

# О НЕКОТОРЫХ ВОЗМОЖНОСТЯХ ИСПОЛЬЗОВАНИЯ СРЕДЫ SCRATCH 2.0 В ОБРАЗОВАТЕЛЬНОМ ПРОЦЕССЕ

**Косярский Александр Алексеевич,**

*педагог дополнительного образования МАОУ ДО ЦДТ «Прикубанский»*

**Мороз Ольга Викторовна,**

*кандидат педагогических наук, доцент кафедры информационных и образовательных технологий факультета математики и компьютерных наук ФГБОУ ВО «Кубанский государственный университет»*

В СТАТЬЕ РАССМАТРИВАЕТСЯ ВОЗМОЖНОСТЬ ИСПОЛЬЗОВАНИЯ СРЕДЫ SCRATCH 2.0 В ОБРАЗОВАТЕЛЬНОМ ПРОЦЕССЕ НА ПРИМЕРЕ РАЗРАБОТАННОЙ СОВМЕСТНО С УЧАЩИМИСЯ ПРОГРАММНОЙ ИМИТАЦИИ АЛГОРИТМИЧЕСКОГО ИСПОЛНИТЕЛЯ «РОБОТ», ПРИМЕНЯЕМОГО В ШКОЛЬНОМ КУРСЕ ИНФОРМАТИКИ В КАЧЕСТВЕ СРЕДСТВА ДЛЯ ИЗУЧЕНИЯ АЛГОРИТМОВ. АВТОРОМ СТАТЬИ ПРЕДСТАВЛЕН ПРИМЕР ПРОГРАММНОЙ РЕАЛИЗАЦИИ ПОДОБНОГО ИСПОЛНИТЕЛЯ С РАЗЛИЧНЫМИ РЕЖИМАМИ РАБОТЫ. В СТАТЬЕ РАССМАТРИВАЮТСЯ НАИБОЛЕЕ ЧАСТЫЕ ПРОБЛЕМЫ, С КОТОРЫМИ СТАЛКИВАЮТСЯ УЧАЩИЕСЯ В ПРОЦЕССЕ ИЗУЧЕНИЯ ПРОГРАММИРОВАНИЯ, А ТАКЖЕ ВОЗМОЖНОСТЬ ЧАСТИЧНОГО РЕШЕНИЯ ПОСТАВЛЕННЫХ ПРОБЛЕМ С ИСПОЛЬЗОВАНИЕМ РАЗРАБОТАННОГО АЛГОРИТМИЧЕСКОГО ИСПОЛНИТЕЛЯ. АВТОРЫ СТАТЬИ ДЕМОНИСТРИРУЮТ ВОЗМОЖНОСТЬ ИСПОЛЬЗОВАНИЯ ПРОГРАММНЫХ РЕЖИМОВ СОЗДАННОЙ СРЕДЫ ДЛЯ РЕШЕНИЯ ПОСТАВЛЕННЫХ МЕТОДИЧЕСКИХ И ОБРАЗОВАТЕЛЬНЫХ ЗАДАЧ. В ПУБЛИКАЦИИ ПРИВОДЯТСЯ НЕКОТОРЫЕ УЧАСТКИ ПРОГРАММНЫХ КОДОВ ОСНОВНЫХ РЕЖИМОВ РАБОТЫ ИСПОЛНИТЕЛЯ, А ТАКЖЕ РАСКРЫВАЮТСЯ ПРИНЦИПЫ РАБОТЫ ОТДЕЛЬНЫХ ПРОЦЕДУР И ФУНКЦИЙ.

- Scratch 2.0 • школьный курс информатики • исполнитель «Робот» • алгоритмизация
- программирование

Стремительное развитие технологий, произошедшее за последние 20 лет, привело к масштабной информатизации всех сфер жизни человека и общества в целом, нося при этом в некоторых случаях неуправляемый, лавинообразный характер.

Сегодня во многих крупных научно-исследовательских институтах используются сверхмощные компьютеры, а исследования направлены на изучение путей создания искусственного интеллекта. Однако такое бурное внедрение и развитие цифровых технологий, безусловно, несёт ряд реальных рисков и угроз.

