТ := convert([192..223], bytes):
ШИФР:= ПИСЬМО -> SearchText(ПИСЬМО,АЛФАВИТ)-1:
буква := i -> АЛФАВИТ[i+1]:
МОДУЛЬ:= Length(АЛФАВИТ):
сдвиг := proc(КЛЮЧ);
i -> (i+КЛЮЧ) mod МОДУЛЬ;
end:
ЦезарьШифр:= (КЛЮЧ,сообщение) ->
Implode(map(буква,map(сдвиг(КЛЮЧ),map(ШИФР,Explode(сообщение))))):
ЦезарьДешифр:= (КЛЮЧ,сообщение) -> ЦезарьШифр(-КЛЮЧ,сообщение):
> ШифрЦезарь:= Maplet(
> Window('title'='Шифр цезаря',
> BoxColumn('halign' = 'none', 'background' = 'green',
> BoxRow('halign' = 'none',
Label("Введите ключ (сдвиг): b = ", Font("arial", 17)),
TextBox['TF1']('font' = Font("arial", bold, 16))
),
> BoxRow('halign' = 'none',
Label("Введите сообщение:", Font("arial", 17)),
TextField['TF2']('background' = 'white', 'font' = Font("arial", bold, 16))
),
> BoxRow('halign' = 'none',
Button['B1']("Зашифровать", 'foreground' = 'black', 'font' = Font("arial", bold, 15),
Evaluate('TF3' = 'ЦезарьШифр(TF1, TF2)' ) ),
TextBox['TF3']('editable' = 'false', 'background' = 'white',
'font' = Font("arial", bold, 16), 2..35),
> HorizontalGlue()
),
> BoxRow('halign' = 'none',
Button['B2']("Расшифровать", 'foreground' = 'black', 'font' = Font("arial", bold, 15),
Evaluate('TF4' = 'ЦезарьДешифр(TF1, TF3)' ) ),
TextBox['TF4']('editable' = 'false', 'background' = 'white',
'font' = Font("arial", bold, 16), 2..35),
> HorizontalGlue()
),
> BoxRow('halign' = 'center', 'background' = 'navy',
Button( "Вырезать все", 'font' = Font("arial", bold, 15),
Action(SetOption('TF1' = "" ), SetOption('TF2' = "" ),
SetOption('TF3' = "" ),SetOption('TF4' = "" )) ),
Button("Выход", 'font' = Font("arial", bold, 15), Shutdown())
),
> Maplets[Display](ШифрЦезарь):
```

Рис. 3. Maplet-код для шифра Цезаря

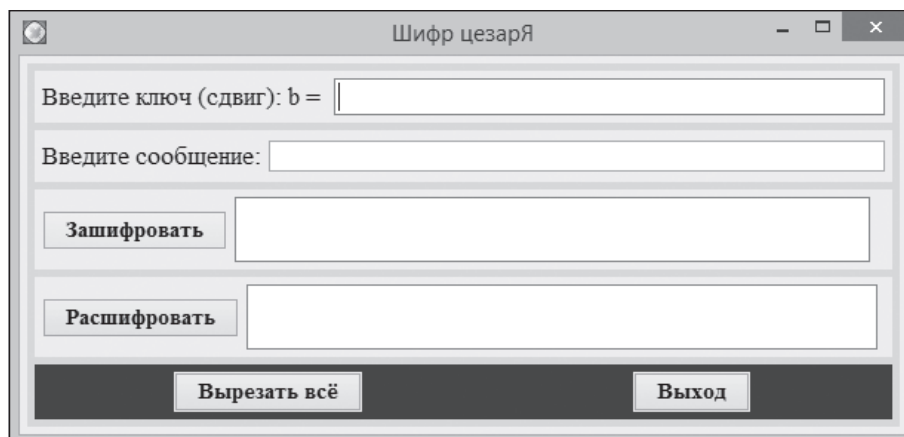


Рис. 4. Вид Maplets для шифра Цезаря

открытый алфавит:

А Б В Г Д Е Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш
Щ Ъ Ь Ъ Э Ю Я

криптоалфавит:

Г Д Е Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ
Ь Ъ Э Ю Я А Б В

То есть при зашифровывании буква А заменяется буквой Г, буква Б — буквой Д, буква В буквой — Е и т. д.

Процедуры шифрования и дешифрования шифром Цезаря тогда можно представить следующим образом.

В качестве открытого алфавита используем прописные буквы кириллицы:

```
> with(StringTools):
> АЛФАВИТ := convert([$192..223], bytes);
```

АЛФАВИТ :=

"АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЬЪЭЮЯ"

Для организации шифрования и дешифрования каждой прописной буквы с заданным ключом используем следующие процедуры:

```
> ШИФР := ПИСЬМО -> SearchText(ПИСЬМО, АЛФАВИТ) - 1:
буква := i -> АЛФАВИТ[i+1]:
> МОДУЛЬ := Length(АЛФАВИТ):
сдвиг := proc(КЛЮЧ);
i -> (i+КЛЮЧ) mod МОДУЛЬ;
end:
> ЦезарьШифр := (КЛЮЧ, сообщение) ->
Implode(map(буква, map(сдвиг(КЛЮЧ),
map(ШИФР, Explode(сообщение))))):
ЦезарьДешифр := (КЛЮЧ, сообщение) ->
ЦезарьШифр(-КЛЮЧ, сообщение):
```

Пример.

Шифрование и дешифрование шифром Цезаря с ключом —19 фразы «ПРИВЕТМИР»:

```
> с := ЦезарьШифр(19, "ПРИВЕТМИР");
с := "ВГХШЕЯЙГ"
> ЦезарьДешифр(19, с);
"ПРИВЕТМИР"
```

Для более наглядного изображения можно использовать интерактивное представление процедуры замены [3–5, 8] — см. рисунки 3, 4.

Количество ключей (сдвигов) для шифра Цезаря ограничено. Показанная ниже процедура позволяет простым перебором найти правильное решение (рас-

шифровать) предложенное сообщение. Процедура вывода в таблицу всех возможных вариантов расшифровки текста представлена ниже:

```
> ЦезарьКриптоанализ := шифртекст ->
table([seq(ЦезарьДешифр(k+1, шифртекст),
k=0..МОДУЛЬ-1)]):
```

Контрольные вопросы и задания.

1. Дайте определение шифра Цезаря.
2. Поясните принцип работы шифра Цезаря.
3. Какое количество ключей можно выбрать для шифра Цезаря и от чего это зависит?
4. Зашифруйте шифром Цезаря слово «Криптология», используя ключ +5.

Шифр перестановки

При шифровании перестановкой символы шифруемого текста переставляются по определенному правилу в пределах блока этого текста. Шифры перестановки являются самыми простыми и, наверно, самыми древними шифрами.

Пример.

Выберем целое положительное число, например 4. Расположим числа от 1 до 4 в верхней строке по порядку, а в нижней строке расположим эти числа в произвольном порядке, так, как представлено в таблице:

1	2	3	4
4	3	2	1

Эта конструкция носит название *подстановки*.

Зашифруем фразу «ПРИВЕТ МИР». В этой фразе 9 знаков (9 прописных букв). В нашем случае добавятся три буквы (например, У) до ближайшего кратного четырем числа. Теперь разбиваем фразу на три четырехзначные группы:

ПРИВ ЕТМИ РУУУ

Далее буквы каждой группы переставляем согласно конструкции подстановки, указанной в таблице: первая буква встает на четвертое место, вторая — на третье, третья — на второе, четвертая — на пятое.

Процедуры шифрования шифром перестановки можно представить следующим образом:

```

> restart;
> Перестановка:= proc(открТекст::string,
  n::integer)
local tempSet, pList, temp, randGen, i, numRows,
  strLen, j, шифр:
randGen := rand(n):
tempSet := {}: pList := []:
for i from 1 to n do
while(nops(tempSet) <> i) do
temp := randGen() + 1:
tempSet := {op(tempSet), temp}:
end do:
pList := [op(pList), temp]:
end do:
strLen := length(открТекст):
if(strLen mod n = 0) then numRows := strLen / n:
else numRows := floor(strLen / n + 1):
end if:
шифр := "":
for i from 1 to n do
temp := pList[i]:
for j from 1 to numRows do
if(temp > strLen) then
шифр := cat(шифр, "У"):
else
шифр := cat(шифр, открТекст[temp]):
end if:

```

```

temp := temp + n:
end do:
end do:
[шифр, pList]:
end proc:

```

С целью упрощения порядка шифрования формирование ключа выполняется случайным образом.

```

> Перестановка("ПРИВЕТМИР", 4);
[ "ПЕРИМУРТУВИУ", [ 1, 3, 2, 4] ]

```

При расшифровывании текст разбивается на четырехзначные группы и буквы каждой группы переставляются в обратном порядке. Ключом шифра являются число 4 и порядок расположения чисел в нижнем ряду таблицы.

