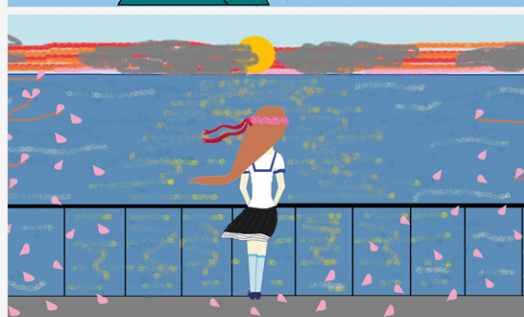
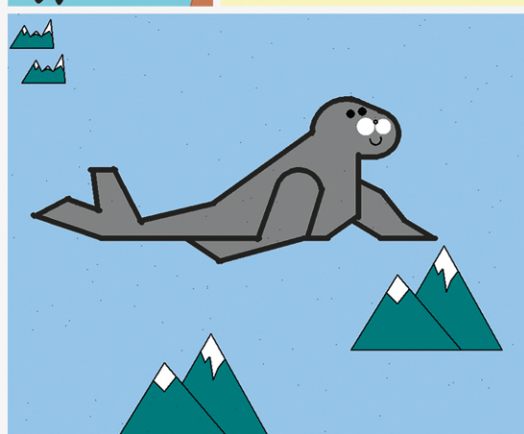
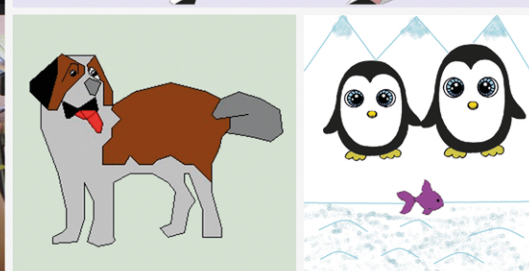
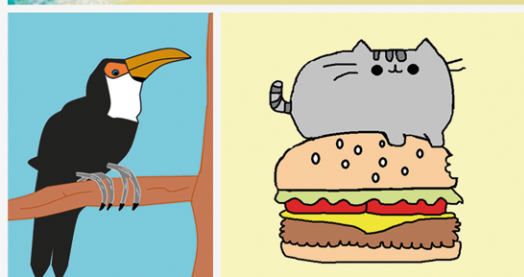
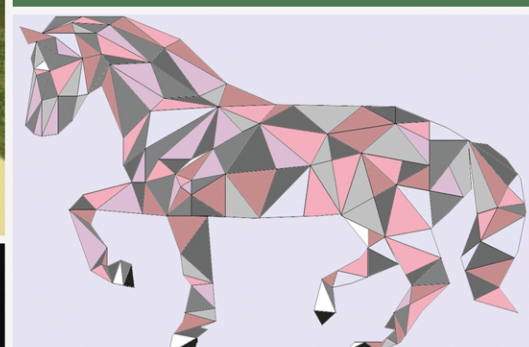
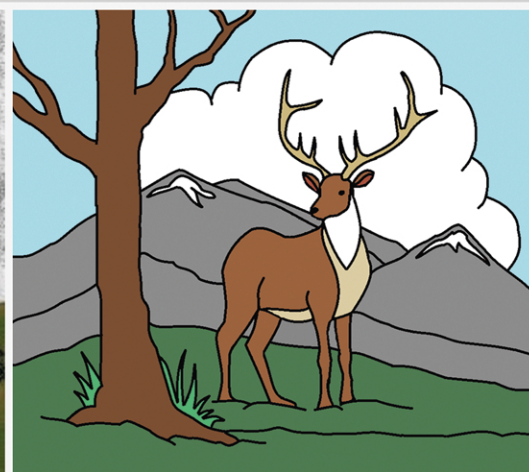


ИНФОРМАТИКА В ШКОЛЕ

ISSN 2221-1993

№ 9'2018

www.infojournal.ru





ХІХ МЕЖДУНАРОДНАЯ НАУЧНО-ПРАКТИЧЕСКАЯ КОНФЕРЕНЦИЯ

НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

Основные направления работы:

- Перспективы развития технологий 1С для создания инфраструктуры цифровой экономики и обновления системы образования.
- Технологическое и методическое обеспечение подготовки граждан к условиям цифровой экономики на основе платформы «1С:Предприятие» и ее прикладных решений.
- Методические, организационные и технологические средства поддержки педагогической деятельности, разработанные на основе решений «1С».
- Создание условий для расширения участия индустрии 1С в системе профессионального образования. Развитие форм сотрудничества образовательных организаций и работодателей.

Мероприятия в рамках конференции:

- Пленарные и секционные заседания
- Мастер-классы по использованию программных продуктов фирмы «1С»
- Вернисаж программных и методических разработок
- Тестирование «1С:Профессионал» по программным продуктам «1С:Предприятие 8»

В 2018 году в конференции приняли участие более 2 300 человек.

Подробнее о тематиках конференции и условиях участия см. сайт 1c.ru/educonf

Участие бесплатное для всех сотрудников образовательных организаций и органов управления образованием (проживание оплачивается отдельно).

Обязательная предварительная регистрация открыта до 28 января 2019 года на сайте 1c.ru/educonf



ФИРМА «1С»
Оргкомитет конференции:
Тел./факс: +7 (495) 688-90-02
Email: npk@1c.ru
www.1c.ru/educonf

29-30 января – 2019 г.
Гостиница «Космос»,
Москва, проспект Мира, д. 150

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.
Шарапова 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 29.11.2018.
Формат 60×90¹/₈. Усл. печ. л. 8,0.
Тираж 2000 экз. Заказ № 705.

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

№ 9 (142) ноябрь 2018

Содержание

ОБЩИЕ ВОПРОСЫ

Босова Л. Л., Каплан А. В. Международная конференция по школьной информатике ISSEP 2018 2

Богданова Д. А. Форум Евросоюза по детской интернет-безопасности SIF 2018 7

МЕТОДИЧЕСКАЯ КОПИЛКА

Лисин Д. В. Философско-мировоззренческие и метапредметные аспекты школьного курса информатики в условиях реализации ФГОС..... 10

Маркушевич М. В. Зависимость между методической системой обучения информатике в школе и типом используемого в ней программного обеспечения..... 14

Белоцерковская И. Е., Втюрин М. Ю. Изучение основ алгоритмизации на примере машины Тьюринга с применением игровой технологии 25

Абухба А. М. Некоторые особенности программирования в Mathcad Prime..... 28

ЗАДАЧИ

Очков В. Ф., Иванова А. О., Алексеев М. Д. Три «жадных» алгоритма 34

Воробьева И. Ю., Дмитриева О. А. Решение загадки Эйнштейна на уроках информатики 43

ОЛИМПИАДЫ ПО ИНФОРМАТИКЕ

Долинский М. С. Введение в решение задач на тему «Исследование» 47

СРЕДСТВА ИКТ В УЧЕБНОМ ПРОЦЕССЕ

Мичкасская Е. И., Вознесенская Н. В. Основы разработки эффективных онлайн-курсов 52

Чечко Д. А., Радионова А. В. Видеокурс «Лицевая анимация в Blender» ... 57



Л. Л. Босова, А. В. Каплан,
Московский педагогический государственный университет

МЕЖДУНАРОДНАЯ КОНФЕРЕНЦИЯ ПО ШКОЛЬНОЙ ИНФОРМАТИКЕ ISSEP 2018

Аннотация

В статье рассмотрены основные направления работы ISSEP — международной конференции по школьной информатике, которая проводится с 2005 года; в 2018 году конференция впервые прошла в России. Кратко изложено содержание основных докладов по секциям: «Фундаментальное обучение информатике учащихся всех возрастов. Роль программирования и алгоритмизации», «Национальные концепции преподавания информатики. Исследования по информатике в школе», «Педагогическое образование в области информатики», «Конкурсы и соревнования по информатике», «Социально-психологические аспекты преподавания информатики», «Компьютерные инструменты в преподавании и изучении информатики».

Ключевые слова: школьная информатика, обучение школьников программированию, вычислительное мышление, конкурс Bebras, гендерные вопросы.

DOI: 10.32517/2221-1993-2018-17-9-2-6

Контактная информация

Босова Людмила Леонидовна, доктор пед. наук, доцент, зав. кафедрой теории и методики обучения математике и информатике Института математики и информатики Московского педагогического государственного университета; *адрес:* 119991, г. Москва, ул. Малая Пироговская, д. 1, стр. 1; *телефон:* (495) 438-79-77; *e-mail:* akull@mail.ru

Каплан Адель Викторовна, магистрант кафедры теории и методики обучения математике и информатике Института математики и информатики Московского педагогического государственного университета; *адрес:* 119991, г. Москва, ул. Малая Пироговская, д. 1, стр. 1; *телефон:* (495) 438-79-77; *e-mail:* adel.caplan@yandex.ru

L. L. Bosova, A. V. Kaplan,
Moscow Pedagogical State University

THE INTERNATIONAL CONFERENCE ON INFORMATICS IN SCHOOLS ISSEP 2018

Abstract

The article describes the main areas of work of ISSEP — the international conference on school informatics, which has been held since 2005; in 2018 the conference was first held in Russia. There is the summary of the main reports by sections: "Sustainable Education in Informatics for Pupils of All Ages. Role of Programming and Algorithmics", "National Concepts of Teaching Computer Science. Research on Informatics in Schools", "Teacher Education in Informatics", "Contests and Competitions in Informatics", "Socio-Psychological Aspects of Teaching Informatics", "Computer Tools in Teaching and Studying Informatics".

Keywords: informatics in school, teaching programming for schoolchildren, computational thinking, Bebras, gender issues.

Как известно, появление в середине 80-х годов прошлого века в старших классах школ Советского Союза обязательного для изучения предмета «Основы информатики и вычислительной техники» «сделало нашу страну одним из мировых лидеров в данной сфере общего образования» [3]. Вместе с тем **вопросы, касающиеся сравнительного анализа современного состояния отечественного и зарубежных курсов школьной информатики**, неизменно вызывают большой интерес не только у специалистов в области школьной информатики, но и у широкого круга представителей ИТ-индустрии, у вузовских преподавателей, у родителей школьников. Информация о месте информатики в современных учебных планах ряда зарубежных стран представлена нами в работе [2]; в статье [1] мы анализировали современные подходы к обучению школьников программированию. В этой статье на материалах недавней международной конференции по проблемам школьной информатики [4] мы хотим показать конкретные большие и маленькие проблемы, над которыми в настоящее время работают наши зарубежные коллеги-информатики.

ISSEP (International Conference on Informatics in Schools: Situation, Evolution and Perspectives) — Международная конференция по преподаванию информатики в школе впервые состоялась в 2005 году в городе Клагенфурт (Австрия); затем она проводилась в городах: Вильнюс (Литва, 2006), Торунь (Польша, 2008), Цюрих (Швейцария, 2010), Братислава (Словакия, 2011), Ольденбург (Германия, 2013), Стамбул (Турция, 2014), Любляна (Словения, 2015), Мюнстер (Германия, 2016), Хельсинки (Финляндия, 2017). В октябре 2018 года ISSEP принимал Санкт-Петербург.

Основное внимание на конференции уделяется образовательным целям и задачам информатики как науки и как предмета начального, среднего и профессионального образования, решению различных учебно-методических проблем и дополнительному образованию в области информатики.

Проблемное поле конференции определяется следующими линиями:

- Качественное образование в области информатики для учащихся всех возрастов.
- Интеграция тематики уроков информатики в повседневную жизнь учащихся.
- Педагогическое образование в области информатики.
- Разнообразные исследования в области преподавания информатики в школах.
- Компьютерные инструменты в преподавании и изучении информатики.