проблемы и её актуальности, открытое письмо человечеству физика-теоретика С. Хокинга, инициированное им в 2015 году. Данное письмо было подписано рядом известных личностей, среди которых Илон Маск, Люк Мюльхаузер (директор Института исследований машинного обучения) и многие другие. В письме Хокинг призывает тщательно координировать разработки в области ИИ и сообщать о своих достижениях, с целью недопущения выхода последнего из-под контроля. В условиях полномасштабного развития IT-индустрии и вступления Российской Федерации в эпоху информационного общества приоритетной задачей органов государственной власти становится обеспечение качественного технического образования. Так, начиная с 2019 года в Российской Федерации был принят национальный проект «Цифровая экономика», одним из приоритетных направлений которого стало

Так, в статье<sup>1</sup> автор приводит в качестве аргумента, указывающего на существование данной про-

<sup>1</sup> Миронов, В.В. Всеобщая информатизация образования: внутри и вне процесса / В.В. Миронов, С.Н. Латынов // Cloud of Science. — 2017. Т. 4. — № 2. — С. 282–302.



Рис. 1. Частые проблемы при изучении раздела «Алгоритмы и элементы программирования»

развитие образования, реализуемого в рамках федерального проекта «Кадры для цифровой экономики». В рамках данного проекта подразумевается создание до 1 января 2022 года системы выявления, поддержки и развития талантов в областях математики, информатики и цифровых технологий<sup>2</sup>. Однако необходимо понимать, что формирование подобной системы невозможно без повышения качества не только основного общего образования по данным предметным областям, но и дополнительного образования, реализуемого при помощи специализированных кружков и факультативов.

Ключевым документом, указывающим на путь развития образования в области цифровых технологий, его главные цели и задачи, выступает федеральный государственный образовательный стандарт основного общего образования (далее ФГОС ООО) по образовательной программе «Информатика»<sup>3</sup>. Так, одной из важнейших задач курса «Информатика» является формирование у учащихся устойчивой системы знаний, умений, навыков, а также компетенций для работы с различными видами информации. Не менее важной, особенно в контексте указов Президента РФ и концепций в рамках названных выше нацпроектов<sup>4</sup>, выступает развитие алгоритмического и логического мышления.

В курсе информатики в школе УМК и рабочими программами предусмотрено изучение такого раздела информатики, как «Алгоритмы и элементы программирования». Но ввиду крайне малого количества часов,

отводимых для изучения информатики, данному разделу уделяется не так много внимания, как это было бы необходимо. Вследствие этого можно выделить 4 ключевые проблемы, с которыми чаще всего сталкиваются учащиеся при изучении этого раздела (рис. 1).

В качестве одного из путей решения поставленных проблем мы видим увеличение количества часов для изучения данного раздела посредством факультативов и кружков. В работах<sup>5</sup> уже описана структура организации учебного курса по изучению программирования с использованием среды Scratch 2.0, созданного с целью систематизации и обобщения знаний по программированию и теории алгоритмов. Поэтому

<sup>2</sup> Кадры для цифровой экономики: официальный сайт. — URL: <https://data-economy.ru/organization> (дата обращения: 26.11.2020).

<sup>3</sup> Федеральный государственный образовательный стандарт основного общего образования (утв. приказом Министерства образования и науки РФ от 17 декабря 2010 г. № 1897) — Доступ из справ.-правовой системы Гарант. — Текст: электронный.

<sup>4</sup> Указ Президента Российской Федерации «О национальных целях и стратегических задачах развития Российской Федерации на период до 2024 года» от 7.05.2018 г. — Доступ из справ.-правовой системы КонсультантПлюс. — Текст: электронный; Кадры для цифровой экономики: официальный сайт. — URL: <https://data-economy.ru/organization> (дата обращения: 26.11.2020); Правительство Российской Федерации: официальный сайт. — Москва. — URL: <http://government.ru> (дата обращения: 26.11.2020). — Текст: электронный.

<sup>5</sup> Косярский, А.А. Методические рекомендации по организации группового взаимодействия в рамках курсов по обучению программированию / А.А. Косярский // Научный аспект. — 2019. — № 2. — С. 162–170; Косярский, А.А. Особенности организации курсов для учащихся средней школы по программированию с использованием среды Scratch 2.0 / А.А. Косярский, О.В. Мороз // Школьные технологии. — 2019. — № 2. — С. 76–85.



Рис. 2. Режимы работы программы

в данной статье мы остановимся на использовании данной среды для частичного решения представленных выше проблем, рассмотрев конкретный пример программы — «Универсальный алгоритмический исполнитель "Робот"».

На рисунке 2 представлено меню выбора режимов разработанной программы.