```

> ПерестановкаДешифр := proc(шифрТекст::string,
  ключ::list)
local rows, cols, ДешТекст, i, j, p;
ДешТекст := "";
cols := nops(ключ);
rows := length(шифрТекст)/cols;
for i from 1 to rows do
for j from 1 to cols do
member(j, ключ, 'p');

```

```

> restart;
> with(Maplets[Elements]):with(StringTools):
Перестановка:= proc(открТекст::string, n::integer)
. . .
end proc:
ПерестановкаДешифр := proc(шифрТекст::string, ключ::list)
. . .
end proc:
> ШифрПерестановки:= Maplet(
> Window('title'="Шифр перестановки",
> BoxColumn('halign' = 'none', 'background' = 'yellow',
> BoxRow('halign' = 'none',
  Label("Введите длину блока разбиения: n = ", Font("arial", 17)),
  TextBox['TF1']('font' = Font("arial", bold, 16))
),
> BoxRow('halign' = 'none',
  Label("Введите ключ (порядок перестановки): [,] = ", Font("arial", 17)),
  TextBox['TF5']('font' = Font("arial", bold, 16))
),
> BoxRow('halign' = 'none',
  Label("Введите сообщение:", Font("arial", 17)),
  TextField['TF2']('background' = 'white', 'font' = Font("arial", bold, 16))
),
> BoxRow('halign' = 'none',
  Button['B1']("Зашифровать", 'foreground' = 'black', 'font' = Font("arial", bold, 15),
  Evaluate('TF3' = 'Перестановка(TF2,TF1)' ) ),
  TextBox['TF3']('editable' = 'false', 'background' = 'white', 'font' = Font("arial", bold, 16),
  2..35 ),
> HorizontalGlue()
),
> BoxRow('halign' = 'none',
  Button['B2']("Расшифровать", 'foreground' = 'black', 'font' = Font("arial", bold, 15),
  Evaluate('TF4' = 'ПерестановкаДешифр(TF3[1],TF5)' ) ),
  TextBox['TF4']('editable' = 'false', 'background' = 'white', 'font' = Font("arial", bold, 16),
  2..35 ),
> HorizontalGlue()
),
> BoxRow('halign' = 'center', 'background' = 'navy',
  Button( "Вырезать все", 'font' = Font("arial", bold, 15),
  Action(SetOption('TF1' = "" ), SetOption('TF2' = "" ),
  SetOption('TF3' = "" ), SetOption('TF4' = "" ), SetOption('TF5' = "" ) ) ),
  Button( "Выход", 'font' = Font("arial", bold, 15), Shutdown() )
),
> Maplets[Display](ШифрПерестановки):

```

Рис. 5. Maplet-код для шифра перестановки

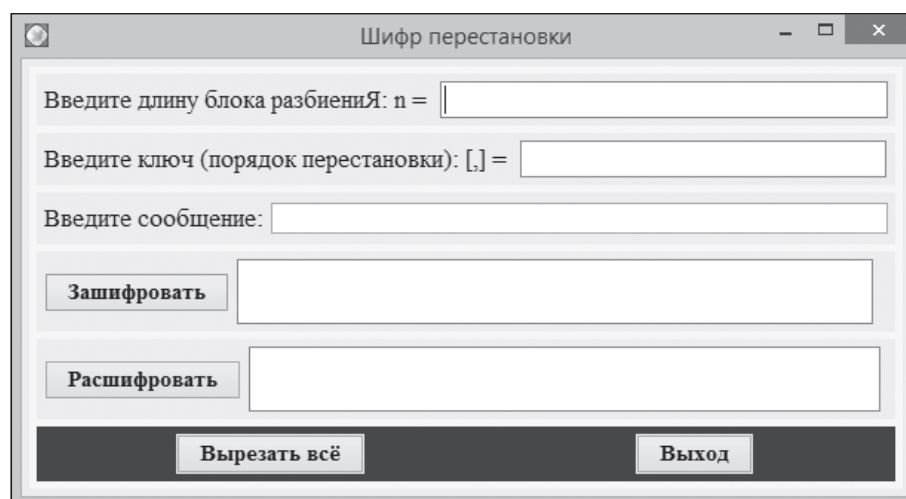


Рис. 6. Вид Maplets для шифра Цперестановки

```

ДешТекст := cat (ДешТекст, шифрТекст[rows*(p-1)+i]);
end do;
end do;
ДешТекст;
end proc;

```

Пример шифрования и дешифрования перестановкой текста «ПРИВЕТМИР»:

```

> e := Перестановка ("ПРИВЕТМИР", 4);
e := ["РТУИМУВИУПЕР", [2, 3, 4, 1]]
> ПерестановкаДешифр (e[1], e[2]);
"ПРИВЕТМИРУУУ"

```

Для более наглядного изображения можно использовать интерактивное представление процедуры перестановки [3–5, 8] — см. рисунки 5, 6.

Контрольные задания.

1. Поясните принцип работы шифра перестановки.
2. Зашифруйте заданную фразу шифром перестановки, используя произвольную степень.

* * *

Мы надеемся, что приведенные программы, позволяющие выполнять действия шифровки и дешифровки простых текстов, будут являться отправной точкой изучения более сложных алгоритмов криптографии. А пока на уроках информатики можно организовать игры: кто-то разрабатывает для переписки шифр, основанный на замене одних символов другими, а другие, «перехватывая» и анализируя эти сообщения, раскрывают «шифровки». Сколько понадобится для этого перехваченных сообщений и времени — это отдельный вопрос. Вернее, задание на модификацию и использование предложенных выше программ для раскрытия текста или ключа шифрования.

Список использованных источников

1. Андреева Е. В., Босова А. Ю., Фалина И. Н. Математические основы информатики. Элективный курс: учебное пособие: 2-е изд., испр. М.: БИНОМ. Лаборатория знаний, 2007.

2. Гатченко Н. А., Исаев А. С., Яковлев А. Д. Криптографическая защита информации. СПб.: НИУ ИТМО, 2012.

3. Говорухин В. И., Цибулин В. Г. Введение в Maple. Математический пакет для всех. М.: Мир, 1997.

4. Кирсанов М. Н. Maple 13 и Maple. Решение задач механики. М.: Физматлит, 2010.

5. Красильников В. В., Оленев А. А., Тоискин В. С. Проверка знаний по теме «множество» на основе системы компьютерной математики Maple // Актуальные вопросы инженерного образования — 2015: Сборник научных трудов международной научно-методической конференции (г. Октябрьский, 27 ноября 2015 года). Екатеринбург: Универсальная Типография «Альфа Принт», 2016.

6. Красильников В. В., Оленев А. А., Тоискин В. С., Тынчеров К. Т. Использование системы компьютерной алгебры Maple при изучении дискретной математики // Актуальные вопросы инженерного образования — 2016: Сборник научных трудов международной научно-методической конференции, посвященной 60-летию филиала УГНТУ в г. Октябрьском (г. Октябрьский, 11 ноября 2016 года). Уфа: УГНТУ, 2016.

7. Малиатаки В. В., Медведева Л. М., Оленев А. А. Совершенствование математической подготовки учителя в вузе на основе использования СКА Maple // Актуальные вопросы инженерного образования — 2015: Сборник научных трудов международной научно-методической конференции (г. Октябрьский, 27 ноября 2015 года). Екатеринбург: Универсальная Типография «Альфа Принт», 2016.

8. Оленев А. А., Красильников В. В., Малиатаки В. В., Тынчеров К. Т., Селиванова М. В. Использование системы компьютерной алгебры Maple для демонстрации классических криптосистем // Актуальные вопросы высшего профессионального образования: Сборник научных трудов Международной научно-методической конференции. Уфа: УГНТУ, 2017.

9. Оленев А. А., Малиатаки В. В. Моделирование логических элементов и простейших узлов ЭВМ в системе компьютерной математики Maple // Информатика в школе. 2017. № 8.

10. Поляков К. Ю., Еремин Е. А. Информатика. 10 класс. Базовый и углубленный уровни: учебник. В 2 ч. Ч. 2. М.: БИНОМ. Лаборатория знаний, 2016.

11. Самылкина Н. Н., Калинин И. А. Информатика. 10–11 классы: методическое пособие. М.: БИНОМ. Лаборатория знаний, 2016.

12. Федеральный государственный образовательный стандарт среднего общего образования (в ред. Приказа Минобрнауки России от 29.12.2014 № 1645). http://минобрнауки.рф/документы/2365/файл/736/12.05.17-Приказ_413.pdf



И. В. Гаврилова,
Минусинский кадетский корпус, Красноярский край

РАЗВИТИЕ АЛГОРИТМИЧЕСКОГО МЫШЛЕНИЯ УЧАЩИХСЯ НА ОСНОВЕ МЕНТАЛЬНО-ЭМПИРИЧЕСКИХ ТРИТ-ЗАДАЧ

Аннотация

В статье обосновывается эффективность использования ментально-эмпирических трит-задач при изучении темы «Алгоритмизация» в школьном курсе информатики. Ведущая идея трит-задачи заключается в ее формулировке в чувственном, модельном (когнитивном) и понятийном видах. В статье представлена процессуальная структура мыслительных операций при составлении алгоритмов, приведены примеры трит-задач для изучения основных алгоритмических конструкций.

Ключевые слова: алгоритмическое мышление, ментальные карты, алгоритмизация, школьный курс информатики.

Контактная информация

Гаврилова Ирина Викторовна, учитель информатики Минусинского кадетского корпуса, Красноярский край; адрес: 662606, Красноярский край, г. Минусинск, ул. Народная, д. 80; телефон: (391) 324-04-72; e-mail: gavrilowa@yandex.ru

I. V. Gavrilova
Minusinsk Cadet Corps, Krasnoyarsk Region

THE DEVELOPMENT OF ALGORITHMIC THINKING OF STUDENTS ON THE BASIS OF MENTAL-EMPIRICAL TRIT-TASKS

Abstract

The article substantiates the effectiveness of using mental-empirical trit-tasks in the study of the theme "Algorithmization" in the school course of informatics. The leading idea of the trit-task lies in its formulation in the sensory, model (cognitive) and conceptual forms. The article presents the procedural structure of thought operations in the compilation of algorithms, gives examples of trit-tasks for the study of basic algorithmic constructions.

Keywords: algorithmic thinking, mental schemes, algorithmization, school course of informatics.

Краевой мониторинг результатов единого государственного экзамена (ЕГЭ) по информатике свидетельствует о том, что с заданиями на знание основных конструкций языка программирования справляются 78 % выпускников, а с заданиями, связанными с анализом и конструированием нелинейного алгоритма, — всего около 30 % [3]. Опираясь на опыт школьных учителей, рекомендации для педагогов по результатам ЕГЭ, можно констатировать, что основная проблема выпускников связана не с освоением языков программирования, а с неумением создавать алгоритмы.

Для повышения эффективности обучения школьников разработке различных алгоритмов мы предлагаем использовать методику изучения алгоритмических структур с опорой на когнитивный подход.