Доступ к материалам конференции за все годы ее существования можно получить по ссылкам в рубрике «История» на сайте: <http://issep2018t.ipospb.ru/>

В Санкт-Петербурге работа конференции состояла из двух частей:

- учительский день, ориентированный на широкую русскоговорящую аудиторию (учителя информатики Санкт-Петербурга и других регионов Российской Федерации);
- основные мероприятия конференции, проводившиеся на английском языке.

В рамках учительского дня свой опыт представили как ведущие отечественные (Л. Л. Босова, Н. В. Макарова, Н. В. Матвеева, К. Ю. Поляков, Д. И. Павлов и др.), так и зарубежные специалисты.

Глава отдела преподавания информатики в Университете Коменского (Братислава, Словацкая Республика) Иван Калаш выступил с докладом «Программирование в начальной школе: исключительная возможность... и исключительная задача».

Директор Института информационных систем Технического университета Вены (Австрия) Гералд Фучек выступил с докладом «Изучение вычислительного мышления».

Профессор факультета математики и информатики Вильнюсского университета (Литва), основатель Международного конкурса по информатике Vebbras Валентина Дагене сделала доклад на тему «Продвижение образования в области информатики через конкурс Бобер».

Основные мероприятия ISSEP-2018 разворачивались по шести секциям:

1. Фундаментальное обучение информатике учащихся всех возрастов. Роль программирования и алгоритмизации.
2. Национальные концепции преподавания информатики. Исследования по информатике в школе.
3. Педагогическое образование в области информатики.
4. Конкурсы и соревнования по информатике.
5. Социально-психологические аспекты преподавания информатики.
6. Компьютерные инструменты в преподавании и изучении информатики.

Представим коротко содержание некоторых докладов по вышеперечисленным секциям, дающих достаточно полное представление о том, как развивается школьная информатика за рубежом.

Секция 1. Фундаментальное обучение информатике учащихся всех возрастов. Роль программирования и алгоритмизации (Sustainable Education in Informatics for Pupils of All Ages. Role of Programming and Algorithmics)

Иван Калаш (Ivan Kalas, Словакия) представил доклад «Развитие навыков контроля при раннем обучении информатике» («Exploring Control in Early Computing Education») об изучении возрастающей познавательной сложности различных уровней контроля, осуществляемого учащимися при программировании Эмиля — виртуального персонажа на экране; привел примеры

заданий для обучения программированию в третьем классе начальной школы.

Жаклин Стауб (Jacqueline Staub, Швейцария) в докладе «Самостоятельное исправление ошибок программирования, допущенных учащимися начальной школы» («Autonomous Recovery from Programming Errors Made by Primary School Children») обратила внимание на уникальные возможности занятий по программированию для развития семантического и синтаксического восприятия, в том числе у детей младшего школьного возраста. Была представлена авторская методика, позволяющая выявить и проанализировать сущность разного типа ошибок, допускаемых школьниками при программировании, а также представлен опыт интеграции этой методики в учебные программы по программированию в среде Logo.

Ангелика Херрера Лойо (Angelica Herrera Loyola, Швейцария) в докладе «Влияние обучения программированию на эффективность школьного обучения в начальной и средней школе» («Effects on the School Performance of Teaching Programming in Elementary and Secondary schools») познакомила собравшихся с некоторыми аспектами обучения программированию подростков 10–13 лет в частных школах Мексики, Швейцарии и Колумбии, курируемых Высшей технической школой Цюриха. Были изложены результаты наблюдений в таких аспектах, как мотивация детей и подростков к программированию, успехи в овладении языком программирования, включая усиление применения математического компонента как основы логического мышления.

Сильвия да Роса Зипитриа (Silva da Rosa Zipitria, Уругвай) в докладе «Учащиеся учат компьютер, как играть в игру» («Students Teach a Computer How to Play a Game») представила свой опыт обучения школьников программированию, основанный на проведении трех циклов (блоков) занятий. В первом блоке учащиеся играют в простую компьютерную игру под названием LumberJack («Лесоруб»), после чего они описывают правила этой игры в виде алгоритма на естественном языке. Во втором блоке ученикам предлагается разработать фрагменты компьютерной игры. Третья (заключительная) часть обучения предполагает написание всей программы на языке TurtleBot. В результате учащиеся могут более четко осознать связь между алгоритмом и тем, как он кодируется и выполняется на языке программирования.

Ноа Рагонис (Noa Ragonis, Израиль) в докладе «Диагностический инструмент для оценки восприятия и понимания учащимися понятия объекта с помощью конструкции “this”» («A Diagnostic Tool for Assessing Students’ Perceptions and Microconceptions Regards the Current Object “This”») обратила внимание на трудности учеников старших классов, связанные с пониманием концепции объекта в объектно-ориентированном программировании (ООП). Она представила диагностический инструмент (опросник), направленный на выявление уровня восприятия и понимания учениками понятия объекта посредством конструкции «this». Инструмент содержит пять вопросов, каждый из которых посвящен различным аспектам программирования. Диагностический инструмент основан на известных подходах к обучению, включая таксономию Блума; он состоит из закрытых вопросов с просьбой объяснить готовые ответы, а также открытых вопросов.

Секция 2. Национальные концепции преподавания информатики.

Исследования по информатике в школе (National Concepts of Teaching Computer Science. Research on Informatics in Schools)

Арно Пастернак (Arno Pasternak, Германия) в докладе «Стандарты обучения информатике на ступени высшего образования в Германии» («Standards for Higher Secondary Education for Computer Science in Germany») представил ситуацию со стандартизацией в области школьного образования в Германии. Почти 10 лет тому назад под влиянием результатов исследований PISA в Германии была осуществлена стандартизация таких основных предметов, как математика, родной язык, иностранный язык и естествознание. Другие предметы процессы стандартизации в тот период, к сожалению, не затронули. В 2013 году немецкое общество преподавателей информатики в инициативном порядке создало рабочую группу для разработки стандартов по информатике в старшей школе; в 2016 году такой стандарт был создан и опубликован. Ожидается, что в ближайшем будущем этот документ окажет положительное влияние на преподавание информатики в различных регионах Германии.

Анна Морпурго (Anna Morpurgo, Италия) в докладе «Основная учебная программа по информатике для итальянского обязательного образования» («A Core Informatics Curriculum for Italian Compulsory Schools») сообщила о том, что Итальянский межвузовский консорциум по информатике (CINI) в сотрудничестве с Академической ассоциацией исследователей в области информатики (GRIN) и компьютерной инженерии (GII) недавно предложили учебную программу по информатике, обязательную для всех учащихся начальных и средних школ, в которой изложили основную мотивацию включения информатики в учебный план и возможную стратегию для обеспечения эффективности ее внедрения в школе.

Наташа Гргурина (Nataša Grgurina, Нидерланды) в докладе «Второе десятилетие информатики в голландском среднем образовании» («The Second Decade of Informatics in Dutch Secondary Education») рассказала об истории становления курса информатики в своей стране. В 1998 году информатика была предложена в качестве факультативного предмета для всех учащихся старших классов среднего общего образования и довузовского образования в Нидерландах. Основной фокус в программе был сделан не на цифровую грамотность или использование офисных приложений, а на информатику как научную дисциплину. В первое десятилетие предмет столкнулся с растущими проблемами, борясь за признание и необходимые ресурсы. В 2007 году учебная программа была слегка оптимизирована, но не обновлена. В настоящее время в стране разработаны и используются программы подготовки учителей информатики; в 2019 году должна быть введена новая учебная программа по информатике для школ.

На этой секции был предложен и доклад одного из авторов данной статьи Л. Л. Босовой — «Сравнительный анализ содержания российского школьного курса информатики и тематики международного кон-

курса “Бобер”» («Comparative Analysis of the Content of School Course of Informatics in Russia and Subjects of the International Competition Bebras»), в котором было отмечено, что традиционно курс информатики в российской школе выстраивается вокруг следующих сквозных содержательных линий:

- информация и информационные процессы;
- представление информации;
- компьютер;
- моделирование и формализация;
- алгоритмизация и программирование;
- информационные технологии;
- компьютерные телекоммуникации;
- социальная информатика.

В целом содержание курса информатики стабильно, его фундаментальная составляющая является основой государственной итоговой аттестации выпускников основной и старшей школы. Вместе с тем ученики, их родители, а также представители высшей школы и ИТ-индустрии выражают озабоченность по поводу содержания современного школьного курса информатики, справедливо полагая, что эта дисциплина обладает значительно большим потенциалом для освоения школьниками таких ключевых компетенций цифровой экономики, как базовое программирование, основы работы с данными, коммуникация в современных цифровых средах.

С 2012 года российские школьники успешно принимают участие в международном дистанционном конкурсе по информатике «Бобер», предметное поле которого (алгоритмы и программирование; данные и структуры данных; компьютерная архитектура и процессы; коммуникации и сетевое взаимодействие; системы и общество) во многом созвучно содержательным линиям российского курса школьной информатики. Основные различия кроются в подходах к представлению ключевых понятий, идей, методов и алгоритмов. Материалы международного конкурса «Бобер» могут стать основой для модернизации российского школьного курса информатики за счет расширения теоретической базы курса (например, более полного представления вопросов, связанных с графами), а также обеспечения единства его теоретической и практической составляющих.

Секция 3. Педагогическое образование в области информатики (Teacher Education in Informatics)

Ноа Рогонис (Noa Ragonis, Израиль) в докладе «Вычислительное мышление: формируя восприятие у учителей, преподающих разные дисциплины» («Computational Thinking: Constructing the Perceptions of Pre-service Teachers from Various Disciplines») отмечает важность понятия «вычислительное мышление», трактовка которого впервые была предложена Жанетт Винг. Формирование вычислительного мышления является обязательной частью учебной программы в разных странах даже для дошкольного возраста. Знать сущность понятия «вычислительное мышление» и основные подходы к его формированию должны учителя, преподающие самые разные дисциплины. Для нескольких групп таких учителей, уже имеющих педагогическое образование (науки для средней школы; гуманитарные и социальные науки для средней школы; различные дисциплины для начальной

школы), предлагается курс, позволяющий им попробовать «на себе» учебные мероприятия, рекомендованные для реализации с их будущими учениками.

Ибрагим Рахими (Ebrahim Rahimi, Нидерланды) в докладе «Изучение развития методической грамотности учителей, участвующих в программе МООС по программированию Scratch» («Investigating the Pedagogical Content Knowledge Development of Teachers Attending a MOOC on Scratch Programming») представил результаты позитивных изменений в знании педагогического контента (методической грамотности) голландских учителей, преподающих информатику в начальной и средней школе, участвующих в МООС по программированию на Scratch.

Виолетта Лонати (Violetta Lonati, Италия) в докладе «Информатика и вычислительное мышление: профессиональное развитие учителя, основанное на социальном конструктивизме» («Informatics and Computational Thinking: a Teacher Professional Development Proposal Based on Social-Constructivism»), исходя из опыта работы со школьниками, где конструктивистские стратегии позволяют им изучать информатику как научную дисциплину, а также развивать навыки вычислительного мышления и решения проблем, обращает внимание на то, что преподавание информатики в соответствии с социально-конструктивистским подходом может стать важным направлением профессионального развития для учителей.

Изабелла Коррадини (Isabella Corradini, Италия) в докладе «Исследование взглядов итальянских учителей начальных классов на кодирование и программирование» («An Investigation on Italian Primary School Teachers' View on Coding and Programming») приводит результаты исследования, показывающего, что недостаточно подготовленные преподаватели придерживаются ошибочных представлений относительно информатики и связанных с ней понятий. Учитывая внимание широкой общественности и средств массовой информации к «кодированию» в школах, которое в настоящее время преподается в основном недостаточно подготовленными преподавателями, действия в области профессионального развития, направленные на изучение научных принципов и методов информатики, являются первоочередной задачей для повышения эффективности обучения в школах.