Каждый из режимов служит для той или иной задачи. Так, режимы «Классический» и «Автоматизированный» могут выступать в качестве демонстрационного пособия для учащихся при самостоятельной реализации своего исполнителя, обладающего статической и динамической системами команд исполнителя. Режим «Автоматизированный +» в первую очередь предназначен для преподавателей, позволяя создавать списки, содержащие последовательности команд для исполнителя, а также проверять команды, написанные учащимися.

Работа учащегося с данной программой может быть разделена на два этапа:

- тестирование готовой программы, анализ скриптов, заполнение карты диагностики;
- создание собственного алгоритмического исполнителя.

Первый этап, в соответствии с ФГОС ООО<sup>6</sup>, ориентирован на достижение следующих результатов.

**Формирование алгоритмической культуры.** Культура написания последовательностей команд для исполнителя формируется в процессе анализа программного кода. Обращается внимание учащегося на обязательное наличие задаваемых команд в системе исполнителя.

**Формирование представления об основных изучаемых понятиях.** Учащимся при помощи списков, содержащих выполняемые команды, показывается связь между алгоритмом для Робота и последовательностью элементарных действий. На примере использования списка «Команды» вводится понятие системы команд исполнителя как совокупности элементарных действий.

В рамках первого этапа учащийся проводит анализ работы созданной программы, посредством её тестирования и наблюдения. Для фиксации замечаний учащемуся предлагается заполнить диагностический лист (рис. 3).

	Классический	Автоматизированный
Какие функции имеются?		
Какие блоки могли быть использованы?		
Степень сложности программы?		
Наличие е автоматизированного управления «Роботом»?		

Рис. 3. Диагностический лист тестирования программы

<sup>6</sup> Федеральный государственный образовательный стандарт основного общего образования (утв. приказом Министерства образования и науки РФ от 17 декабря 2010 г. № 1897) — Доступ из справ.-правовой системы Гарант. — Текст: электронный.

Роль этого этапа крайне велика с точки зрения формирования информационной компетентности и знаний в области программирования. В рамках первой половины данного этапа учащийся должен научиться выявлять сильные и слабые стороны созданного алгоритма, фиксировать, если имеются, ошибки в работе программного кода, предлагать альтернативные пути реализации программы. Также при анализе программы учащийся при поддержке педагога знакомится с такими понятиями теории алгоритмов, как «исполнитель» (в нашем случае Робот), «система команд исполнителя», «программа для исполнителя».

В рамках второй половины первого этапа учащийся изучает программный код для конкретного исполнителя. На этой стадии важно обратить внимание учащегося на необходимость создания специальных списков для хранения элементарных команд исполнителя — система команд, и для хранения последовательности команд для исполнителя — программа. Также важно подвести учащегося к осознанию необходимости использования процедур и функций для выполнения многократных действий, таких как передача команд исполнителю посредством функции **Запись**.

После проведения данного анализа учащемуся ставится задача самостоятельного ис-

полнителя на основании сформулированных замечаний и выявленных «слабостей» и «уязвимостей».

По завершении второго этапа планируется достижение следующих результатов.

**Развитие умений составления и записи алгоритма для конкретного исполнителя.** При самостоятельной разработке своего алгоритмического исполнителя учащийся ищет наиболее удачные способы организации системы команд исполнителя, программ для конкретного исполнителя и т.д. В процессе данного поиска учащийся определяет наиболее удобные формы представления алгоритмов, среди которых использование списков.

**Приобретение навыков и опыта разработки программ в выбранной среде программирования, включая тестирование и отладку.** Задача написания собственного алгоритмического исполнителя имеет очень важное методическое значение для изучения программирования в среде Scratch. Методическая значимость обусловлена в первую очередь тем, что при написании программы используются практически все типы блоков, объединяющиеся в самых различных комбинациях, благодаря чему учащиеся формируют связи между отдельными темами программирования.