М. Б. Зыков, Н. Р. Сабанина выделяют три вида интерпретации действительности, лежащие в основе мышления: инстинктивно-эмоциональная, понятийно-знаниевая, нерукотворно-природная (чувственно-ценностная) [1]. Они выделяют особенность современной российской образовательной системы — почти исключительно знаниевый характер. По их оценкам, от 95 % — в обычном классе или вузовской аудитории — до — во время подготовки и проведения ЕГЭ-экзаменов — 100 %. То есть из трех видов активного психического отражения используется только один — понятийно-знаниевый [1].

В статье [2] определяются три составляющие алгоритмического мышления — чувственная, модельная и понятийная, которые необходимо учитывать в процессе обучения для его успешности. Авторы статьи обосновывают возможность использования ментальных карт при обучении программированию как эффективного средства развития алгоритмического мышления. Ментальные карты и схемы — «удобная техника для представления процесса мышления или структурирования информации в визуальной форме, позволяющей человеку справиться с информационным потоком» [2].

Т. А. Степанова в своем пособии «Теория алгоритмического мышления» [4] делает попытку «формализовать основные положения теории мышления», уточняет, что, согласно информационной модели мышления, «мысль — это образ в памяти; мыслительный процесс — выстраивание цепочки образов; мышление — способ выстраивания таких цепочек. Цепочки могут выстраиваться различным способом: используя образы одной области памяти или нескольких, komponуя их в различных сочетаниях» (рис. 1).

В разрезе этой модели ассоциативное, интуитивное мышление — это выстраивание цепочек только из образов нижнего, чувственного уровня. Чем больше образов задействовано с верхних уровней, тем выше уровень развития алгоритмического мышления. Процесс развития мышления, как и процесс развития памяти, «происходит от простого к сложному, от сложного — к еще более сложному, т. е. области памяти заполняются постепенно» [4].

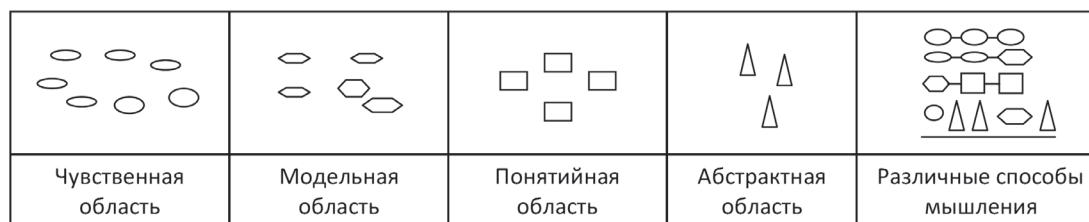


Рис. 1. Информационная модель памяти и способы мышления

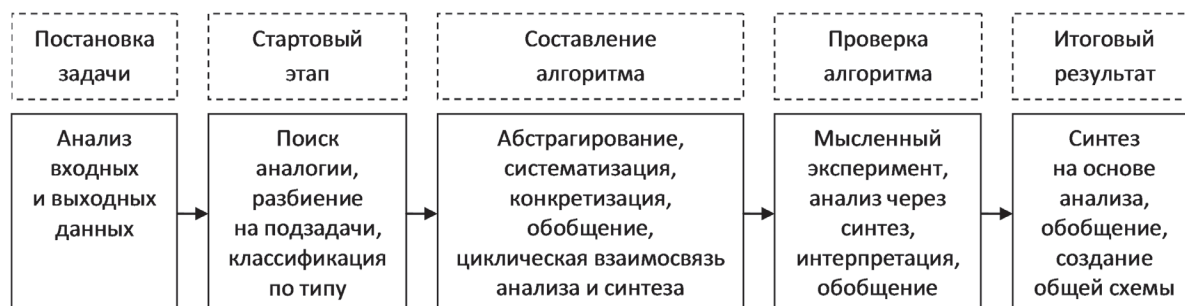


Рис. 2. Процессуальная структура мыслительных операций при составлении алгоритмов

Цепочку мыслительных операций при создании алгоритмов можно схематично представить, как на рисунке 2.

При анализе условия задачи учащийся должен определить, что можно узнать из уже имеющихся данных и какие новые сведения ему необходимы.

Успешность стартового этапа определяется запасом знаний и сформированными способами решения задач данного класса. Интеллектуальный поиск строится по опознавательным признакам, по сопоставлению с основными алгоритмическими структурами или известными алгоритмами.

Этап составления алгоритма в большей степени обусловлен уровнем сформированности алгоритмического мышления. Алгоритм в общем случае сначала должен быть представлен в форме, понятной создателю, т. е. человеку, который его разрабатывает, а затем уже представлен в форме, понятной исполнителю. Среди разнообразия технологий преподавания раздела «Алгоритмизация» следует выделить «отзадачный подход, реализуемый на основе практико-ориентированных «живых» задач» [4].

Учитывая вышесказанное, обоснованным и эффективным средством развития алгоритмического мышления школьников является **система ментально-эмпирических трит-задач**.

Термин «**трит-задача**» выбран потому, что трит — это наименьшая целая единица измерения количества информации источников с тремя равновероятными сообщениями. Трит-задача состоит из трех частей и опирается на три понятия — «жизненная ситуация», «ментальная карта», «блок-схема». Она задействует три области мышления и памяти — чувственную, понятийную и модельную. В трит-задаче ученики:

- 1) на основе визуализированной жизненной ситуации выделяют познавательную проблему;
- 2) составляют некую схему ее решения;
- 3) а затем конструируют блок-схему соответствующего алгоритма.

Данный способ изучения алгоритмических структур представляется эффективным за счет опоры на эмпирический опыт учащихся, чувственную область памяти. Пере-

ход к ментальной схеме развивает мышление за счет построения логических цепочек из различных областей памяти (см. рис. 1). Достаточно сложный для школьников абстрактный процесс составления блок-схемы облегчается именно за счет взаимосвязи с двумя предыдущими составляющими — жизненной ситуацией и моделью ее решения, представленной в виде ментальной схемы.

Трит-задача представляет собой карточку, состоящую из трех столбцов:

- в первом размещается графическое изображение, демонстрирующее алгоритмический процесс окружающей действительности;
- второй столбец предназначен для ментальной схемы;
- третий — для составления блок-схемы.

Данные карточки могут использоваться как в бумажном, так и в электронном виде. При этом электронный вариант использования трит-задач позволяет отрабатывать навыки работы обучающегося в графическом и текстовом редакторах.

Применение трит-задач при изучении алгоритмических структур предполагает три этапа:

- 1) совместное решение задачи учителем и учащимися;
- 2) самостоятельная или групповая работа по решению трит-задач;
- 3) самостоятельное составление учащимися аналогичных задач.

При совместном решении трит-задачи учитель заполняет карточку, разъясняя процесс составления ментальной схемы, задавая поясняющие вопросы обучающимся, проводит показательное решение задачи.

Затем, работая в парах, малых группах или самостоятельно, учащиеся решают подобные трит-задачи самостоятельно. Количество решаемых задач определяется уровнем усвоения материала учащимися.

Завершающий этап — составление учащимися аналогичных карточек трит-задач — позволяет ученикам выделить алгоритмические процессы в повседневной жизни.

На рисунках 3–9 приведены **примеры задач, представляющих разные типы алгоритмов**. Центральная и правая части представленных карточек заполнены учащимися.



Рис. 3. Линейный алгоритм

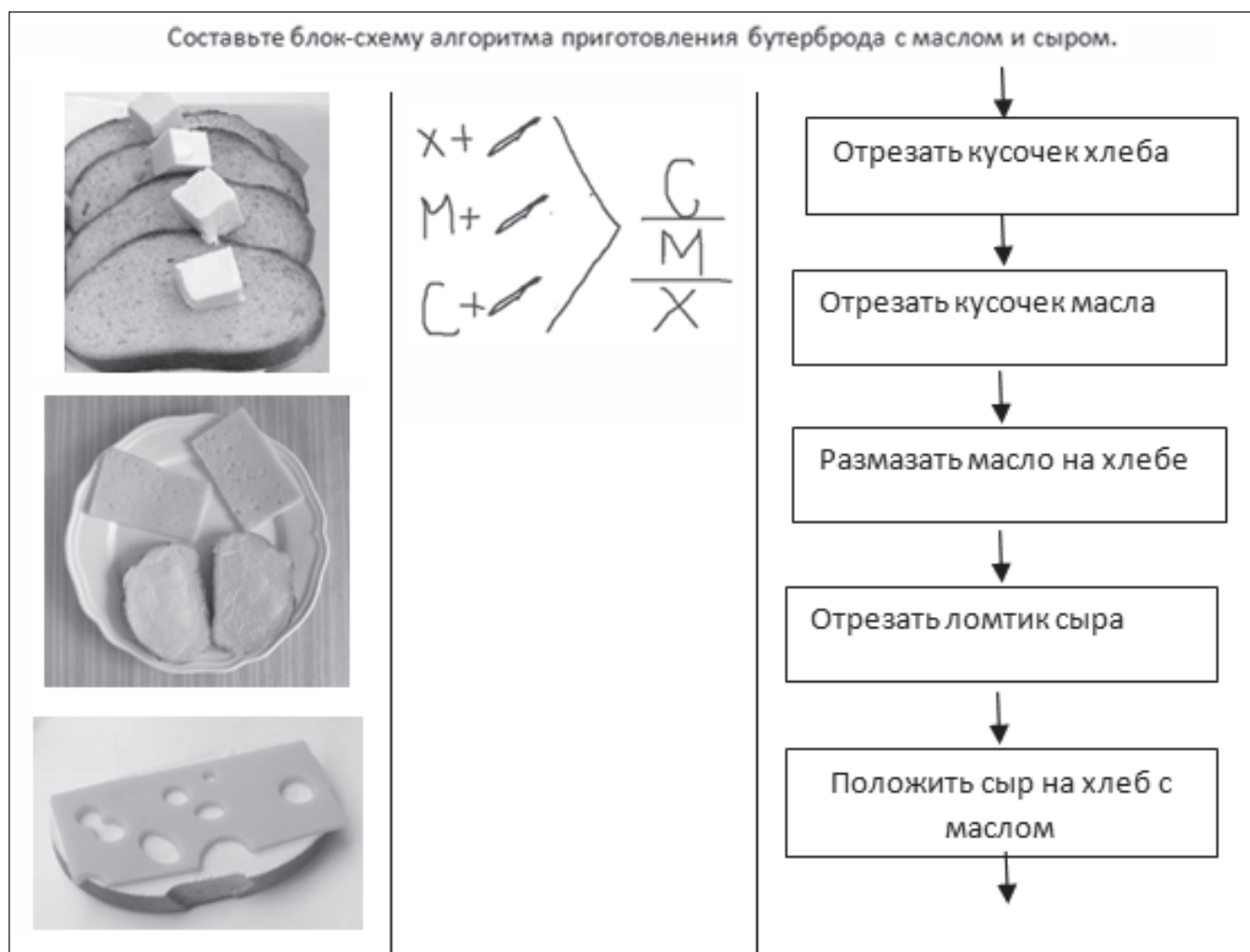


Рис. 4. Линейный алгоритм

Поздней осенью может пойти снег. После снегопада могут образоваться сугробы, лужи или лёд. Почему? Как узнать, что нас ждет после снегопада?

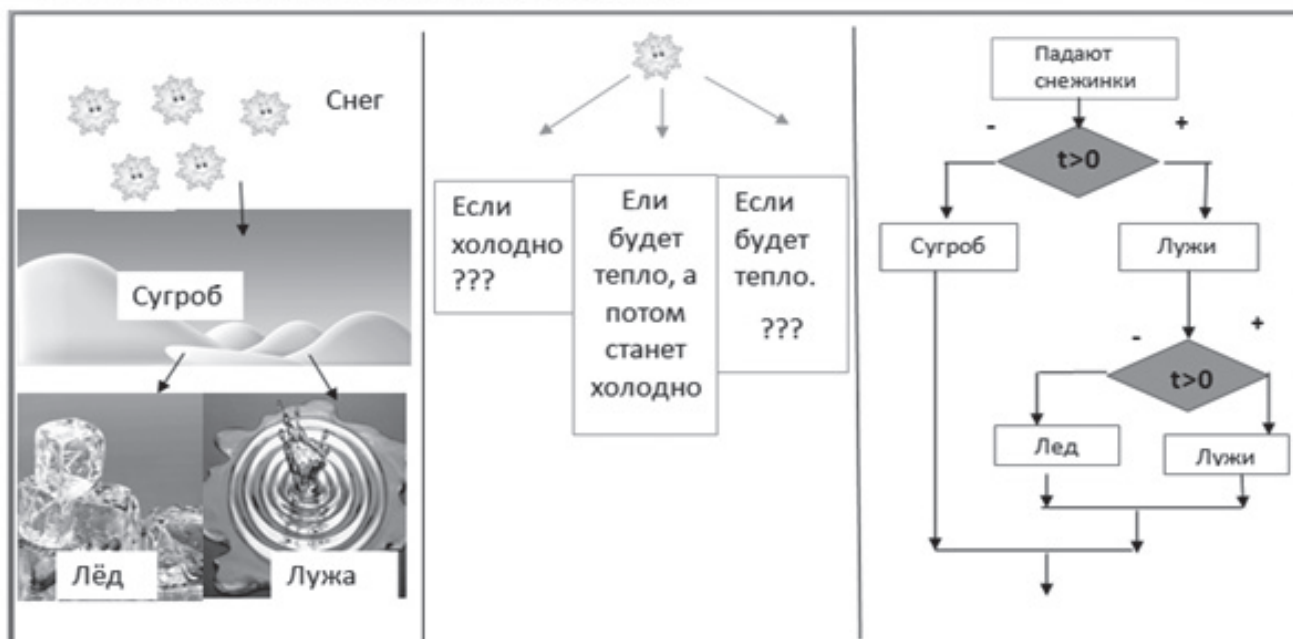


Рис. 5. Алгоритм с ветвлением