Секция 4. Конкурсы и соревнования по информатике (Contests and Competitions in Informatics)

В докладе Кристине Лутц (Christine Lutz, Германия) «Познавательное развитие по Пиаже в задачах Bebras: описательный анализ заданий с учетом возрастных групп» («Piaget's Cognitive Development in Bebras Tasks — A Descriptive Analysis by Age Groups») говорится о том, что проект Bebras, начавшийся как небольшой конкурс по информатике в Восточной Европе, в настоящее время хорошо известен во всем мире; в Германии конкурс проводится с 2015 года. Немецкие специалисты подвергли тщательному анализу задачи конкурса на предмет их соответствия возрасту участников. Они установили, что как текст задач, так и изображения различаются по возрастным группам. Их сложность возрастает с возрастом обучающихся, демонстрируя четкое разделение стадий познавательного развития Пиаже между учащимися начальной и средней школы.

Мичал Уинцзер (Michal Winczer, Словакия) в докладе «Гендерные различия при решении заданий с графиками — существуют ли они и в заданиях Bebras для старших классов?» («Gender Differences in Graph Task — Do They Exist in High School Bebras Categories Too?») обратил внимание на гендерные различия в возможностях младших (15–17 лет) и старших (17–19 лет) учащихся в решении задач со схемами и графиками. В результате проведенного анализа соответствующих заданий конкурса была обнаружена значительная разница в коэффициентах успеха решения задач между девочками и мальчиками в 38 из 65 проанализированных графических задач. В 35 заданиях мальчики были значительно более успешными, но в то же время были задачи и с менее высоким уровнем успешности. Кроме того, были выявлены несколько задач с противоречивыми результатами в отношении успеха решений для девочек и мальчиков. Важно понимать методы, используемые девочками и мальчиками при решении таких задач. Это позволит учителям минимизировать гендерные различия на занятиях по информатике.

Лусия Будинска (Lucia Budinská, Словакия) в докладе «Различия между результатами словацкого и чешского конкурса “Бобер” среди участников 9–10 лет» («Differences Between 9–10 Years Old Pupils' Results from Slovak and Czech Bebras Contest») отметила, что в целом системы образования в Чехии и Словакии очень похожи, но в Словакии реформа образования (вместе с реформой учебной программы по информатике) была реализована несколько лет назад, а в Чехии она все еще находится в стадии подготовки. Информатику в Словакии преподают начиная с начальной школы, в отличие от чешской модели, где она появляется только в некоторых типах средних школ. Тем не менее обе страны организуют конкурс «Бобер». Анализ результатов этих соревнований в младшей возрастной категории показал, что словацкие ученики четвертого года обучения более успешны, чем их чешские сверстники в области цифровой грамотности и в задачах, связанных с алгоритмами. В логических задачах, задачах, связанных с графами, существенных различий между результатами словацких и чешских школьников не установлено.

Секция 5. Социально-психологические аспекты преподавания информатики (Socio-psychological Aspects of Teaching Informatics)

Карстен Шулте (Carsten Schulte, Германия) в докладе «Оценка учебной информатики в начальном образовании с точки зрения учителей и учащихся» («Evaluation of Learning Informatics in Primary Education. Views of Teachers and Students») сообщает о результатах оценки распределенного проекта в Германии (Северный Рейн-Вестфалия), в рамках которого учеными и практиками были совместно разработаны три модуля по информатике для начальной школы и соответствующие материалы для обучения учителей. Результаты его внедрения показывают, что изучение информатики в начальном образовании возможно; все участники образовательных отношений в целом считают его полезным и увлекательным. При этом ученикам и преподавателям начальной школы было трудно понять более широкую картину образования в области информатики

и, таким образом, связать преподаваемые темы с повседневным опытом.

Франческа Агатолио (Francesca Agatolio, Италия) в докладе «Как амбициозная учебная программа по информатике может повлиять на алгебраическое мышление детей младшего школьного возраста» («How an Ambitious Informatics Curriculum Can Influence Algebraic Thinking of Primary School Children») представляет авторский вариант учебной программы по информатике, реализованной в отдельной начальной школе. В течение одного учебного года детям предложили использовать разные языки программирования, в том числе текстовые, и они стали уверенными в работе со многими наборами учебных роботов, таких как LEGO WeDo, MBot и Arduino. Дети уверенно оперировали важным для программирования понятием переменной, даже если они еще не получили соответствующего опыта на уроках математики. Установлено, что компетенции, сформированные на уроках информатики, оказали положительное влияние на формирование у учащихся алгебраического мышления и на введение некоторых сложных математических понятий.

Секция 6. Компьютерные инструменты в преподавании и изучении информатики (Computer Tools in Teaching and Studying Informatics)

Ирина Горбунова (Irina Gorbunova, Россия) в докладе «Музыкальные компьютерные технологии в области информатики и музыки в школе для детей с глубокими нарушениями зрения: из опыта» («Music Computer Technologies in Informatics and Music Studies at School for Children with Deep Visual Impairments: From the Experience») отмечает, что освоение музыкальных компьютерных технологий (МКТ) детьми с глубоким нарушением зрения имеет ряд характерных особенностей, которые наиболее ярко проявляются в начальный период изучения информатики. С помощью современных компьютерных технологий (синтезаторов речи, дисплеев Брайля и т. д.), а также благодаря возможности использования «горячих» клавиш слепые ученики быстро овладевают многими МКТ-программами. Предлагается рассмотреть прикладное программное обеспечение — инструмент, который был бы доступен для зрячих и слепых и не создавал бы трудностей в работе с ним в разных ситуациях, обеспечивая обмен информацией между учителем и слепым учеником или же слепым учителем и обычным/видящим учеником.

Анна Дронзик (Anna Dronzik, Россия) в докладе «Компьютерное моделирование задачи секретаря и его интересные результаты» («Computer Modeling of Secretary Problem and its Interesting Results») рассказала о несложном обобщении задачи о разборчивой невесте (в англоязычной литературе известной под названием «задача секретаря») в качестве исследовательской работы для учащихся средней школы. Задача заключается в следующем. Придирчивая невеста выбирает жениха. Женихи приходят к ней один за другим, и она может либо выйти замуж за того, который просит ее руки сейчас, либо отказать навсегда. Невеста умеет сравнивать любых двух женихов. Как ей действовать? Решение этой задачи известно: ей нужно отказать первым n/e (где e — основание натурального логарифма) женихам и выйти замуж за следующего, который будет лучше всех, кому она отказала. Такая стратегия максимизирует вероятность выбора лучшего жениха. Но что если мы не перфекционисты и хотим не лучшего жениха, а, скажем, жениха из лучших трех? Оказывается, тогда вероятность добиться желаемого существенно выше и ждать придется меньше. Этот эмпирический результат школьники смогли получить с помощью компьютерного моделирования.

* * *

Завершая рассмотрение материалов конференции ISSEP-2018, можно отметить стремительное развитие школьной информатики за рубежом и большой интерес, проявляемый во многих странах к становлению современных по целям и содержанию курсов информатики для школьников, а также приоритетное внимание в зарубежных курсах информатики к формированию вычислительного мышления обучающихся (computational thinking).

Список использованных источников

1. Босова Л. Л. Как учат программированию в XXI веке: отечественный и зарубежный опыт обучения программированию в школе // Информатика в школе. 2018. № 6.
2. Босова Л. Л. Школьная информатика в России и в мире // Информатизация образования и науки. 2018. № 3 (39).
3. Хеннер Е. К. Предмет «Информатика»: межстрановые сопоставления и перспективы развития // Информатика и образование. 2016. № 10.
4. Pozdniakov S. N., Dagiene V. (eds.) Informatics in Schools. Fundamentals of Computer Science and Software Engineering // 11th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2018, St. Petersburg, Russia, October 10–12, 2018, Proceedings. Series: Theoretical Computer Science and General Issues. Springer, 2018. <https://www.springer.com/gp/book/9783030027490>

Д. А. Богданова,

Федеральный исследовательский центр «Информатика и управление» Российской академии наук, г. Москва

ФОРУМ ЕВРОСОЮЗА ПО ДЕТСКОЙ ИНТЕРНЕТ-БЕЗОПАСНОСТИ SIF 2018

Аннотация

В статье представлен обзор мероприятий форума Евросоюза, посвященного вопросам детской интернет-безопасности Safer Internet Forum (SIF) 2018, который состоялся в Брюсселе, Бельгия, в ноябре 2018 года. Описаны рассмотренные на форуме актуальные интернет-риски, которым подвергаются пользователи Всемирной паутины, особенно дети.

Ключевые слова: самоидентификация, социализация, дипфейк, deepfake, персональные данные, онлайн-оффлайновая дихотомия, datadating brokers.

DOI: 10.32517/2221-1993-2018-17-9-7-9

Контактная информация

Богданова Диана Александровна, канд. пед. наук, ст. научный сотрудник Института кибернетики и образовательной информатики Федерального исследовательского центра «Информатика и управление» Российской академии наук, г. Москва; *адрес:* 119333, г. Москва, ул. Вавилова, д. 44, корп. 2; *телефон:* (499) 135-62-60; *e-mail:* d. a.bogdanova@mail.ru

D. A. Bogdanova,

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

THE EUROPEAN UNION FORUM ON KIDS' SAFETY IN INTERNET SIF 2018

Abstract

The article provides an overview of the activities of the European Union Forum on kids' safety in Internet Safer Internet Forum (SIF) 2018, which was held in Brussels, Belgium, in November 2018. The current Internet risks addressed by the World Wide Web users, especially children, are described.

Keywords: self-identification, socialization, deepfake, personal data, online offline dichotomy, datadating brokers.

В ноябре 2018 года в Брюсселе, Бельгия, состоялся ежегодный форум государств Евросоюза, посвященный вопросам детской интернет-безопасности [4]. Форум SIF (Safer Internet Forum) является ключевой ежегодной международной конференцией в Европе, на которой политики, исследователи, представители правоохранительных органов, молодежь, родители и воспитатели, преподаватели, представители промышленности, эксперты и другие участники встречаются для обсуждения последних тенденций, рисков и решений, связанных с безопасностью детей в интернете.

В этом году все мероприятия форума проходили под **общей темой «Влияние технологий на детей, молодежь и общество»**. Помимо участников из государств Евросоюза, а также из России, Армении, Украины в работе форума приняли участие специалисты из Нигерии, Марокко, Пакистана, Танзании, Албании, всего около 200 человек. На форуме 2018 года не было запланировано отдельных заседаний родителей-активистов, но традиционно была представлена довольно многочисленной группой молодежь, и для них была организована отдельная секция.

Программа форума включала пленарное заседание, церемонию награждения победителей конкурса образовательных ресурсов по интернет-безопасности, шесть секционных сессий и заключительную, итоговую, сессию.

Секционные заседания были посвящены обсуждению таких актуальных вопросов, как влияние технологии на самоидентификацию и личные отношения (включая некоторые реальные случаи), молодежные инициативы, проблемы и решения вокруг секстинга, логистика приема и обработки сообщений о противозаконном контенте в борьбе с сексуальной эксплуатацией детей. Каждый участник форума мог помимо пленарных заседаний посетить две секции на выбор. Все секции проходили дважды. Представленный в статье материал посвящен обзору тех заседаний, в работе которых приняла участие автор.

Форум открыл **пленарный доклад профессора психологии Линды Пападопулос (Linda Papadopoulou)**, в котором основное внимание было уделено влиянию технологий на самоидентификацию и формирование личных отношений. Профессор рассмотрела, как цифровые технологии трансформируют наше восприятие себя и наши отношения с другими.