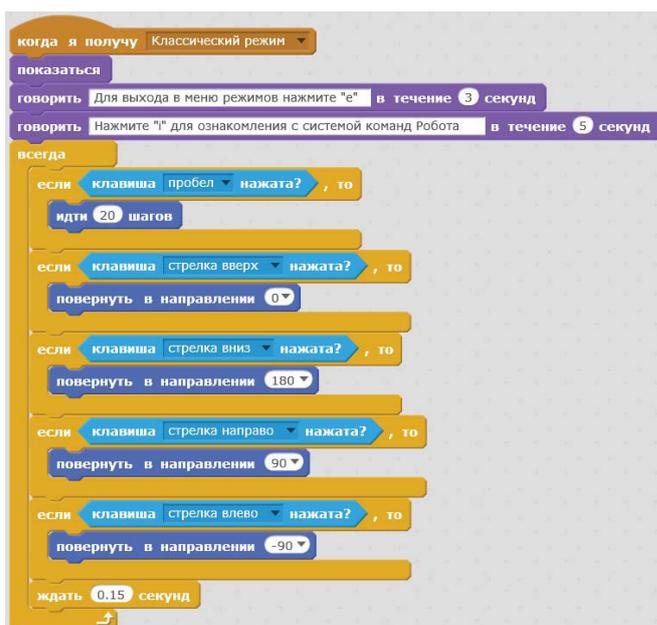


Рис. 4. Некоторые скрипты режима «Классический»

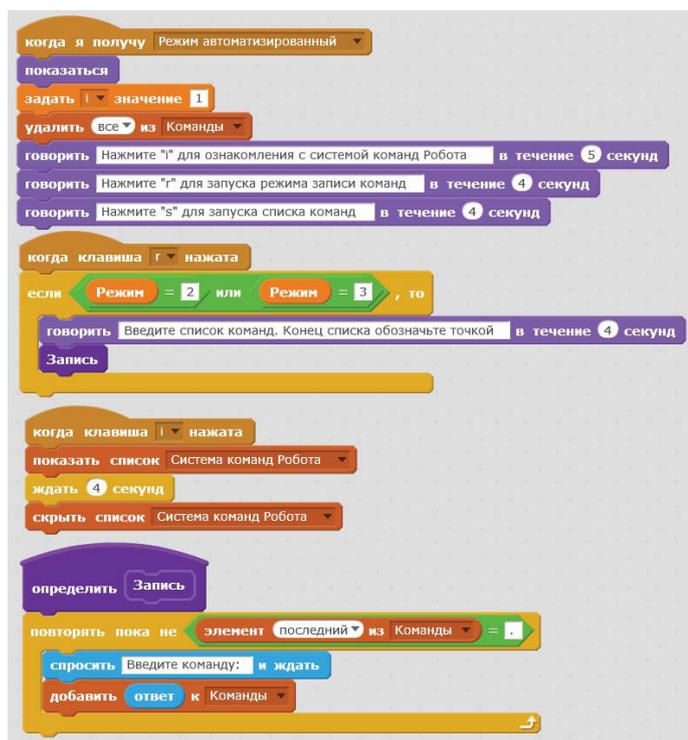


Рис. 5. Некоторые скрипты режима «Автоматизированный»

В первой половине этапа учащемуся предлагается создать универсального исполнителя алгоритмов, обладающего уже выявленными особенностями. В качестве образца — наглядного пособия для учащихся доступны функции и алгоритмы, разработанные для исполнителя **Робот**. На рисунках 4 и 5 представлены некоторые скрипты для разных режимов работы исполнителя.

Принцип работы кода, представленного на рисунке 5, заключается в следующем:

- 1) участнику выводится два сообщения, указывающие на способ перехода в главное меню и режим справки по системе команд;



Рис. 6. QR-код для доступа к файлу программы

- 2) в процессе работы исполнитель постоянно отслеживает нажатие кнопок: пробел, стрелки вверх, вниз, влево, направо и выполняет указанные для нажатия этих кнопок действия.

С полным программным кодом, а также самой программой можно ознакомиться по ссылке: <https://yadi.sk/d/HffPdWDTonLHug> или при помощи QR-кода (рис. 6).

Вторая половина этапа связана с индивидуальной проектной деятельностью, ориентированной на усложнение созданной программы. В таблице 1 представлены возможные пути усложнения проекта, формируемые в процессе их выполнения навыки, а также методическая значимость.

### Выводы

Данная программа может быть рассмотрена как важный инструмент при изучении элементов теории алгоритмов, а также основ программирования в среде Scratch 2.0.