Составьте блок-схему алгоритма принятия решения: в выходной день, если будет хорошая погода, то вы с друзьями пойдете гулять в парк, а если будет дождь, то в кино.

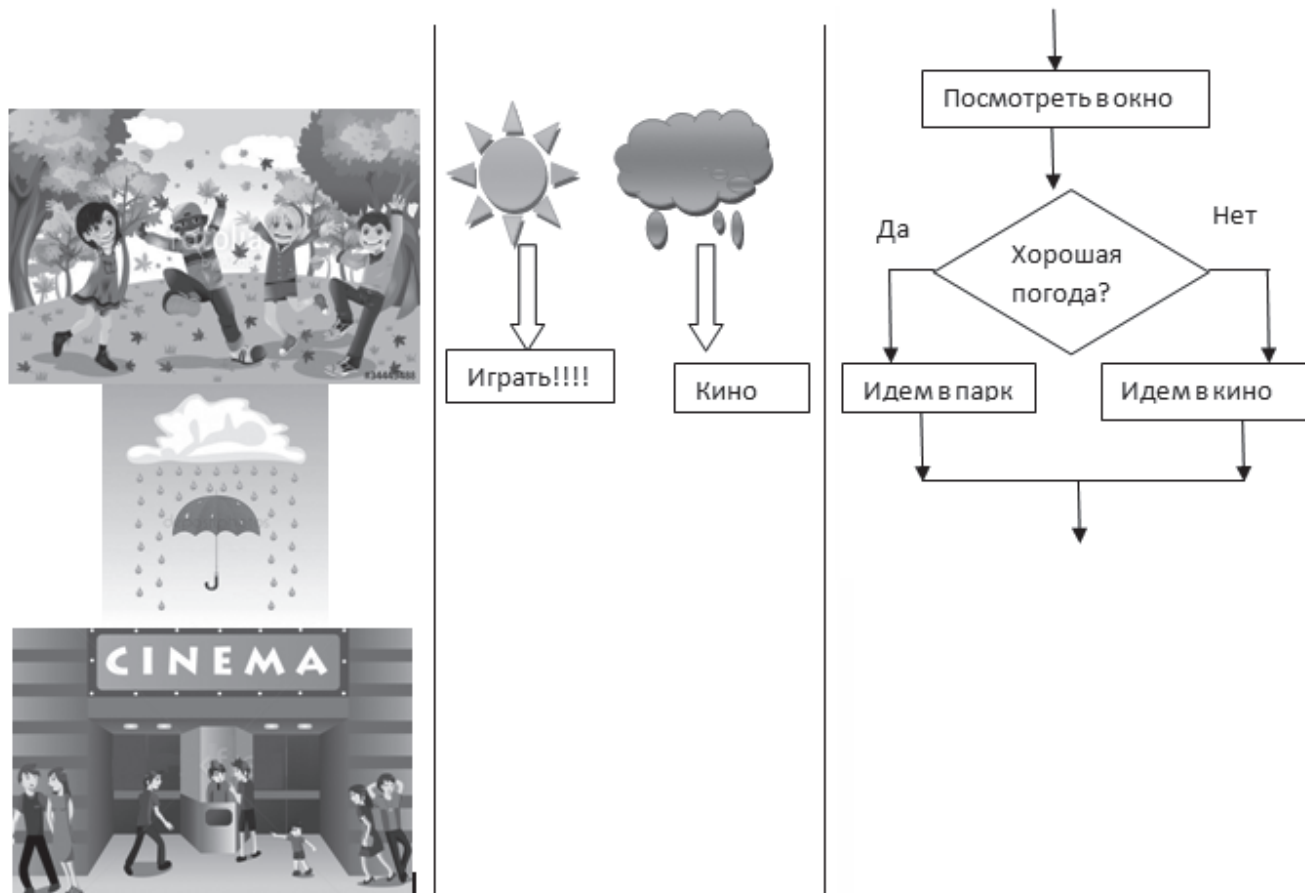


Рис. 6. Алгоритм с ветвлением

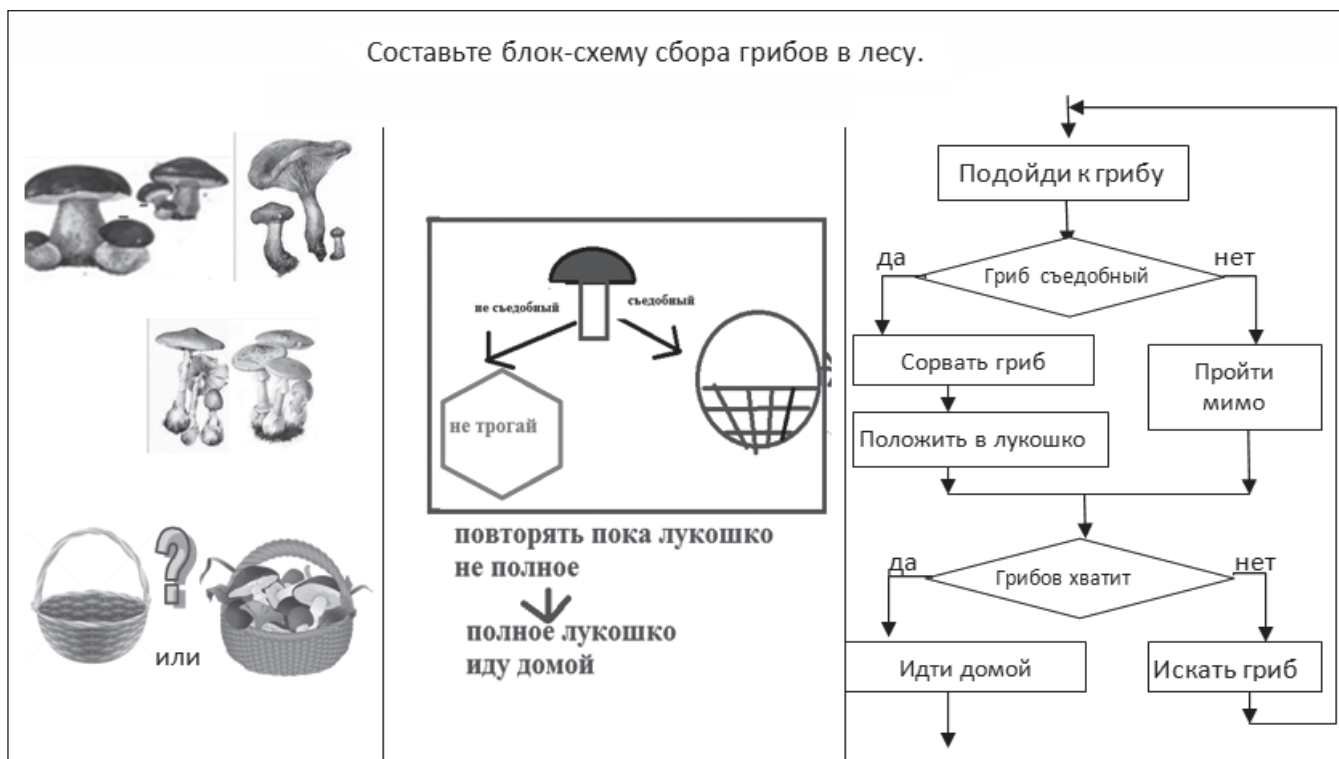


Рис. 7. Алгоритм с циклом и ветвлением

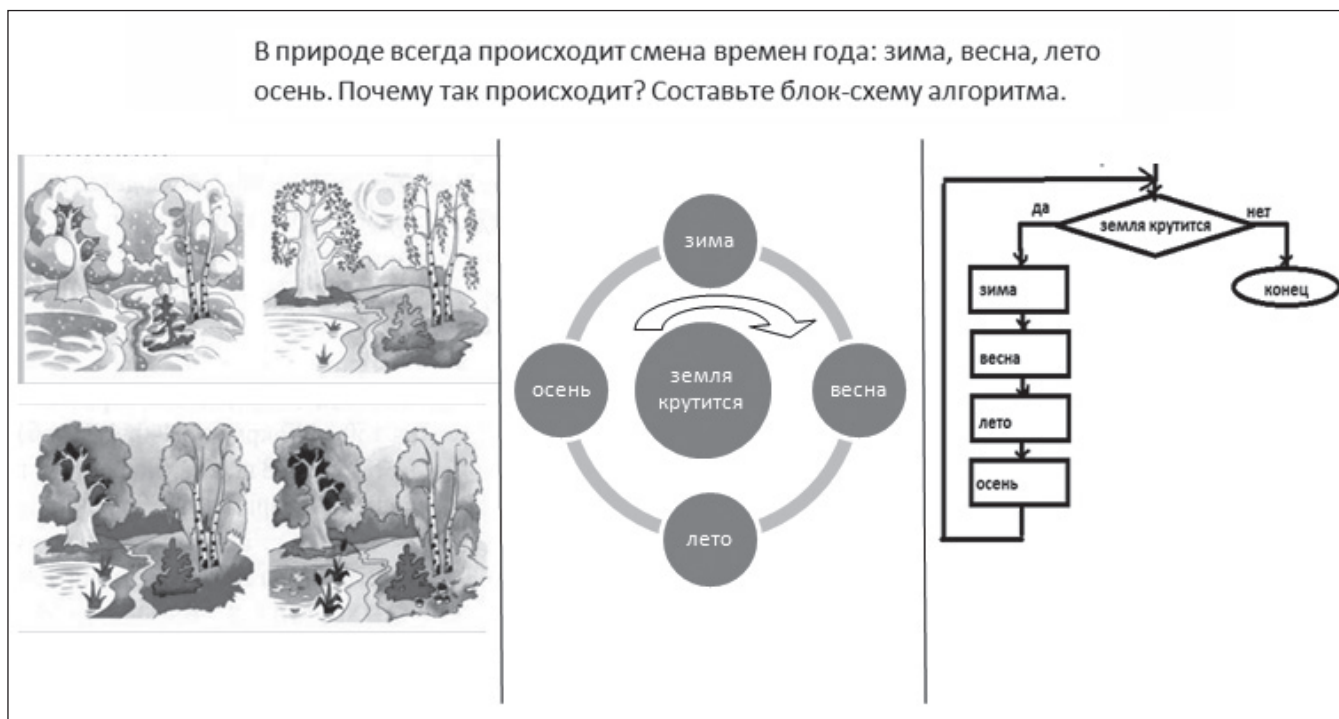


Рис. 8. Циклический алгоритм

Проведенная апробация предложенной методики показала ее эффективность — при использовании трит-задач учащиеся успешно конструировали алгоритмы различных типов. Необычная форма подачи материала, опора на эмпирический опыт учащихся позволили повысить мотивацию и вследствие этого прочность усвоения материала. Это подтверждается результатами проверочных работ.

Предложенный прием позволяет повысить уровень понимания и усвоения учебного материала по алго-

ритмизации, так как учитывает особенности процесса мышления. Промежуточный этап между постановкой задачи и конструированием блок-схемы алгоритма позволяет, опираясь на чувственную и образную области памяти, задействовать понятийную область и тем самым способствует простому и прочному усвоению понятий. Составление ментальной схемы дает некоторую свободу выражения мыслей в строго регламентированном разделе «Алгоритмизация и программирование», которая упро-



Рис. 9. Алгоритм сложной конструкции

щает переход от жизненной ситуации к абстрактному алгоритму. Процесс решения предложенных трит-задач соответствует описанной процессуальной структуре мыслительных операций при составлении алгоритмов.

Таким образом, представление традиционных алгоритмических задач в форме ментально-эмпирических позволяет более эффективно развивать алгоритмическое мышление обучаемого, прививает ему навыки осознанной разработки алгоритмов.

Список использованных источников

1. Зыков М. Б., Сабанина Н. Р. О необходимости строгого различия понятий гуманизации и гуманитаризации в школьном образовании // Личность, семья и общество: вопросы педагогики и психологии: Сборник статей по материалам

XXIX международной научно-практической конференции. Новосибирск: СибАК, 2013.

2. Калинина В. В., Пушкарева Т. П., Степанова Т. А. Алгоритмические ментальные карты как эффективное средство обучения программированию // Сборник статей Международной научно-практической конференции «Фундаментальные и прикладные научные исследования» (г. Москва, 17 мая 2015 года). М.: ООО «Европейский фонд инновационного развития», 2015.

3. Лецинер В. Р., Ройтберг М. А. Методические рекомендации для учителей, подготовленные на основе анализа типичных ошибок участников ЕГЭ 2016 года. <http://www.fipi.ru/sites/default/files/document/1472532815/informatika.pdf>

4. Степанова Т. А. Теория алгоритмического мышления: учебное пособие для магистрантов, учителей общеобразовательных учреждений, преподавателей вузов. Красноярск: КГПУ им. В. П. Астафьева, 2014.



А. С. Путина,

Московский педагогический государственный университет

SCRATCH-ОЛИМПИАДА ПО КРЕАТИВНОМУ ПРОГРАММИРОВАНИЮ

Аннотация

В статье говорится о важности программирования для участия в олимпиадах и для будущей профессиональной ориентации. Рассматриваются популярная среда программирования Scratch и олимпиада по созданию проектов с помощью данного языка программирования уже с начальной школы. Приведены примеры проектов, удовлетворяющих требованиям рассматриваемой олимпиады.

Ключевые слова: программирование, Scratch, спрайт, скрипт, олимпиада.

Контактная информация

Путина Анастасия Сергеевна, студентка 5-го курса бакалавриата математического факультета Московского педагогического государственного университета; *адрес:* 107140, г. Москва, ул. Краснопрудная, д. 14; *телефон:* (499) 264-25-56; *e-mail:* ak251192@yandex.ru

A. S. Putina,
Moscow Pedagogical State University

SCRATCH-OLYMPIAD ON CREATIVE PROGRAMMING

Abstract

The article talks about the importance of programming to participate in the Olympiad and for future professional orientation. Also the popular programming environment Scratch and the Olympiad to create projects using this programming language since primary school are discussed. The article presents examples of projects for this Olympiad.

Keywords: programming, Scratch, sprite, script, Olympiad.

Предметные олимпиады — это важная составляющая образовательного процесса, в том числе в обучении информатике. В старших классах олимпиада по информатике — это фактически олимпиада по программированию. Совершенно очевидно, что за один-два года подготовиться к такой олимпиаде очень сложно, а значит, подготовку необходимо начинать как можно раньше. Сегодня одним из наиболее популярных языков для раннего обучения программированию является Scratch. Изучая этот язык, школьники разного возраста, в том числе учащиеся начальной школы, могут не только освоить основы программирования, но и приобрести опыт участия в олимпиадах по программированию.

Компания «РОББО» в партнерстве с различными вузами, в том числе Московским педагогическим государственным университетом, начиная с 2017 года проводит международную Scratch-олимпиаду по программированию для учащихся I—XI классов. Творческие задания олимпиады различаются по сложности для разных возрастных категорий. Задания представляют собой описание сценария, который надо реализовать в виде проекта среды программирования Scratch 2.0 [1]. Так, в 2017 году участникам олимпиады были предложены следующие задания:

- I—II классы — создание анимированной истории на произвольную тему;
- III—IV классы — создание мультфильма по литературному произведению, изучаемому в рамках школьной программы;
- V—VI классы — создание викторины по любому предмету школьной программы;
- VII—VIII классы — создание игры;
- IX—XI классы — создание образовательного проекта по одному из школьных предметов.

Подробнее с темами и требованиями олимпиады можно познакомиться на сайте [1].

С 2018 года в олимпиаде возможно как индивидуальное, так и групповое участие. Для участия школьнику необходимо зарегистрироваться на сайте, указанном на странице олимпиады (<http://robbo.ru/olimp/>), создать проект, удовлетворяющий условиям конкурса, и разместить его на сайте (возможно, с помощью учителя, родителя или старшего товарища). После этого члены жюри проверят работы и выявят победителей.