Увеличивается понимание и признание того, что платформы, приложения и службы, которыми мы пользуемся, создавались не для того, чтобы помочь пользователям выразить себя, поддержать общение с другими, а *с целью извлечения прибыли*. И для этого разработчики данных сервисов делают все возможное, чтобы удержать пользователей, в частности детей, «на крючке». Показателем успеха каждой новой разработки является способность удерживать ребенка у компьютера как можно дольше. Из чего исходят разработчики:

- дети нуждаются в одобрении;
- они стремятся овладеть навыками;
- они хотят получить признание и ощущать себя частью группы.

Поэтому пользование продуктом должно быть максимально простым, пользователи необходимо стимулировать и напоминать ему о продукте. И платформы добиваются поставленных целей настолько успешно, что дети получают удовлетворение от общения с гаджетами гораздо большее, чем от потенциальных возможностей реального мира.

Каким образом компании добиваются этого?

Если речь идет об играх:

- игра дает детям чувство, что они являются мастерами в каком-то деле (чувство мастерства);
- дофамин — гормон радости, вырабатываемый организмом в процессе игры, — способствует формированию и закреплению привычек;

- еще совсем недавно перед разработчиками не стояла задача удерживать ребенка у компьютера как можно дольше; сегодня именно здесь происходят изменения;
- если успех игры зависит от времени, проведенного у компьютера, а не от времени, потраченного хорошо, — в этом случае создание принудительных игровых петель является неизбежным.

Если речь идет о социальных сетях, то здесь важны моменты социализации:

- потребность детей в социальном признании и боязнь отторжения по-прежнему остаются неизменными, но в социальных группах в режиме онлайн это выглядит иначе;
- подростки, естественно, стремятся к признанию сверстников, и это стремление возникает именно в том возрасте, когда они начинают использовать социальный инструмент, который предоставляют платформы (например, получение «лайков»);
- как и в играх, используются приемы психологии, направленные на удержание детей на социальных платформах;
- детей окружает новый набор элементов, о которых нужно беспокоиться, так как ребята уже усвоили, что есть социальная валюта, которая видна всем. Поэтому забота о продвижении себя не дает им отдохнуть от беспокойства о том, какими их видят другие.

Что могут сделать взрослые? Как они обеспечат безопасное путешествие во Всемирной паутине для детей, осваивающих новый цифровой мир, чего им, взрослым, делать никогда не приходилось? Здесь роль родителей и сложность вызовов, стоящих перед ними, трудно переоценить. Это необходимость бесед с детьми, необходимость установления регулирующих ограничений, воспитание сознательности — все тот же стандартный набор рекомендаций для родителей, но адаптированный к условиям меняющейся среды, в которой проводят время и взрослеют дети.

Первое секционное заседание было посвящено дипфейкам (**deepfake**). Что это за новый термин — **deepfake**? Он образован комбинацией двух терминов — **deep learning** и **fake**. Так названа техника синтеза изображения на основе системы искусственного интеллекта, используемого в машинном обучении (**deep learning**). С ее помощью можно производить совмещение и наложение выбранных фото или видео на исходные фото или видео.

С чего все началось? В ноябре 2017 года в социальной сети Reddit появилась площадка **r/deepfakes subreddit** (в феврале 2018 года она была запрещена и удалена) с представленными на ней сексуально откровенными видеороликами, в которых лица субъектов были заменены лицами знаменитостей. Как впоследствии выяснилось, эти ролики размещал программист, имевший опыт работы с системами машинного обучения. Создание этих роликов было его хобби, и он развлекался, накладывая лицо какой-либо знаменитости на изображение чужой головы. В течение месяца площадка собрала более 15 тысяч подписчиков. В новом сложившемся сообществе слово **deepfake** стало использоваться и вошло в обиход для обозначения различных фальшивых видео, сгенерированных с помощью нейронных сетей.

В январе 2018 года другой пользователь сети Reddit разработал удобное в использовании мобильное приложение, с помощью которого любой, даже не имеющий

специального компьютерного образования, человек может создавать подобные видео со своим набором данных. Весь необходимый инструмент доступен бесплатно и снабжен подробными инструкциями, позволяющими новичку освоить весь процесс. Тем самым был открыт ящик Пандоры. За прошедшее время были созданы всевозможные мемы: лицо Николаса Кейджа с телом Эми Адамс и постановка с участием комика Джордана Пила в качестве бывшего президента США Барака Обамы, который предупреждает о некоторых возможных зловещих применениях цифровых технологий [5], и многие другие.

Нельзя сказать, что аналогичные компьютерные технологии не были известны и не применялись в киноиндустрии ранее. Например, в фильме «Форест Гамп» есть эпизод, в котором герой общается с президентом США Джоном Кеннеди. Однако сейчас подобный инструмент с серьезными возможностями стал доступен практически каждому.

По мнению специалистов, в настоящее время мы вплотную подошли к границе нового мира, в котором очень просто изготовить правдоподобные видеоролики о людях, делающих то, чего они никогда не делали. И это несет миру серьезную угрозу. Хотя многие подписчики **r/deepfakes subreddit** занялись изготовлением с разными целями порнографического видео (это и порнография местности, и различной природы хейтерство, и другие причины), некоторые пользователи стали создавать видеоролики иного рода. Так, например, один пользователь разместил отснятые документальные кадры с Гитлером, совмещенные с видео президента Аргентины Маурисио Макри. На YouTube можно увидеть фальшивые видео с Дональдом Трампом, Владимиром Путиным, Хилари Клинтон.

Разнообразные примеры демонстрируют возможные далеко идущие последствия использования рассматриваемой технологии, позволяющей любому, имеющему достаточный материал, разместить в сети любое лицо в любом видео и сделать это довольно убедительно. По мнению специалистов, любой, кто может загрузить и запустить **FakeApp** (мобильное приложение), может создавать подобные видео, имея лишь один-два высококачественных видеоролика или несколько фотографий лиц, которые он хочет подделать. Утверждается, что **FakeApp** — это настольное приложение, разработанное сообществом для запуска алгоритма **deepfake** без установки Python, Tensorflow и т. д. И все, что нужно иметь, — это хороший графический процессор, которого требуют высококачественные 3D-видеоигры, и поддержку CUDA (параллельной вычислительной платформы и модели программирования, разработанной корпорацией NVIDIA). Если у пользователей нет соответствующего графического процессора, его можно арендовать с помощью таких сервисов, как Google Cloud Platform. Запуск всего процесса, от извлечения данных до кадрового преобразования одного образа в другой, занимает от восьми до 12 часов.

Размывая границы между фактом и вымыслом, **deepfake**-технологии могут подорвать доверие общества к записанным изображениям и видео как объективным отражениям реальности. В настоящее время еще можно невооруженным глазом идентифицировать значительную часть подделок. Но, как и всё в цифровом мире, улучшения происходят экспоненциально. И там, где технология не сможет быть убедительной, двойственность подтверждения скомпенсирует этот недостаток.

Распространение дезинформации через социальные сети является одним из серьезнейших глобальных

рисков для нашего будущего. Опровержение дезинформации представляет очень серьезную проблему, а борьба с фальшивыми новостями оказалась совсем не простой [2]. К тому же, если содержание будет соответствовать нашим убеждениям, нас, скорее всего, не насторожит неестественное выражение лица на видео или приглушенный голос.

Политики должны проявить инициативу, а правительства разработать комплекс мер для противостояния новым угрозам. Соответствующие меры могут включать упреждающее сообщение о наказаниях — как производителям, так и издателям вредоносного контента, а также четкие протоколы для обеспечения их соблюдения.

Однако не менее важной мерой противодействия угрозам был бы охват традиционных и цифровых медиaplatform, чтобы кардинально переосмыслить систему безопасности вокруг онлайн-контента. Возникает вопрос об организации законодательной защиты граждан от действий указанного рода. Ведь только в 2015–2016 годах пользователи выгрузили на Google Photo 24 биллиона (!) селфи. Несложно представить, что можно создать с помощью любительской программы, используя фотографии человека, которого хотят скомпрометировать. Крупные корпорации сейчас вкладывают огромные средства в разработку программ, позволяющих отличить реальное видео от подделки. А пока что специалисты по безопасности напоминают об известной возможности защитить свои фотографии во Всемирной паутине — использовать водяные знаки, а еще лучше — не размещать фотографии вовсе [1].

Вторая секция, которую посетила автор статьи, была посвящена **персональным данным**. Первоначально от участников, зарегистрировавшихся на эту секцию, попросили письменное согласие на исследование информации, которую можно найти о них в интернете, используя только легальные средства поиска. Впоследствии предполагалось, что найденная информация будет обнародована на секционном заседании. В случае обнаружения «деликатной» информации у участника будет испрошено дополнительное согласие на обнародование найденного. В действительности обещанного публичного обнародования сделано не было. Заинтересованным участникам было предложено по окончании секции подойти к ведущему и ознакомиться с результатом его поисков. Относительно автора ничего найдено не было, возможно, отчасти потому, что основной материал автора представлен на русском языке. У нескольких участников, как было видно, список найденного выглядел внушительным, но ничего обнародовано не было.

На этой секции было рассказано об инструментах, которыми пользуются профессиональные поисковики. Поиск может быть многоступенчатым. Было рассказано, какие сайты являются основными держателями персональных данных. Так, например, Amazon имеет огромный банк данных по своим покупателям, Google держит все данные по сайтам знакомств.

Данными торгуют, как товаром. Занимаются торговлей так называемые *datadating brokers* (брокеры данных).

На заседании секции приводились примеры рекламных объявлений о продаже данных и цен на эти данные. Например, объявление о предвесенней распродаже гласило: «87 мужчин и 80 женщин, курящие, выпивают в компании, по случаю, протестанты», и данные о них продавались за 95,19 долларов. А далее фрагмент данных: семейное положение, род занятий, религиозная принадлежность, рост, телосложение, цвет глаз, цвет волос и т. д. Для человека, далекого от сайтов знакомств, приведенные примеры выглядели интересными, с одной стороны, и пугающими, с другой.

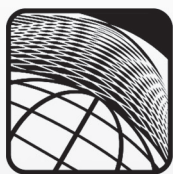
Регистрируясь на каком-либо сайте, например, бронируя гостиницу или совершая покупку, человек не представляет, что его данные пополняют базу владельца сайта, и не может со стопроцентной уверенностью сказать, что его право на неприкосновенность персональных данных не будет нарушено. К тому же в сети действует онлайн-оффлайновая дихотомия, когда пользователь сети подсознательно предполагает, что с его данными онлайн-продавцы обращаются так же, как в обычном магазине, а это не так [3].

По тому, какой объем данных и за какие деньги предлагался, можно примерно представить, насколько это серьезный вид бизнеса. Представленные на секциях презентации наглядно продемонстрировали, что то, о чем пишут специалисты по безопасности, реально и очень серьезно.

Форум SIF ежегодно знакомит с новыми тенденциями, представляет новые риски и предлагает возможные пути противостояния этим рискам. В 2014 году автору довелось выступить на форуме и рассказать о совместной работе с Институтом открытого образования Архангельской области по обучению детей интернет-безопасности. Фактически SIF — это площадка для обмена мнениями, опытом, для установления новых контактов, это источник новых идей и профессионального совершенствования. Хотелось бы, чтобы наша педагогическая общность тоже была в курсе современных реалий в ситуации с детской безопасностью во Всемирной паутине.