В процессе самостоятельной разработки данной программы учащиеся формируют следующие навыки программирования:

Таблица 1

**Пути совершенствования программы «Универсальный алгоритмический исполнитель»**

Путь совершенствования	Формируемые навыки
Написание функции автоматического составления произвольного списка команд исполнителя указанной длины	<ul style="list-style-type: none"> <li>• Работы с процедурами и функциями;</li> <li>• Работы с циклическими структурами;</li> <li>• Работы с условными операторами в полной и краткой формах</li> </ul>
Написание функции рисования геометрических фигур (произвольных и правильных)	<ul style="list-style-type: none"> <li>• Работы с пером;</li> <li>• Работы с графическими эффектами;</li> <li>• Работы с растровыми и векторными редакторами;</li> <li>• Работы блоками автоматизации действий</li> </ul>
Написание функции редактирования обстановки исполнителя	<ul style="list-style-type: none"> <li>• Создания скриптов автоматической перерисовки обстановки;</li> <li>• Работы с сенсорами для организации взаимодействия объектов;</li> <li>• Работы с координатами объектов;</li> <li>• Работы со списками и переменными</li> </ul>

- работы с циклическими структурами при написании алгоритмов;
- работы с условными операторами в полной и краткой формах;
- работы с сенсорами для организации взаимодействия объектов;
- работы с подпрограммами для оптимизации программного кода;
- работы переменных и списков для обработки данных.

Достигаются следующие результаты в соответствии с ФГОС ООО:

- формирование алгоритмической культуры;
- формирование представления об основных изучаемых понятиях;
- развитие умений составления и записи алгоритма для конкретного исполнителя;
- приобретение навыков и опыта разработки программ в выбранной среде программирования, включая тестирование и отладку.

В заключение необходимо сказать, что использование в образовательном процессе объектно-ориентированных сред программирования, например среды Scratch, позволяет не только повысить качество образовательного процесса, но и увеличить степень вовлечённости в этот процесс всех его участников. Применение мощностей среды программирования Scratch 2.0 в рамках изучения школьного курса информатики и во внеучебной деятельности, например при написании проектных работ или при проведении интерактивных опытов, позволяет успешно реализовывать поставленные

государством задачи в области подготовки высококвалифицированных кадров для цифровой экономики. □

**Литература**

1. Указ Президента Российской Федерации «О национальных целях и стратегических задачах развития Российской Федерации на периода до 2024 года» от 7.05.2018 г. — Доступ из справ.-правовой системы КонсультантПлюс. — Текст: электронный.
2. Федеральный государственный образовательный стандарт основного общего образования (утв. приказом Министерства образования и науки РФ от 17 декабря 2010 г. № 1897) — Доступ из справ.-правовой системы Гарант. — Текст: электронный.
3. Босова, Л.Л. Методика применения интерактивных сред для обучения младших школьников программированию [Текст] / Л.Л. Босова, Т.Е. Сорокина // Информатика и образование. — 2014. — № 7 — С. 61–68.
4. Голиков, Д.В. 40 проектов на Scratch для юных программистов / Д.В. Голиков — СПб.: БХВ-Петербург, 2019. — 192 с.
5. Голиков, Д.В. Scratch для юных программистов / Д.В. Голиков — СПб.: БХВ-Петербург, 2019. — 192 с.
6. Денисова, Л.В. Вокруг треугольника Серпинского. Игра в хаос и программирование в среде Scratch / Л.В. Денисова, В.О. Дженжер // Информатика в школе. — 2020. — № 4. — С. 20–27.