Работать в Scratch можно как онлайн, на сайте [4], так и скачав программу со страницы официального сайта [2].

Ребята могут научиться программировать на Scratch на таком уровне, чтобы создавать конкурсные проекты олимпиады, довольно быстро, буквально за несколько месяцев. Причем можно обучать школьников, предлагая им задания на реализацию проектов, аналогичные заданиям Scratch-олимпиады. Ниже рассмотрены примеры таких заданий. Данные проекты приведены в книге [3], планируемой к выпуску в июне 2018 года.

Мультфильм «Рыцарь и принцесса» (I—II классы).

Представим себе, что принцесса заблудилась в лесу. Она долго пытается найти дорогу в свой замок, но у нее ничего не получается. Темнеет, принцессе становится страшно. Кругом ей мерещатся пугающие привидения... Кто же спасет бедную принцессу? Ну, конечно, храбрый рыцарь!

Реализация проекта.

Текст, который дан ниже, предлагается школьникам в виде распечатки или в электронном виде.

Откройте программу.

Прежде чем начать создавать проект, сохраните программу, выбрав команду *Файл, Сохранить как*.

Спрайты, которые нам понадобятся, находятся в библиотеке спрайтов под именами *Princess* (принцесса), *Ghost2* (привидение), *Khight* (рыцарь) (рис. 1).



Рис. 1

Мурзик  для мультфильма нам не понадобится, поэтому его мы удалим, вызвав правой кнопкой мыши меню и выбрав в нем команду *удалить*. Привидения принцессе будут мерещиться и слева, и справа, поэтому дублируем спрайт *Ghost2* с помощью кнопки *Дублировать*, расположенной в верхней части окна (рис. 2).



Рис. 2

Но как же сделать так, чтобы одно из привидений было повернуто в другую сторону? Для этого нажмите на тот спрайт в области спрайтов, который вы хотите повернуть. В правом верхнем углу окна программы во вкладке *Костюмы* есть специальная кнопка *Отразить слева направо* , нажмите ее. Спрайт повернется. В итоге сцена должна выглядеть так, как показано на рисунке 3 (расположение спрайтов неважно).

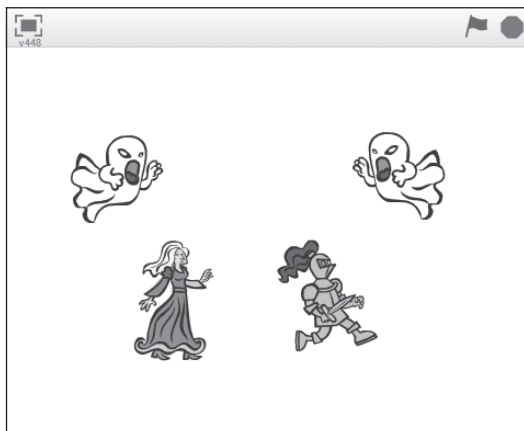


Рис. 3

Сейчас на сцене спрайты очень большие, поэтому с помощью инструментов *Увеличить*  и *Уменьшить*  сделайте спрайты поменьше.

Также нам понадобятся два фона — *castle3* (замок) и *forest* (лес), добавьте их из библиотеки фонов. Белый фон (*фон 1*) нам не нужен, удалим его (с помощью правой кнопки мыши). В итоге должны остаться только два добавленных из библиотеки фона (рис. 4).

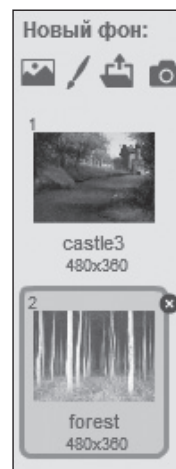


Рис. 4

Начнем писать программу с фона, как самого простого. Нажмите на вкладку *Фоны* под сценой, а затем на вкладку *Скрипты* в верхней части окна (рис. 5).

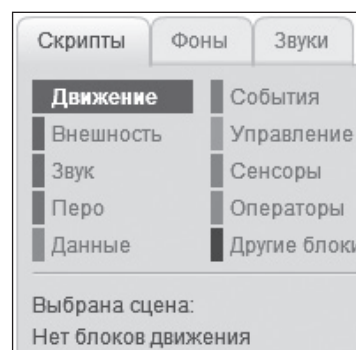


Рис. 5

Справа появится область, где мы можем писать программу.

Представим себе, что сначала принцесса идет по светлому лесу и в нем поют птицы, а потом в лесу темнеет и просыпается сова.

Первый скрипт мы начинаем с команды *когда щелкнут по* (рис. 6).



Рис. 6

Для начала сбросим значение яркости (выберем группу *Внешность*, а затем из выпадающего списка эффект *яркость*) для того, чтобы при каждом следующем просмотре яркость менялась одинаково. Поскольку у нас два фона, то в процессе анимирования фон нужно будет сменить на *forest* (также выберем его из выпадающего списка) (рис. 7).

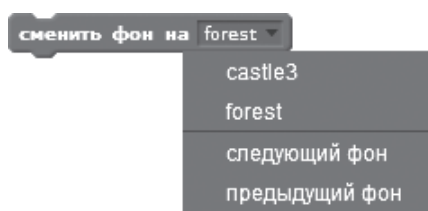


Рис. 7

Поскольку в начале мультфильма должны петь птицы, а потом, когда стемнеет, должна начать ухать сова, на вкладке *Звуки* в верхней части окна программы сразу добавим нужные звуки из библиотеки: *bird* (пение птицы), *rattle* (грохот), *whoop* (уханье совы) (рис. 8).



Рис. 8

Если в группе *Звуки* выбрать команду *играть звук bird* (звук выбираем из выпадающего списка), то можно услышать, что этот звук очень короткий, а нам хотелось бы, чтобы он повторялся трижды. Как это сделать? Очень просто! В группе *Управление* есть команда *повторить ... раз*. Перетащим ее в программу. Внутрь поместим команду *играть звук bird*, после — *ждать 0.5 секунд*.

Далее вставим команду *ждать 1 секунд*.

После этого нужны команды *играть звук rattle*, *ждать 5 секунд* и *изменить яркость эффект на -20*. Последняя команда нужна для создания темноты, так как в лесу потемнело. Должно получиться так, как показано на рисунке 9. Скрипт для фона готов!

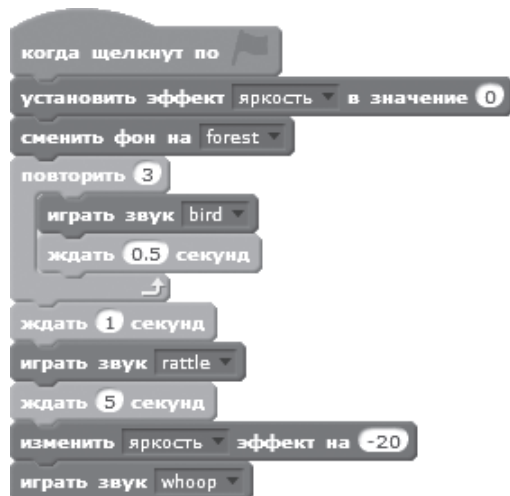


Рис. 9

В начале мультфильма рыцарь и привидения нам не нужны. Для каждого спрайта создадим скрипты, позволяющие спрятать их (рис. 10).

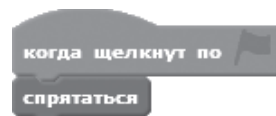


Рис. 10

Теперь после нажатия на зеленый флажок на сцене остается только принцесса. Для спрайта *Princess* создайте подпрограмму так, как показано на рисунке 11.

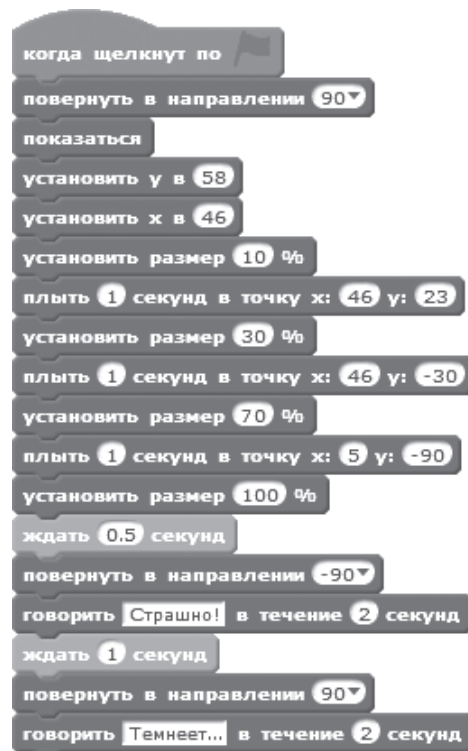


Рис. 11

Кратко прокомментируем команды этого скрипта. Напомним, что *x* и *y* — это координаты: координата *x* отвечает за положение по горизонтали, координата *y* — за положение по вертикали. Сначала спрайт *Princess* установим в положение с координатами (46, 58). Изначальный размер установим равным 10 % от исходного. После этого начнем постепенно увеличивать размер спрайта для создания эффекта приближения. При этом установим движения таким образом, чтобы принцесса шла по дороге.

Внимание! Проследите, чтобы принцесса не перевернулась вниз головой. Для этого в поле спрайтов надо нажать на ее спрайт, далее на букву *i* и в строке *стиль вращения* выбрать $\leftrightarrow$ (это меню настройки *info*). То есть должно получиться так, как показано на рисунке 12.

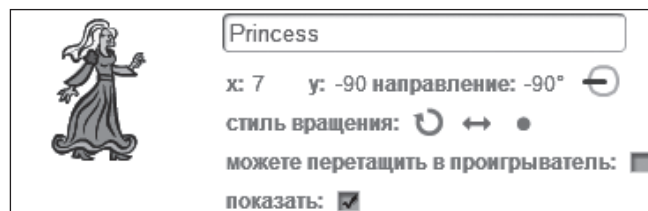


Рис. 12

Теперь принцесса идет по лесу, произносит слова. Стемнело, и принцессе начинают мерещиться привидения.

А как же привидения должны понять, когда им нужно вылетать? Для этого в группе *События* есть две команды: *когда я получу сообщение* (рис. 13) и *передать сообщение* (рис. 14).



Рис. 13



Рис. 14

В конец нашей подпрограммы вставим команду *передать первый призрак* — в выпадающем списке выберите *новое сообщение* (рис. 15) и там напишите *Имя сообщения: первый призрак* (рис. 16).

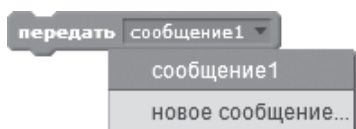


Рис. 15

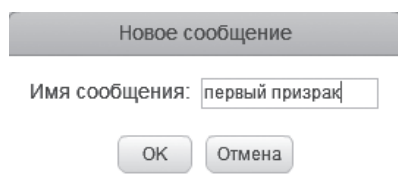


Рис. 16