Список использованных источников

1. Богданова Д. А. О создании и мониторинге цифровой идентичности // Современные информационные технологии в образовании: Материалы XXVIII Международной конференции (Троицк — Москва, 27 июня 2017 года). М.: Московский издательско-полиграфический колледж им. И. Федорова, 2017. С. 120–122.
2. Богданова Д. А. Об одном глобальном риске и способах борьбы с ним // Народное образование. 2018. № 3–4. С. 191–196.
3. Богданова Д. А., Бровка Н. В. Дружественный Интернет: каким его видят британские школьники // Информатика и образование. 2017. № 10. С. 54–59.
4. Safer Internet Forum. The impact of technology on children, young people and society // Better Internet for Kids. 2018. November 20. <https://www.betterinternetforkids.eu/web/portal/policy/safer-internet-forum>
5. You Won't Believe What Obama Says In This Video! // YouTube. 2018. April 17. <https://www.youtube.com/watch?v=cQ54GDm1eL0>



Д. В. Лисин,

гимназия имени Н. В. Пушкина, г. Москва, г. о. Троицк

ФИЛОСОФСКО-МИРОВОЗЗРЕНЧЕСКИЕ И МЕТАПРЕДМЕТНЫЕ АСПЕКТЫ ШКОЛЬНОГО КУРСА ИНФОРМАТИКИ В УСЛОВИЯХ РЕАЛИЗАЦИИ ФГОС

Аннотация

Рассмотрен потенциал школьного курса информатики в аспекте достижения поставленных ФГОС личностных и метапредметных результатов обучения в основной школе. Показано, что школьный курс информатики обладает значительным философско-мировоззренческим и метапредметным потенциалом.

Ключевые слова: информатика, философия, психология, социология, управление, мировоззрение, воспитание, метапредметные результаты.

DOI: 10.32517/2221-1993-2018-17-9-10-13

Контактная информация

Лисин Дмитрий Валерьевич, канд. физ.-мат. наук, учитель информатики гимназии им. Н. В. Пушкина, г. Москва, г. о. Троицк; *адрес:* 108840, г. Москва, г. о. Троицк, ул. Школьная, д. 10; *телефон:* (495) 851-03-11; *e-mail:* lisin_maz@mail.ru

D. V. Lisin,
Gymnasium named after N. V. Pushkov,
Moscow, Troitsk

PHILOSOPHICAL AND INTERDISCIPLINARY ASPECTS OF THE SCHOOL COURSE OF INFORMATICS

Abstract

The potential of the school course of informatics in the aspect of achieving the set of personal and metasubject learning outcomes in elementary school is considered. It is shown that the school course of informatics has a significant philosophical, ideological and metasubject potential.

Keywords: informatics, philosophy, psychology, sociology, management, worldview, education, metasubject outcomes.

В действующей редакции ФГОС основного общего образования [10] впервые в отечественной практике управления образованием сделана попытка определить вектор целей системы образования как «портрет выпускника школы». В качестве приоритетных компонентов вектора целей указаны:

- воспитание личности, включая формирование нравственности, мировоззрения и управленческих способностей выпускника (личностные результаты освоения ООП) (ФГОС ООО, разд. 2, п. 8 [10])^{*};
- формирование мозаичного^{**} типа мировоззрения выпускника (ФГОС ООО, разд. 2, п. 9-3, дословно: «формирование целостного мировоззрения, соответствующего современному уровню развития науки и общественной практики, учитывающего социальное, культурное, языковое, духовное многообразие современного мира» [10]).

Подобное целеполагание можно только приветствовать, однако открытым остается вопрос о следующем за целеполаганием этапе полной функции управления — формировании концепции достижения поставленных целей, т. е. ответы на основные вопросы педагогики: чему учить и как учить? Современный ЕГЭ решает очень важную задачу — он замыкает обратные связи в системе образования, но только по предметным результатам, а по личностной и метапредметной областям обратные связи в системе пока разомкнуты^{***}.

Поскольку предмета «Формирование нравственности и мировоззрения» в учебных планах современных учебных заведений нет, да и невозможно воспитать личность и сформировать ее мировоззрение в рамках какого-либо отдель-

^{*} Следует отметить, что воспитание личности указано первым приоритетом, что должно окончательно прекратить спекуляции на тему, является ли образование «государственной услугой» или общественно значимым благом. Ровно об этом же недвусмысленно говорит Федеральный закон № 273-ФЗ «Об образовании в Российской Федерации» в ст. 2: «Образование — единый целенаправленный процесс воспитания и обучения, являющийся общественно значимым благом и осуществляемый в интересах человека, семьи, общества и государства, а также совокупность приобретаемых знаний, умений, навыков, ценностных установок, опыта деятельности и компетенции определенных объема и сложности в целях интеллектуального, духовно-нравственного, творческого, физического и (или) профессионального развития человека, удовлетворения его образовательных потребностей и интересов» [11].

^{**} Мозаика — цельная картина, сложенная из отдельных фрагментов, в отличие от калейдоскопа — случайного набора отдельных фрагментов.

^{***} За исключением так называемых «метапредметных диагностик» весьма сомнительного качества, по крайней мере, в той части, которая попадалась автору на глаза.

ного формального предмета, необходимо так строить предметные курсы, чтобы обеспечить согласованное достижение указанных выше целей*.

Современный тематический спектр школьного курса информатики значительно расширился со времени введения этого курса в обязательную школьную программу в 1985 году. В отличие от достаточно узконаправленного курса компьютерной грамотности «Основы информатики и вычислительной техники» (ОИВТ) учебники информатики в современной школе затрагивают такие фундаментальные области, как:

- философия (понятия: информация, процесс, методология познания и т. п.) [4, 5, 7, 9];
- социология (социальная информатика, информационные процессы в культуре) [5];
- психология (информационные процессы в психике человека как информационно-алгоритмической системе) [5, 9];
- история человечества как глобальный исторический процесс [9];
- общая теория управления (универсальные закономерности управления, информационное моделирование как основа любого управления, прямые и обратные связи и т. п.) [4, 9];
- основы теории вероятности и математической статистики, без которых вхождение в проблематику внутрисоциального управления** попросту невозможно;
- логика***, без которой невозможно решение задач в любой области деятельности человека;
- основы теории игр ([5], задача 26 ЕГЭ и др.), формирующие важнейшие управленческие навыки [2, с. 42].

Указанные тематические области в современных учебниках информатики затрагиваются, но, как правило, весьма поверхностно****, что значительно снижает возможности информатики как потенциально метапредметного, интегративного курса (в силу универсальности понятия «информация»), идеально подходящего для достижения указанных выше компонентов вектора целей, заданных в ФГОС. Поясним сказанное на примерах.

Философский блок вопросов открывается при первом же знакомстве с предметом информатики. Что

такое информация? Объективна ли она? Каково ее место в системе предельных обобщающих категорий? Какие вообще имеются системы предельных обобщений? Позиция современной педагогической науки по данному вопросу в наиболее краткой и лаконичной форме выражена И. В. Роберт: «Таким образом, понятие “информация” не описывается основными философскими категориями и/или понятиями “материя”, “пространство”, “время” и не сводится к этим понятиям так же, как и к производным (по отношению к основным) понятиям, например, понятиям “вещество”, “энергия”. То есть “информация” является одним из основных понятий объективной реальности, рассмотрение которого на современном этапе развития научно-технического прогресса следует осуществлять на более обобщенном уровне: “информация” — это одна из основных философских категорий» [8].

Аналогичной позиции придерживаются и авторы учебника [7], выделяя тезис «Информация *нематериальна*», однако, не показывая место информации в различных системах предельных обобщающих категорий и не рассматривая связанного с этим блока философско-мировоззренческих вопросов. К вопросам, совершенно не затронутым в рассмотренных учебниках информатики, можно отнести следующие.

1. Что такое мировоззрение и миропонимание человека с точки зрения информатики?

Одним из ответов на данный вопрос является следующее понимание. *Мировоззрение* — это система «образно-музыкальных», внелексических образов в психике человека, отражающих его знания об окружающем мире. А «образ» — это один из фундаментальных синонимов предельно обобщенного понятия об информации. Таким образом, мировоззрение человека определяется как система информационных модулей *первичной природы*, т. е. образов объективной реальности, запечатленной и упорядоченной в психике индивида непосредственно. Понятие — это связка внелексического образа и слова (кодовой комбинации в некоторой системе кодирования) [6], отсюда *миропонимание* можно определить как некоторым образом упорядоченную систему *вторичных* информационных модулей (понятий), структура организации которой отражает понимание индивидом объективных взаимосвязей явлений в объективной реальности. Подобное понимание позволяет далее выйти на анализ возможных типов мировоззрений (и сопряженных с ними миропониманий) по критерию упорядоченности информационных модулей в психике индивида и характеру их взаимосвязей, показать роль русского языка как важнейшего средства формирования миропонимания (здесь лежит очень важная метапредметная связь информатики с предметными областями «Русский язык и литература» и «Иностранные языки»).

2. Что такое нравственность человека с точки зрения информатики?

Если рассматривать психику человека как информационно-алгоритмическую систему, которая некоторым образом самоуправляется, а также управляется извне, то объективное явление, которое обычно называют «нравственность», будет соответствовать системе констант (нравственных мерил) в «операторах условного перехо-

* Указанные образовательные результаты обозначены в ФГОС как метапредметные.

** Соответственно этому осмысленное и результативное соучастие выпускника школы как гражданина государства в процессах внутрисоциального управления (гражданское общество и демократия) без *понимания* этих разделов математики попросту невозможно.

*** Следует отметить, что по сравнению с общеобразовательным курсом «Логика» на основе учебника С. Н. Виноградова и А. Ф. Кузьмина 1954 года [3] в современном школьном образовании наблюдается чудовищный регресс в части преподавания логики как элемента культуры рассуждений, а не как оторванных от жизни математических абстракций, которые в современных курсах информатики и задачах ЕГЭ мало кто из выпускников способен связать с жизнью.

**** Например, в качестве одного из компонентов вектора целей изучения предметной области «Естественно-научные предметы» указано «осознание значимости концепции устойчивого развития» ([10], разд.2, п. 11.5). Указанная предметная область наиболее полно освещена в монографии [1] и никак не освещается в рассмотренных учебниках информатики [4, 7, 9].

да» внутренней алгоритмики данной системы. Каждая из этих величин определяет результат оценки какой-либо конкретной ситуации по шкале типа «хорошо/плохо/безразлично». При этом различные значения величин этих констант, соответствующие различным их наборам, будут приводить к различным траекториям выполнения одних и тех же алгоритмов в психике. Подобный подход хорошо иллюстрирует роль нравственности в судьбе (траектории выполнения множества сопряженных алгоритмов) как отдельного человека, так и различного рода социальных групп, вплоть до масштаба всего человечества. Он также открывает возможности к осмысленному, целенаправленному управлению системой нравственных стандартов, т. е. к тому, что в различных религиозных и этических системах понимается под нравственным самосовершенствованием. Данный вопрос демонстрирует метапредметную связь информатики с вопросами воспитания и самовоспитания, религиозно-этическими и социокультурными областями знания.

3. Что такое психология с точки зрения информатики?

Как отмечено выше, психика человека является информационно-алгоритмической системой, информация и алгоритмика в которую попадают из различных источников. Соответственно этому пониманию, психология — это одна из прикладных областей изучения теории алгоритмов и область применения прочей методологии информатики, если понимать ее как науку об информации и закономерностях информационных процессов. Следует отметить чрезвычайную важность и актуальность для всех без исключения выпускников школ грамотности в области адекватной стоящим перед современным обществом задачам психологической теории*, позволяющей самостоятельно выходить на уровень осмысленного самоуправления собственной психикой, без чего достижение требуемых ФГОС личностных результатов общего образования практически невозможно. Этот вопрос раскрывает метапредметный потенциал информатики с вопросами воспитания и самовоспитания, сопряженными с ними вопросами прикладной психологии, а также выводит на область охраны здоровья, если задаться вопросом о роли психосоматики (различного рода дефективности в информационно-алгоритмическом обеспечении функционирования психики человека) в генезисе большинства заболеваний.

4. Что такое социология с точки зрения информатики?