7. Еремин, Е.А. Среда Scratch — первое знакомство / Е.А. Еремин // Газета «Информатика» — М.: Первое сентября, 2008 — № 20 (573) — С. 17–24.
8. Кадры для цифровой экономики: официальный сайт. — URL: <https://data-economy.ru/organization> (дата обращения: 26.11.2020).
9. Косярский, А.А. Методические рекомендации по организации группового взаимодействия в рамках курсов по обучению программированию / А.А. Косярский // Научный аспект. — 2019. — № 2. — С. 162–170.
10. Косярский, А.А. Особенности организации курсов для учащихся средней школы по программированию с использованием среды Scratch 2.0/ А.А. Косярский, О.В. Мороз // Школьные технологии. — 2019. — № 2. — С. 76–85.
11. Миронов, В.В. Всеобщая информатизация образования: внутри и вне процесса / В.В. Миронов, С.Н. Латынов // Cloud of Science. — 2017. Т. 4. — № 2. — С. 282–302.
12. Патаракин, Е.Д. Школа SCRATCH / Е.Д. Патаракин // Школьные технологии. — 2010. — № 4. — С. 132–135.
13. Правительство Российской Федерации: официальный сайт. — Москва. — URL: <http://government.ru> (дата обращения: 26.11.2020). — Текст: электронный.
14. Путина, А.С. SCRATCH-олимпиада по креативному программированию / А.С. Путина // Информатика в школе. — 2018. — №4 (137). — С. 56–61.
15. Савченкова, М.В. Scratch-Хакатон «Программируем в среде Scratch» / М.В. Савченкова // Информатика в школе. — 2018. — №4(137). — С. 24–27.
3. Bosova, L.L. Metodika primeneniya interaktivnykh sred dlya obucheniya mladshikh shkol'nikov programmirovaniyu [Tekst] / L.L. Bosova, T.Ye. Sorokina// Informatika i obrazovaniye. — 2014. — № 7 — S. 61–68.
4. Golikov, D.V. 40 proyektov na Scratch dlya yunykh programmistov / D.V. Golikov — SPb.: BKHV-Peterburg, 2019. — 192 s.
5. Golikov, D.V. Scrtach dlya yunykh programmistov /D.V. Golikov — SPb.: BKHV-Peterburg, 2019. — 192 s.
6. Denisova, L.V. Vokrug treugol'nika Serpinskogo. Igra v khaos i programmirovaniye v srede Scratch /L.V. Denisova, V.O. Dzhenzher // Informatika v shkole. — 2020. — № 4. — S. 20–27.
7. Yeremin, Ye.A. Sreda Scratch — pervoye znakovstvo / Ye.A. Yeremin // Gazeta «Informatika» — M.: Pervoye sentyabrya, 2008 — № 20 (573) — S. 17–24.
8. Kadry dlya tsifrovoy ekonomiki: ofitsial'nyy sayt. — URL: <https://data-economy.ru/organization> (data obrashcheniya: 26.11.2020).
9. Kosyarskiy, A.A. Metodicheskiye rekomendatsii po organizatsii gruppovogo vzaimodeystviya v ramkakh kursov po obucheniyu programmirovaniyu / A.A. Kosyarskiy // Nauchnyy aspekt. — 2019. — № 2. — S. 162–170.
10. Kosyarskiy, A.A. Osobennosti organizatsii kursov dlya uchashchikhsya sredney shkoly po programmirovaniyu s ispol'zovaniyem sredy Scratch 2.0 / A.A. Kosyarskiy, O.V. Moroz // Shkol'nyye tekhnologii. — 2019. — № 2. — S. 76–85.
11. Mironov, V.V. Vseobshchaya informatizatsiya obrazovaniya: vnutyri i vne protsesssa / V.V. Mironov, S.N. Latynov // Cloud of Science. — 2017. T. 4. — № 2. — S. 282–302.
12. Patarakin, Ye.D. Shkola SCRATCH / Ye.D. Patarakin // Shkol'nyye tekhnologii. — 2010. — № 4. — S. 132–135.
13. Pravitel'stvo Rossiyskoy Federatsii: ofitsial'nyy sayt. — Moskva. — URL: <http://government.ru> (data obrashcheniya: 26.11.2020). — Tekst: elektronnyy.
14. Putina, A.S. SCRATCH-olimpiada po kreativnomu programmirovaniyu / A.S. Putina // Informatika v shkole. — 2018. — № 4 (137). — S. 56–61.
15. Savchenkova, M.V. Scratch-Khakaton «Programmiruyem v srede Scratch» / M.V. Savchenkova // Informatika v shkole. — 2018. — №4(137). — S. 24–27.

## Literatura

1. Ukaz Prezidenta Rossiyskoy Federatsii «O natsional'nykh tselyakh i strategicheskikh zadachakh razvitiya Rossiyskoy Federatsii na perioda do 2024 goda» ot 7.05.2018 g. — Dostup iz sprav.-pravovoy sistemy KonsultantPlyus. — Tekst: elektronnyy.
2. Federal'nyy gosudarstvennyy obrazovatel'nyy standart osnovnogo obshchego obrazovaniya (utv. prikazom Ministerstva obrazovaniya i nauki RF ot 17 dekabrya 2010 g. № 1897) — Dostup iz sprav.-pravovoy sistemy Garant. — Tekst: elektronnyy.
13. Pravitel'stvo Rossiyskoy Federatsii: ofitsial'nyy sayt. — Moskva. — URL: <http://government.ru> (data obrashcheniya: 26.11.2020). — Tekst: elektronnyy.
14. Putina, A.S. SCRATCH-olimpiada po kreativnomu programmirovaniyu / A.S. Putina // Informatika v shkole. — 2018. — № 4 (137). — S. 56–61.
15. Savchenkova, M.V. Scratch-Khakaton «Programmiruyem v srede Scratch» / M.V. Savchenkova // Informatika v shkole. — 2018. — №4(137). — S. 24–27.