Кстати! На самом деле сообщение может быть любым, но лучше делать их понятными для облегчения чтения кода программы.

Спрайт *Ghost2* должен получить сообщение *первый призрак*, появиться и начать летать слева от принцессы, после чего спрятаться, передав спрайту *Ghost3* сообщение *второй призрак*. Таким образом, мы получим скрипт, представленный на рисунке 17.

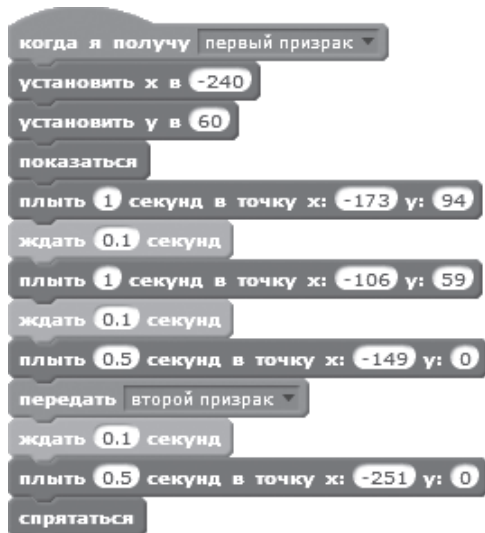


Рис. 17

Спрайт *Ghost3* должен его получить, показаться и также начать летать (рис. 18).

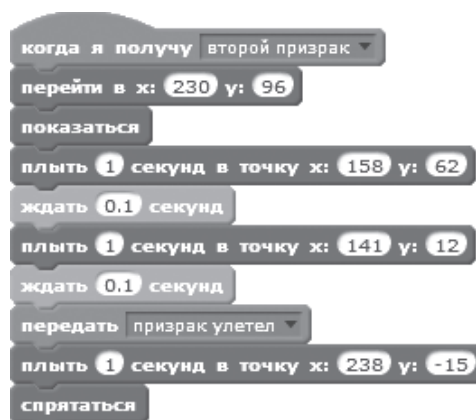


Рис. 18

Принцесса должна оборачиваться в конце их полета и говорить: «Кто здесь?» и «Показалось...»

В итоге основной скрипт для принцессы нужно дописать так, как показано на рисунке 19.

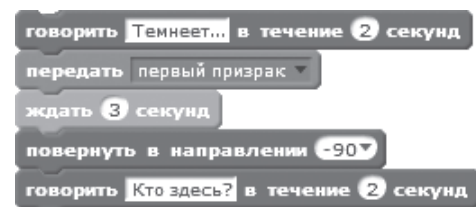


Рис. 19

Также для спрайта *Princess* нужно дописать скрипт, представленный на рисунке 20.

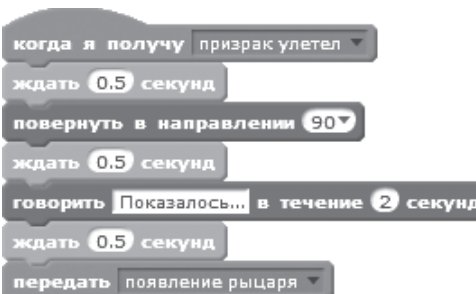


Рис. 20

После выполнения нажмите на зеленый флажок и проверьте правильность выполнения программы. Вы можете придумать свои траектории движения привидений и действия принцессы.

А нам осталось только дождаться спасения принцессы благородным рыцарем! Он должен появиться, когда улетят привидения, и со словами: «Принцесса! Я спасу тебя!» вывести принцессу к замку.

Попробуйте подпрограмму для рыцаря написать самостоятельно, потом сравните с нашей (рис. 21). Не забудьте, что принц при поворотах не должен оказаться вниз головой!

Соответственно, принцесса при получении сообщения *поворот* должна поворачиваться в сторону принца (рис. 22), а при получении сообщения *благодарит рыцаря* говорить ему слова благодарности и уходить с ним (рис. 23).

Осталось только сменить фон на *castle3* и отправить принцессу домой.

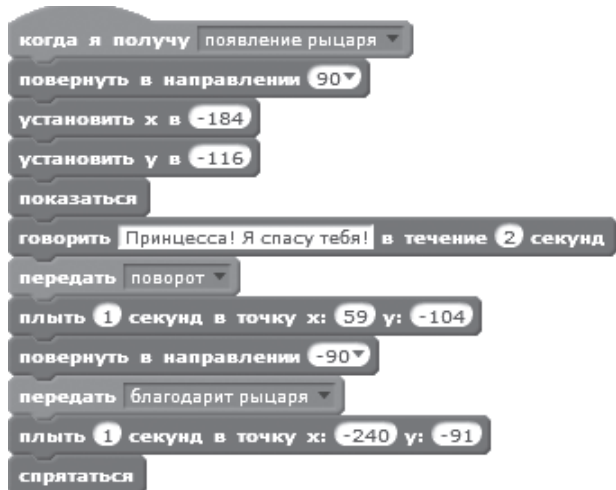


Рис. 21

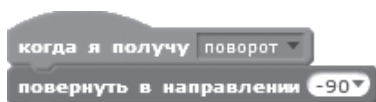


Рис. 22

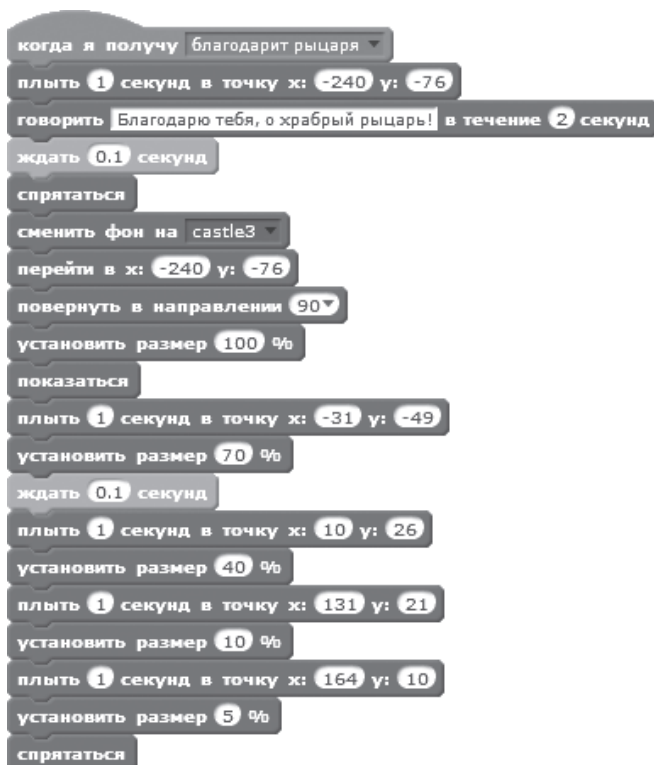


Рис. 23

Наш мультфильм готов!

Игра «Ловим фрукты» (VII—VIII классы).

Приведем одну из игр, созданием которой можно заинтересовать учащихся. В этой игре с волшебного дерева будут падать разные фрукты, а мы с помощью шляпы будем их ловить!

Реализация проекта.

Для игры нам потребуются три фона.

Прежде всего, это само дерево — фон *tree* (дерево) (рис. 24).



Рис. 24

Второй фон (*party*) нам потребуется в случае выигрыша (рис. 25). Он есть в библиотеке, нужно только сделать надпись «You win!» («Ты выиграл!»).



Рис. 25

И третий фон, который нам потребуется, — это фон проигрыша. С помощью кнопки *Нарисовать новый фон* создайте его самостоятельно (*blue sky3*) и сделайте на нем надпись «Game over» («Игра окончена») (рис. 26).

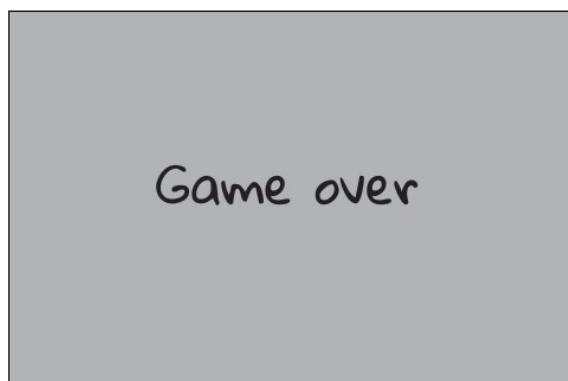


Рис. 26

Что нам еще потребуется для игры? Конечно, спрайты. Добавьте такие спрайты: *Hat* (шляпа, в нее мы будем ловить фрукты) и сами фрукты: *Apple* (яблоко), *Bananas* (бананы), *Orange* (апельсин), *Watermelon* (арбуз).

Для спрайта-шляпы напишем скрипт, благодаря которому спрайт в начале игры будет перемещаться в определенную точку. Также напишем скрипты, с помощью которых шляпой будет управлять пользователь (с помощью клавиш-стрелочек) (рис. 27).

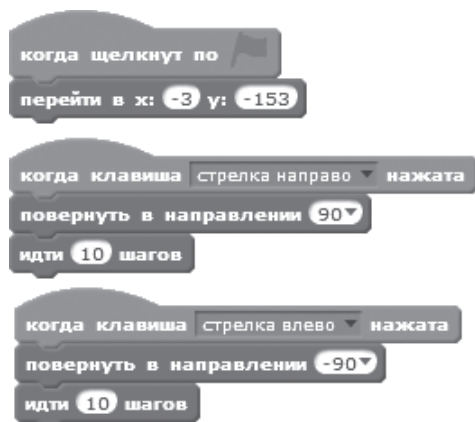


Рис. 27

В начале игры фон должен быть *tree*. Создадим соответствующий скрипт (рис. 28).

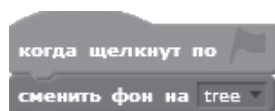


Рис. 28

Для каждого спрайта напишем скрипт, благодаря которому спрайт будет двигаться. Спрайт должен появляться в произвольном месте сверху сцены и двигаться вниз. Создайте приведенный на рисунке 29 скрипт и дублируйте его для всех спрайтов-фруктов. Подробно скрипт мы комментировать не будем, скажем лишь, что необходимо создать переменную *count* и обнулить ее в самом начале скрипта. Если фрукт коснется шляпы, переменную нужно увеличить на единицу, если коснется нижнего края сцены — уменьшить на единицу.

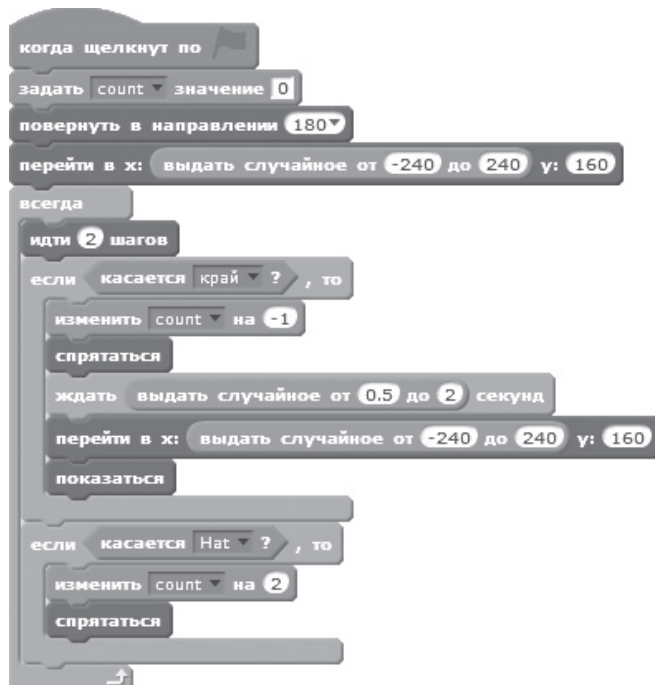


Рис. 29

Отлично! Игра почти готова, осталось совсем немного.

Дополним скрипт, меняющий изначальный фон на *tree*. Если количество очков достигнет 50, должно

появиться сообщение о выигрыше (должен смениться фон). Если же количество очков станет отрицательным (меньше нуля), игрок проиграет, и также должно появиться соответствующее сообщение (опять меняем фон) (рис. 30).

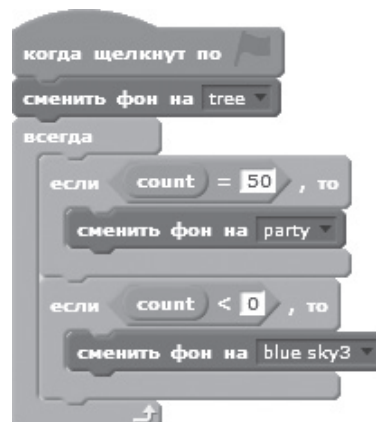


Рис. 30

Создайте нужный скрипт и проверьте, как работает программа. Все должно работать правильно, кроме того, что при смене фона не исчезают спрайты и сообщение о выигрыше или проигрыше выводится, а спрайты-фрукты продолжают двигаться. Исправим это. Для спрайта *Hat* добавим скрипт, с помощью которого при отрицательном значении переменной либо при значении переменной больше 50 спрайты исчезают (рис. 31). Этот скрипт дублируем для абсолютно всех спрайтов.



Рис. 31

Наша игра готова!

Как видно из приведенных выше примеров, подобные проекты сделать не так уж сложно, главное, захотеть, и все получится! Победить в Scratch-олимпиаде может каждый! Можно с уверенностью сказать, что количество подобных соревнований будет только расти. Конечно же, в них стоит участвовать учащимся, а учителям необходимо отслеживать сроки и требования соревнований. Ну и, безусловно, стоит периодически программировать и совершенствовать свои умения.

Список использованных источников

1. Международная Scratch-олимпиада по креативному программированию 2018 // РОББО — образовательная роботехника. <http://robbo.ru/olimp/>
2. Офлайн-редактор Scratch 2.0 // Scratch — Imagine, Program, Share. <https://scratch.mit.edu/scratch2download/>
3. Путина А. С. Scratch 2.0: от новичка к продвинутому пользователю. Пособие по подготовке к Scratch-олимпиаде. М.: Лаборатория знаний, 2018. (В печати.)
4. Scratch — Imagine, Program, Share. <https://scratch.mit.edu>

Уважаемые авторы и читатели!

Издательство «Образование и Информатика» объявляет о проведении конкурса цифровых изображений и фотографий, которые будут размещены на обложке журнала «Информатика в школе» во втором полугодии 2018 года.

Подробности — на сайте издательства «Образование и Информатика»: <http://www.infojournal.ru/>

В оформлении обложки данного номера журнала «Информатика в школе» использованы работы следующих авторов — победителей конкурса цифровых изображений и фотографий:



Ангелина Пыжук,
ученица средней общеобразовательной школы № 8,
г. Ноябрьск, Ямало-Ненецкий автономный округ



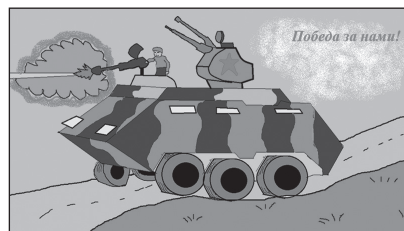
Ксения Юферева,
ученица лицея № 6,
г. Качканар,
Свердловская область



О. Р. Пименова,
учитель информатики
Одинцовской гимназии № 13,
Московская область



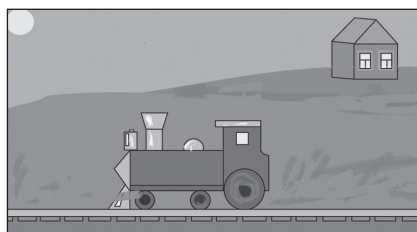
Анастасия Гульченко,
ученица средней общеобразовательной школы № 8,
г. Ноябрьск, Ямало-Ненецкий автономный округ



Матвей Чернятьев,
ученик средней общеобразовательной школы № 8,
г. Ноябрьск, Ямало-Ненецкий автономный округ



Н. В. Лукьянова,
преподаватель Барнаульского
государственного педагогического
колледжа, Алтайский край



Алмаз Гильманов,
студент Мамадышского политехнического
колледжа, Республика Татарстан



Яна Левченко,
ученица средней общеобразовательной
школы № 17, г. Белая Калитва,
Ростовская область



Альбина Каринг,
студентка Куйбышевского
педагогического колледжа, г. Куйбышев,
Новосибирская область



Оксана Разина,
ученица гимназии № 133
имени Героя Социалистического Труда
М. Б. Оводенко, г. о. Самара

ПРАВИЛА ДЛЯ АВТОРОВ

Общие положения

Все присланные статьи рецензируются. Публикация статей возможна только при наличии положительного отзыва рецензентов.

Поскольку рецензирование и предпечатная подготовка материалов занимают не менее трех месяцев, статьи следует присылать в редакцию заблаговременно.

Требования к файлам рукописи

1. Текст статьи должен быть представлен в формате текстового редактора Microsoft Word (*.doc, *.rtf):

- формат листа — А4;
- все поля по 2 см;
- шрифт — Times New Roman, кегль — 12 пт, расстояние между строками — 1,5 (полтора) интервала;
- графические материалы вставлены в текст.

2. Файл со статьей должен содержать следующие данные для публикации, **необходимо строго придерживаться указанной ниже последовательности** (пожалуйста, проверяйте оформление по образцу статьи, представленному на сайте ИНФО):

- **И. О. Фамилия** автора(ов) на русском языке.
- **Место работы** автора(ов) на русском языке. Необходимо указать место работы **каждого** автора. Если из названия организации не следует принадлежность к населенному пункту, через запятую надо указать название населенного пункта.
- **Название статьи** на русском языке.
- **Аннотация** на русском языке (3–5 строк в указанном выше формате).
- **Ключевые слова** на русском языке (не более 10, через запятую).
- **Подробная информация об авторах** — для каждого из авторов:
 - фамилия, имя, отчество (полностью);
 - ученая степень;
 - ученое звание;
 - должность;
 - место работы;
 - адрес места работы (обязательно с индексом);
 - рабочий телефон (обязательно с кодом города);
 - адрес электронной почты (e-mail).
- **И. О. Фамилия** автора(ов) на английском языке.
- **Место работы** автора(ов) на английском языке.
- **Название статьи** на английском языке.
- **Аннотация** на английском языке.
- **Ключевые слова** на английском языке (через запятую).
- **Текст статьи** в указанном выше формате.
- **Список литературных и интернет-источников**, упорядоченный в алфавитном порядке.

3. При отправке статьи в редакцию в полях электронной формы необходимо указать подробные сведения об авторе:

- фамилия, имя, отчество (полностью);
- домашний почтовый адрес (с индексом);
- домашний телефон (обязательно с кодом города);
- мобильный телефон;
- адрес электронной почты (e-mail).

Данные сведения необходимы для оперативной связи с автором статьи и пересылки авторского экземпляра журнала и **не подлежат публикации**.

Если авторов несколько, необходимо представить указанные сведения **обо всех авторах**.

4. При необходимости статья может сопровождаться дополнительным материалом в электронном виде (презентации, листинги программ, книги Excel, примеры выполнения работ и др.), который будет размещен на сайте ИНФО.

5. Иллюстрации следует представлять в виде отдельных графических файлов (даже при их наличии в документе Word) в формате TIFF или JPG, разрешение — не менее 300 пикселей на дюйм.

Уважаемые коллеги!

Статьи для публикации в журналах «Информатика и образование» и «Информатика в школе» должны отправляться в редакцию **только через электронную форму на сайте ИНФО (раздел «Авторам → Отправка статьи»):**

<http://infojournal.ru/authors/send-article/>

Обращаем ваше внимание, что для отправки статьи необходимо предварительно зарегистрироваться на сайте ИНФО (или авторизоваться — для зарегистрированных пользователей).

С требованиями к оформлению представляемых для публикации материалов можно ознакомиться на сайте ИНФО в разделе **«Авторам»:**

<http://infojournal.ru/authors/>

Дополнительную информацию можно получить в разделе **«Авторам → Часто задаваемые вопросы»:**

<http://infojournal.ru/authors/faq/>

а также в редакции ИНФО:

e-mail: readinfo@infojournal.ru

телефон: (495) 140-19-86

Журнал «Информатика в школе»

Индексы подписки (агентство «Роспечать»)
на 2-е полугодие 2018 года

- 81407 — для индивидуальных подписчиков
- 81408 — для организаций

Периодичность выхода: 5 номеров в полугодие (в июле не выходит)

Редакционная стоимость:
индивидуальная подписка — 150 руб.
подписка для организаций — 300 руб.



Федеральное государственное унитарное предприятие "Почта России" Ф СП - 1															
Бланк заказа периодических изданий															
АБОНЕМЕНТ						На газету журнал									
Информатика в школе								(индекс издания)							
(наименование издания)						Количество комплектов									
На 20 18 год по месяцам															
1	2	3	4	5	6	7	8	9	10	11	12				
Куда															
				(почтовый индекс)				(адрес)							
Кому _____															
----- Линия отреза -----															
									ДОСТАВОЧНАЯ						
ПВ			место			литер			КАРТОЧКА			(индекс издания)			
На газету журнал						Информатика в школе									
						(наименование издания)									
Стои- мость		подписки		руб.		Количество комплектов									
		каталож- ная		руб.											
		переадре- совки		руб.											
На 20 18 год по месяцам															
1	2	3	4	5	6	7	8	9	10	11	12				
почтовый индекс				Город				село				область			
код улицы				Район				улица				дом			
корпус				квартира				Фамилия И.О.							