Если психика отдельного индивида — это сложная информационно-алгоритмическая система, то общество представляет собой информационно-алгоритмическую суперсистему [2], управлять процессами в которой, будучи неграмотным в области универсальных алгоритмических закономерностей, осмысленно невозможно. А участие каждого гражданина государства в управлении им составляет сущность того, что обычно декларируется как *демократия***.

* Психология как учебный предмет была в нашей стране окончательно изгнана из курса средней школы в 1958 году.

** Таким образом, реальная демократия в обществе невозможна без всеобщей управленческой грамотности, в противном случае она неизбежно становится ширмой для различного рода закулисных манипуляций.

5. Методология познания и творчества.

Одно из базовых понятий в курсе информатики — алгоритм. Изучению различного рода алгоритмов и их практической реализации в большинстве курсов уделяется достаточно много внимания. Однако вопрос: «Существует ли алгоритм создания алгоритмов?» приводит к вопросам методологии познания и творчества, которые, по сути, являются ключевыми в достижении задач личностного развития человека, зафиксированных в ФГОС. Как создается новое знание? По каким алгоритмам оно может быть проверено на истинность (адекватность жизни)? Каковы алгоритмы творчества? Как раскрыть собственные способности к созданию нового? Вот лишь небольшой перечень жизненно важных вопросов, на которые выводит данный раздел.

* * *

Информация — одна из основных фундаментальных категорий бытия, а информационные процессы, среди которых одним из наиболее практически значимых является процесс *управления*, протекают во всех областях жизни и деятельности человека.

Основная непреходящая задача общеобразовательной школы — дать фундаментальные знания, умения и навыки, инвариантные относительно быстроменяющихся предметных областей, как основу для осмысленной деятельности человека и гражданина в обществе и государстве. В этом смысле знания методологического характера намного более эффективны, чем изучение фактологии отдельных предметных областей.

По этой причине курс школьной информатики обладает значительным потенциалом, который на данное время используется достаточно слабо. Использование философско-методологического и метапредметного потенциала информатики может оказать существенную помощь всей системе образования в решении поставленных в ФГОС общего образования задач достижения личностных и метапредметных результатов.

Список использованных источников

1. *Величко М. В.* Научно-методологические основы государственного управления в русле концепции устойчивого развития: Обеспечение безопасности: биосферно-социальной, экономической, техносферной // Санкт-Петербургский государственный политехнический университет. Санкт-Петербург, 2012. <http://elibrary.spbstu.ru/dl/2/2547.pdf/view>
2. *Величко М. В., Ефимов В. А., Зазнобин В. М.* Экономика инновационного развития. Управленческие основы экономической теории: монография. М. — Берлин: Директ-Медиа, 2015.
3. *Виноградов С. Н., Кузьмин А. Ф.* Логика: учебник для средней школы. 8-е изд. М.: Учпедгиз, 1954.
4. *Гейн А. Г., Ливчак А. Б., Сенокосов А. И., Юнгерман Н. А.* Информатика. 10 класс: учебник для общеобразовательных организаций: базовый и углубленный уровни. 3-е изд. М.: Просвещение, 2017.
5. *Гейн А. Г., Сенокосов А. И.* Информатика. 11 класс: учебник для общеобразовательных организаций: базовый и углубленный уровни. 3-е изд. М.: Просвещение, 2017.
6. *Лисин Д. В., Солдатова Е. В.* Модифицированная методика измерения уровня полноты усвоения в процессе формирования понятий // Педагогика и просвещение. 2018. № 1. С. 76–86. DOI: 10.7256/2454-0676.2018.1.22548. http://e-notabene.ru/pped/article_22548.html

7. Поляков К. Ю., Еремин Е. А. Информатика. Углубленный уровень: учебник для 10 класса: В 2 ч. Ч. 1. 5-е изд. М.: БИНОМ. Лаборатория знаний, 2016.

8. Роберт И. В. Теория и методика информатизации образования (психолого-педагогический и технологический аспекты). 2-е изд., доп. М.: ИИО РАО, 2008.

9. Угринович Н. Д. Информатика: учебник для 9 класса. 4-е изд. М.: БИНОМ. Лаборатория знаний, 2016.

10. 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)

11. Федеральный закон от 29 декабря 2012 года № 273-ФЗ (ред. от 29.07.2017) «Об образовании в Российской Федерации». http://www.consultant.ru/document/cons_doc_law_140174/

НОВОСТИ

Российские школьники завоевали 11 золотых медалей на Международном турнире по информатике в Болгарии

24 ноября в г. Шумен (Болгария) стали известны итоги X Международного турнира по информатике (IATI), в котором принимали участие школьники IX—XI классов. В турнире состязались более 200 школьников из России, Армении, Азербайджана, Болгарии, Молдавии, Польши, Румынии, Сербии и Швейцарии. Российские школьники привезли 29 медалей из 87, в том числе 11 золотых.

IATI — единственное состязание, участники которого соревнуются в двух возрастных группах — «Юниоры» до 15,5 лет и «Сениоры» старше 15,5 лет. За два дня школьники должны решить шесть задач на языке программирования C++. За Россию выступали 11 команд: шесть команд среди юниоров и пять команд — среди старшеклассников.

У российских юниорских команд пять золотых медалей. В команде старшеклассников российские школьники завоевали шесть золотых медалей.

С достойным выступлением на международном турнире российских школьников поздравила министр просвещения Российской Федерации О. Ю. Васильева.

(По материалам, предоставленным пресс-службой Министерства просвещения РФ)

Финансирование ФП «Цифровая образовательная среда» составляет 77,4 млрд рублей — Васильева

Общий объем финансирования федерального проекта «Цифровая образовательная среда» за счет средств федерального бюджета составляет 77 млрд 400 млн рублей, сообщила министр просвещения Российской Федерации Ольга Васильева. Об этом она заявила в рамках правительственного часа в Госдуме.

Министр подчеркнула, что проект направлен на решение задачи по созданию к 2024 году современной и безопасной цифровой образовательной среды, обеспечивающей высокое качество и доступность образования всех видов и уровней.

По словам Васильевой, в рамках федерального проекта планируется внедрить целевую модель цифровой образовательной среды, которая позволит создать во

всех образовательных организациях на территории Российской Федерации профили «цифровых компетенций» для обучающихся, педагогов и административно-управленческого персонала, создать и внедрить федеральную информационно-сервисную платформу цифровой образовательной среды, набор типовых информационных решений, обновить информационное наполнение и функциональные возможности официальных сайтов образовательных организаций и использовать при реализации основных образовательных программ современные технологии, в том числе технологии виртуальной и дополненной реальности.

Также запланировано создание сети из 340 центров цифрового образования для детей «IT-куб».

(По материалам федерального портала «Российское образование»)

М. В. Маркушевич,
школа № 1352, г. Москва

ЗАВИСИМОСТЬ МЕЖДУ МЕТОДИЧЕСКОЙ СИСТЕМОЙ ОБУЧЕНИЯ ИНФОРМАТИКЕ В ШКОЛЕ И ТИПОМ ИСПОЛЪЗУЕМОГО В НЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Аннотация

Проводится покомпонентный анализ элементов методической системы обучения информатике в средней общеобразовательной школе в зависимости от типа используемого в учебном процессе программного обеспечения (свободного или проприетарного). На основании проведенного исследования проводится коррекция структурной схемы методической системы обучения информатике и формулируется вывод о наличии зависимости методической системы в целом от типа применяемого в обучении программного обеспечения.

Ключевые слова: методическая система обучения, свободное программное обеспечение, информатика.

DOI: 10.32517/2221-1993-2018-17-9-14-24

Контактная информация

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

M. V. Markushevich,
School 1352, Moscow

DEPENDENCE BETWEEN THE METHODOICAL SYSTEM OF TEACHING INFORMATICS IN SCHOOL AND THE TYPE OF USED SOFTWARE

Abstract

Component analysis of the elements of the methodical system of teaching informatics in the secondary school, depending on the type of software used in the educational process (free or proprietary), is carried out. On the basis of the conducted research, the structural scheme of the methodical system of teaching informatics is corrected and a conclusion is formulated about the dependence of the methodical system on the type of software used in training.

Keywords: methodical training system, free software, freeware, methods of teaching informatics.

Обзор программного обеспечения, используемого в качестве базового в учебно-методической литературе по информатике

Современную ситуацию с преподаванием информатики в общеобразовательной российской школе можно охарактеризовать как имеющую высокий уровень программно-методического обеспечения. Действительно, в настоящее время в распоряжении школьных учителей находятся высококачественные методики преподавания информатики и разработанные на их основе учебно-методические комплексы таких авторов, как Л. Л. Босова, А. Ю. Босова, Н. В. Матвеева, К. Ю. Поляков, Е. А. Еремин, Н. Д. Угринович, Е. К. Хеннер, И. Г. Семакин, И. А. Калинин, Н. Н. Самылкина, Н. В. Макарова, и многих других.

Некоторые авторы учебно-методических комплексов указывают рекомендуемое для поддержки преподавания информатики на основе их методики программное обеспечение, другие же не делают акцента на этом вопросе, вполне справедливо полагая, что квалифицированный педагог сам может выбрать наиболее подходящий для его конкретной ситуации тип программного обеспечения. Так как на выбор типа ПО, используемого в учебно-воспитательном процессе, в каждой отдельной ситуации влияет большое количество факторов, то затруднительно рассмотреть все возможные варианты и предложить оптимальное решение для каждого из них. Отметим, что к упомянутым факторам можно отнести такие, как [15]:

- уровень квалификации преподавателя информатики;
- конфигурация компьютеров, которыми оборудован кабинет информатики;
- тип операционной системы и прикладного ПО, уже установленных на компьютерах класса информатики, а также тип и срок лицензии на данное ПО в том случае, если оно относится к проприетарному ПО;
- уровень ИКТ-компетенций учащихся данной образовательной организации;
- объем финансирования, отпущенный на приобретение платного программного обеспечения.

В качестве примера явного указания типа применяемого программного обеспечения можно привести методическое пособие для учителя информатики предметной линии учебников К. Ю. Полякова, Е. А. Еремина «Информатика» для X—XI классов общеобразовательных учреждений (углубленный уровень). Автор пособия [3] М. Н. Бородин в разделе «Требования к программному обеспечению компьютеров» приводит следующий перечень необходимого программного обеспечения с указанием ресурсов сети Интернет, откуда они могут быть скачаны:

- текстовый редактор Блокнот или Gedit и текстовый процессор Word или OpenOffice.org Writer (<http://www.openoffice.org/ru/download/index.html>);
- табличный процессор Excel или OpenOffice.org Calc (<http://www.openoffice.org/ru/download/index.html>);
- средства для работы с базами данных Access или OpenOffice.org Base (<http://www.openoffice.org/ru/download/index.html>);
- графический редактор Gimp (<http://gimp.org>);
- редактор звуковой информации Audacity (<http://audacity.sourceforge.net>);
- среда программирования «КуМир» (<http://www.niisi.ru/kumir/>);
- среда программирования FreePascal (<http://www.freepascal.org/>);
- среда программирования Lazarus (<http://lazarus.freepascal.org/>).

Обратим внимание на тот факт, что автор пособия рекомендует к использованию или только свободное программное обеспечение, или на выбор — проприетарное ПО и соответствующий ему свободный аналог.

Аналогично другие авторы — Н. Д. Угринович, Н. Н. Самылкина в примерной рабочей программе по информатике для VII—IX классов [20], посвященной методике работы с учебно-методическим комплексом Н. Д. Угриновича, в качестве рекомендуемого ПО приводят и проприетарное программное обеспечение, и — как возможную альтернативу — его свободный аналог.

На основе вышеприведенных фактов, с одной стороны, можно сделать промежуточный вывод о том, что *в настоящее время свободное программное обеспечение уверенно занимает свою нишу в области отечественной теории и методики обучения информатике в общеобразовательной школе.*

С другой стороны, возникает вопрос: **целесообразно ли вести преподавание информатики по одной и той же методике на базе как свободного, так и проприетарного программного обеспечения?**

На первый взгляд, ответ на данный вопрос очень прост и очевиден. Действительно, в чем разница в методике преподавания, например, темы «Создание и обработка текстовой информации» в проприетарном текстовом процессоре Microsoft Word и свободном текстовом процессоре OpenOffice.org Writer? Как в первом, так и во втором случае, видимо, мы будем последовательно обучать детей вводить текст, редактировать его, форматировать, создавать списки, вставлять в текстовый документ рисунки и таблицы и т. д. Вряд ли в этом плане методики будут в чем-либо различаться, кроме уже совсем мелких, технических нюансов, обусловленных пользовательским интерфейсом каждого из рассматриваемых нами текстовых процессоров.

Но здесь надо отметить, что **свободное и проприетарное программное обеспечение базируются на принципиально разных идеологиях.** Почему в данном вопросе используется термин «идеология», имеющий отношение скорее к политике, чем к науке? С точки зрения автора, в нашем случае речь идет как раз о совокупности системных упорядоченных взглядов, выражающих интересы различных социальных классов и других социальных групп, на основе которых вырабатывается концепция того или иного типа программного обеспечения, что соответствует определению термина «идеология» [8].

Свободное программное обеспечение разрабатывается сообществами программистов-энтузиастов и затем распространяется таким образом, что каждый пользователь имеет право свободно скачать данное ПО, установить его на свой компьютер и использовать неограниченный период времени, передавать дистрибутив другим пользователям, просматривать программный код, вносить в него изменения и далее распространять в измененном виде. Таким образом, **продукт интеллектуального труда коллектива единомышленников, которым является программное обеспечение, свободно передается всем заинтересованным в нем людям.** Очевидно, что процесс создания программного обеспечения — это творческий процесс, который очень верно охарактеризовал великий русский философ Николай Александрович Бердяев следующим образом: «Творческий акт есть не только движение вверх, но и движение к другому, к миру, к людям» [1]. Именно этой

ориентацией на людей можно в целом описать идеологию открытого программного обеспечения.

Интересно, что некоторые версии ОС Linux транслируют описываемую нами идеологию в название операционной системы. Например, одну из веток дистрибутива Debian Linux разработчики назвали Ubuntu, что в переводе с южно-африканского языка банту означает человечность и взаимопомощь [4].

Техническая поддержка свободного программного обеспечения обычно осуществляется на форумах, где более продвинутые пользователи добровольно помогают новичкам освоить используемое ими ПО, разобраться в его тонкостях и нюансах настройки.

Проприетарное программное обеспечение разрабатывают коммерческие корпорации, нанимающие программистов для целей написания программного кода, причем код является собственностью фирмы-разработчика и, следовательно, закрыт для изучения. ПО данного типа распространяется по закрытой лицензии, предполагающей обычно краткий бесплатный период ознакомления пользователя с предлагаемым ПО, а затем дальнейшую оплату за использование в том случае, если качество удовлетворяет пользователя. В данном случае **продукт интеллектуального труда является товаром, который, как любой другой товар, выставляют на продажу.** В ситуации, когда программное обеспечение является объектом продажи, на него автоматически распространяются свойства материальных предметов, которыми можно торговать, обмениваться, право владения и пользования которыми целесообразно охранять законодательно [17]. Техническая поддержка проприетарного программного обеспечения осуществляется отделами технической поддержки фирм-разработчиков и обычно предполагает идентификацию пользователя в плане использования им лицензионной копии ПО.

Таким образом, можно предположить, что **методики обучения информатике, основанные на программном обеспечении с такими совершенно разными подходами, как идеологии свободного и проприетарного программного обеспечения, должны иметь определенные принципиальные отличия.** Здесь важно отметить, что, исходя из приведенного выше примера с текстовыми процессорами, данные отличия будут в основном касаться не всей методики в целом, а некоторых ее определенных структурных компонентов.

Как отмечают авторы учебного пособия [11] А. А. Кузнецов, Т. Б. Захарова, А. С. Захаров, сегодня используются принципиально разные **трактовки понятия «методика обучения»** — от узкого понимания методики обучения как совокупности приемов проведения занятий до широкого понимания ее как науки об организации образовательного процесса. В указанном пособии авторы предлагают свою трактовку данного понятия, которую для информатики можно представить в следующем виде: **методика обучения информатике включает в себя, с одной стороны, научное обоснование методической системы обучения информатике и, с другой стороны, рассмотрение профессиональной деятельности учителя информатики (структуры и содержания компонентов данной деятельности).**

Таким образом, переходя от понятия «методика обучения» к более узкому понятию «методическая система обучения», А. А. Кузнецов, Т. Б. Захарова, А. С. Захаров предлагают следующее описание и структурную схему

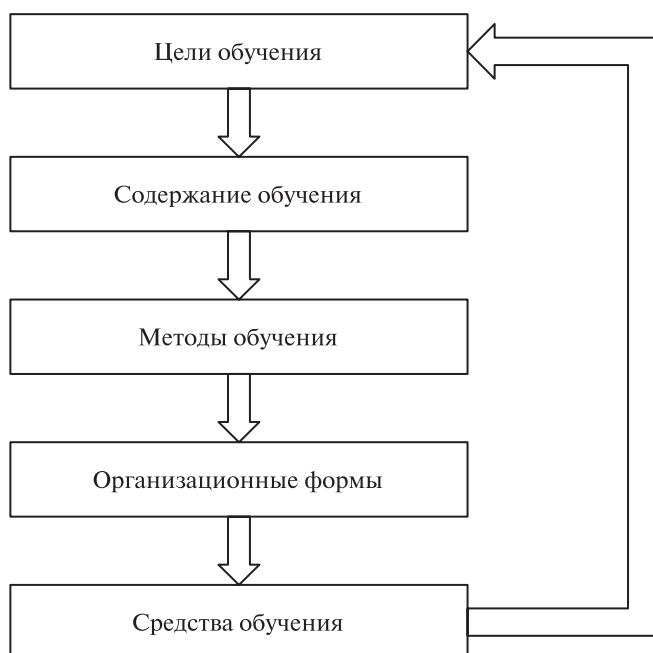


Рис. 1. Методическая система обучения
по А. А. Кузнецову, Т. Б. Захаровой, А. С. Захарову [11]

последней, представленную на рисунке 1. **Методическая система обучения информатике** — совокупность пяти иерархически связанных между собой компонентов: целей, содержания, методов, организационных форм и средств обучения [11].

Исходя из целей, поставленных в настоящей работе, далее рассмотрим, каким образом влияет выбираемая авторами идеология программного обеспечения на каждый из вышеприведенных компонентов методической системы обучения информатике, и постараемся понять, какие из компонентов существенно подвержены этому влиянию, а какие испытывают слабое влияние или не испытывают его вовсе.

Введем термин **«программно зависимый компонент методической системы»** для тех элементов, которые испытывают на себе сильное влияние выбираемого в качестве базового типа программного обеспечения, и термин **«программно независимый компонент методической системы»** для тех элементов, которые испытывают на себе слабое влияние выбираемого в качестве базового типа программного обеспечения или не испытывают его вообще.

1. Цели обучения

Цели обучения в ФГОС общего образования определены как формирование совокупности личностных, метапредметных и предметных результатов. В современных условиях в первую очередь *развитие личности* рассматривается как смысл и цель среднего общего образования. В связи с этим рассмотрим, какое влияние оказывает выбираемая нами идеология программного обеспечения на развитие личности учащегося.

Как уже отмечалось нами в предыдущих работах, тип программного обеспечения, применяемый в качестве базового при обучении школьников информатике, оказывает непосредственное влияние на развитие личности учащихся через положительное воздействие на их духовно-нравственную сферу [22], а именно:

- формирование у учащихся ментальности квалифицированного члена современного информационного общества, различающего наличие потенциальных информационных угроз и готового противостоять им;
- воспитание информационной культуры и этики учащихся, одной из основных составляющих которых является компетенция квалифицированного выбора типа ПО, используемого для решения конкретных пользовательских задач;
- раскрытие творческих способностей личности учащегося путем освобождения его от скрытых духовных деформаций, обусловленных наличием навязанных и многократно закрепленных неоптимальных алгоритмов выбора проприетарного программного обеспечения иностранного производства.

Таким образом, можно отметить, что *в случае применения свободного программного обеспечения цели обучения информатике в школе в плане развития личности учащегося становятся очевидно шире, чем в случае использования проприетарного ПО и, следовательно, компонент методической системы «цели обучения» является программно зависимым.*

2. Содержание обучения

В результате перехода от знаниевого к системно-деятельностному подходу в настоящее время понятие «содержание обучения» претерпело сильные изменения и в редакции А. А. Кузнецова, Т. Б. Захаровой, А. С. Захарова может быть сформулировано следующим образом: *под содержанием обучения сегодня понимают характеристику осваиваемой системы знаний и способов деятельности, обеспечивающих реализацию сформулированных целей обучения в виде требований к образовательным результатам* [11].

Исходя из принципа проектирования образовательного процесса «от образовательных результатов к содержанию образования», постараемся рассмотреть в таблице 1, какими именно темами и модулями может быть дополнено содержание образования по информатике для достижения личностных результатов освоения основной образовательной программы основного общего образования при условии миграции учебного процесса на свободное программное обеспечение.

В таблице 2 рассмотрим, каким образом может быть дополнено и скорректировано содержание образования по информатике для достижения метапредметных результатов освоения основной образовательной программы основного общего образования при условии миграции учебного процесса на свободное программное обеспечение.

В таблице 3 рассмотрим, каким образом может быть дополнено и скорректировано содержание образования по информатике для достижения предметных результатов области «Информатика» (углубленный уровень) освоения основной образовательной программы основного общего образования при условии миграции учебного процесса на свободное программное обеспечение.

Резюмируя все вышесказанное, можно отметить, что *раздел методической системы обучения информатике «Содержание обучения» является в достаточно серьезной степени программно зависимым.*

Таблица 1

Соответствие между личностными результатами и содержанием образования

№ п/п	Личностный результат	Содержание обучения	Комментарий
1	Воспитание российской гражданской идентичности: патриотизма, уважения к Отечеству, прошлому и настоящему многонационального народа России	Целесообразно было бы дополнить модуль «Аппаратные и программные средства ИКТ» темой « Типы программного обеспечения, их преимущества и недостатки » и в содержании образования по данной теме рассматривать подробную сравнительную характеристику, а также преимущества и недостатки свободного и проприетарного, отечественного и импортного программного обеспечения	В случае миграции учебно-воспитательного процесса на свободное программное обеспечение отечественного производства уже сам этот факт станет одним из существенных факторов гражданско-патриотического воспитания учащихся
2	Формирование ответственного отношения к учению, готовности и способности обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию, осознанному выбору и построению дальнейшей индивидуальной траектории образования на базе ориентировки в мире профессий и профессиональных предпочтений	Содержание образования в девятом классе разумно дополнить предпрофильным модулем « Введение в ИТ-специальность », в котором рассматривать классификацию современных ИТ-специальностей, а также связь изучаемого на уроках информатики материала с некоторыми из данных специальностей. Обозначенный подход, совместно с идеологией СПО, дополнительно мотивировал бы учащихся в плане обучения информатике в школе по предлагаемой методике	Использование свободного программного обеспечения в образовании косвенно способствует формированию ментальности внутренне свободной, сознательной и самостоятельной личности , что связано с самой идеологией СПО, о которой говорилось выше
3	Освоение социальных норм, правил поведения, ролей и форм социальной жизни в группах и сообществах, включая взрослые и социальные сообщества	Целесообразно было бы ввести в содержание образования интегрированный с обществознанием модуль « Правовая информатика », в содержании которого рассматривать авторское право и смежные с ним права, а также типы лицензий, по которым распространяется тот или иной вид программного обеспечения, виды ответственности за нарушение данных лицензий	Сознательное использование учащимися свободного программного обеспечения в своей учебной и досуговой деятельности возможно только при достаточно высоком уровне гражданской ответственности и правовой грамотности в области авторского права
4	Формирование осознанного, уважительного и доброжелательного отношения к другому человеку, готовности и способности вести диалог с другими людьми и достигать в нем взаимопонимания	Содержание образования было бы разумно дополнить модулем « Информационная этика », в котором рассматривались бы моральные проблемы, возникающие в связи с развитием и применением информационных технологий. Информационные моральные дилеммы становятся особенно важными в информационном обществе. Уже сейчас информационные технологии затрагивают фундаментальные права человека, касаясь защиты авторских прав, интеллектуальной свободы, ответственности и безопасности. Информационная этика рассматривает проблемы собственности, доступа, неприкосновенности частной жизни, безопасности и общности информации	Сама идеология СПО является коммюнитарной, т. е. предполагает межчеловеческое общение членов общества, основанное на свободе, искренности, взаимопомощи и братстве
5	Развитие морального сознания и компетентности в решении моральных проблем на основе личностного выбора, формирование нравственных чувств и нравственного поведения, осознанного и ответственного отношения к собственным поступкам	На достижение данного результата в том числе нацелено обновленное содержание модулей « Правовая информатика » и « Информационная этика » разрабатываемой методики преподавания	С одной стороны, осознанное ответственное решение учащихся не использовать в своей учебной и повседневной деятельности пиратского программного обеспечения также можно отнести и к области морали, так как использование контрафактного ПО можно рассматривать как разновидность кражи интеллектуальной собственности. С другой стороны, решение учащихся не тратить родительские деньги на приобретение официальных версий проприетарного ПО также можно отнести к моральной сфере, так как это связано с заботой о материальном благополучии семьи

Таблица 2

Соответствие между метапредметными результатами и содержанием образования

№ п/п	Метапредметный результат	Содержание обучения	Комментарий
1	Использовать все возможные ресурсы для достижения поставленных целей и реализации планов деятельности; выбирать успешные стратегии в различных ситуациях	В новую методику обучения при изучении каждой новой темы необходимо вводить фрагмент, в котором кратко будут рассматриваться основные доступные в настоящее время на рынке программные средства для решения тех пользовательских задач, которые решаются в рамках изучаемой темы, а также их преимущества и недостатки	Наличие кругозора в области как свободного программного обеспечения, так и проприетарных аналогов позволяет учащемуся сделать квалифицированный выбор оптимального программного инструмента для решения данной пользовательской задачи
2	Владение навыками проектной деятельности, навыками решения проблем; способность и готовность к самостоятельному поиску методов решения практических задач	Для достижения данного результата представляется целесообразным изучение материала каждой темы строить в виде создания тематического проекта по технологии, описанной ниже в разделе 3 настоящей работы	Именно применение одного и того же программного инструмента учащимися как на школьных компьютерах, так и дома дает возможность реализации проектов по предложенной технологии. Данный подход возможен только в случае использования свободного программного обеспечения
3	Умение использовать средства ИКТ для эффективной реализации правовых и этических норм, норм информационной безопасности	На достижение данного результата направлено содержание новых модулей, описанных выше, таких как « Правовая информатика », « Информационная этика ». Кроме того, расширяется содержание модуля « Информационная безопасность » за счет рассмотрения вопросов устойчивости операционной системы в целом к вредоносному воздействию	Применение ОС Linux повышает уровень информационной безопасности деятельности учащихся за счет малой подверженности данной операционной системы вирусным угрозам. Это связано с тем фактом, что обычная работа в Linux осуществляется под пользователем с ограниченными полномочиями, а не под root (администратором). В результате этого под угрозой заражения находится только домашняя папка пользователя, а не вся файловая система
4	Умение самостоятельно оценивать и принимать решения, определяющие стратегию поведения, с учетом гражданских и нравственных ценностей	В модуле « Информационная безопасность » в виде отдельной темы необходимо рассматривать виды угроз информационной инфраструктуре России со стороны ряда иностранных государств, обусловленных использованием проприетарного программного обеспечения иностранного производства, и методы борьбы с ними	Использование свободного программного обеспечения отечественного производства является крайне важным для формирования гражданской позиции учащихся

Таблица 3

Соответствие между предметными результатами и содержанием образования

№ п/п	Предметный результат	Содержание обучения	Комментарий
1	Владение универсальным языком программирования высокого уровня (по выбору)	В случае использования свободного программного обеспечения в учебном процессе в качестве языка программирования можно выбрать процедурный язык Кумир или свободную реализацию языка Pascal — Free Pascal, в качестве объектно-ориентированного языка программирования можно применять свободную среду Lazarus	Свободный язык программирования Кумир является наиболее подходящим для использования в учебном процессе по следующим причинам: • является свободным и бесплатным, распространяется по лицензии GNU GPL; • в синтаксисе языка Кумир используются русские ключевые слова, что облегчает их запоминание учащимися; • является кроссплатформенным (для ОС Mac OS X версия находится в разработке)
2	Сформированность представлений о понятии «операционная система» и основных функциях операционных систем	В случае миграции учебного процесса на СПО рассматриваемый предметный результат будет достигаться на примере свободной операционной системы семейства Linux. Возможно использование	Использование свободной операционной системы при достижении данного предметного результата имеет некоторые преимущества по сравнению с использованием закрытой проприетарной

Окончание табл. 3

№ п/п	Предметный результат	Содержание обучения	Комментарий
		как отечественных версий Linux, таких как ROSA, Альт Образование 8, Astra Linux, так и ОС иностранного производства, таких как Linux Mint, Ubuntu, openSUSE и т. п.	ОС, обусловленные именно открытостью ее программного кода, а также всех процессов, происходящих внутри самой операционной системы
3	Сформированность представлений о нормах информационной этики и права, принципов обеспечения информационной безопасности, способов и средств обеспечения надежного функционирования средств ИКТ	Достижение данного предметного результата достигается при изучении материала вновь вводимых в методику обучения модулей « Правовая информатика », « Информационная этика » и « Информационная безопасность »	Еще раз можно отметить, что идеология свободного программного обеспечения сама по себе более этична, чем идеология проприетарного ПО, так как она, по терминологии Н. А. Бердяева, более человечна и коммюнитарна. Что касается информационной безопасности, то ОС семейства Linux более устойчивы к вирусным угрозам ввиду крайне редкого выполнения операций под именем суперпользователя (администратора)
4	Владение основными сведениями о базах данных, их структуре, средствах создания и работы с ними	Достижение данного предметного результата достигается при использовании следующих свободных СУБД: MySQL, OpenOffice.org Base, PostgreSQL	Наличие большого количества свободных СУБД дает возможность, во-первых, выбрать наиболее подходящий вариант ПО для данных конкретных учебных задач, во-вторых, использовать выбранную СУБД и в школе, и дома. Эквивалентность используемого в школе и дома ПО обеспечивает преподавателю перспективу давать практические домашние задания учащимся
5	Владение опытом построения и использования математических моделей, интерпретации результатов, получаемых в ходе моделирования реальных процессов	Достижение данного предметного результата достигается внедрением интегрированных уроков «физика + информатика», «химия + информатика», «география + информатика», а также интегрированных методов обучения, описанных в разделе 3 настоящей статьи, организованных по формуле «метод обучения + компьютерное моделирование»	В данном случае именно применение свободного программного обеспечения, позволяющего и в школе, и дома использовать один и тот же программный инструмент (например, электронные таблицы OpenOffice.org Calc) в качестве среды для создания компьютерной модели, дает возможность широко использовать метод компьютерного моделирования физических, химических или биологических процессов с последующим исследованием созданных моделей

3. Методы обучения

В настоящее время, с одной стороны, реализация системно-деятельностного подхода как методологической основы ФГОС общего образования и, с другой стороны, специфика информатики как высокотехнологичного учебного предмета предполагают использование современных методов обучения, таких как:

- компьютерное моделирование (симуляция);
- метод информационного ресурса;
- разбор проблемных ситуаций в дискуссии;
- метод проектов;
- фронтальное обучение в диалоговом режиме;
- дидактические игры.

С точки зрения автора, миграция учебного процесса на свободное программное обеспечение открывает дополнительные возможности по интенсификации использования современных методов обучения в преподавании школьной информатики, более того, открывается перспектива применения интегрированных методов обучения, что было подробно рассмотрено в статье [21].

Важно отметить, что использование современных и особенно интегрированных методов обучения является

не самоцелью, а органично обусловлено самой логикой развития методической системы обучения информатике в условиях системно-деятельностного и компетентностного подходов к результатам обучения, так как направлено на стимуляцию познавательной активности учащихся путем включения их в практическую работу «собственными руками», нацеленную на конкретный «осязаемый» результат.

В качестве примера можно привести интегрированный метод «метод проектов + компьютерное моделирование». В случае интеграции метода моделирования с методом проектов мы получаем исследовательский проект, в ходе которого учащийся создает компьютерную модель какой-либо системы, процесса или явления (физического, биологического или социального) и затем всесторонне тщательно изучает созданную модель при разных значениях параметров, характеризующих состояние исследуемой системы.

В случае применения свободного программного обеспечения можно использовать данный интегрированный метод обучения в связи с тем, что у школьного преподавателя появляется возможность организовать учебный процесс таким образом, что учащиеся смогут работать

как в школе, так и дома в одних и тех же программах, но при необходимости под разными операционными системами.

Продолжим иллюстрацию данной технологии следующим примером: в школе в рамках интегрированных уроков «физика + информатика» изучается тема «Движение тела, брошенного горизонтально», причем в качестве среды моделирования движения тела используются свободные динамические электронные таблицы OpenOffice.org Calc. Будем считать, что в школе на компьютеры класса информатики установлена операционная система семейства Linux, в которой OpenOffice.org Calc установлен по умолчанию. Ребята работают над учебным проектом, закончить который за один урок не представляется возможным. В таком случае они имеют возможность сохранить неоконченную работу в формате OpenOffice.org Calc и затем продолжить ее усовершенствование на домашних компьютерах, на которых могут быть установлены операционные системы семейства Windows, Mac OS X или Linux. Так как данные электронные таблицы являются бесплатными и кроссплатформенными, то они могут быть без труда и дополнительных материальных затрат установлены под любой из перечисленных выше операционных систем.

Описанный выше процесс передачи компьютерной модели в виде файла в формате .ods между школьными и домашними компьютерами учащихся на промежуточных стадиях создания и исследования компьютерной модели схематично показан на рисунке 2.

Предложенная схема передачи файлов на промежуточной стадии создания и исследования компьютерной модели между компьютерами участников учебного процесса была бы в принципе не реализуема в том случае, если бы вместо свободных электронных таблиц OpenOffice.org Calc использовались закрытые проприетарные электронные таблицы Microsoft Excel, так как это потребовало бы как минимум приобретения лицензионной версии офисного пакета Microsoft Office 2016 для домашнего использования учащимися, розничная цена которого в настоящее время составляет

примерно 4000 рублей [16]. В результате из методики преподавания пришлось бы исключить использование интегрированного метода обучения «метод проектов + компьютерное моделирование», заключающегося в доработке, усовершенствовании и исследовании компьютерной модели, создание которой было начато учащимися в ходе урока в классе. Данное вынужденное урезание методики, в свою очередь, может привести к понижению уровня заинтересованности у мотивированных учащихся, стремящихся выполнить работу с высоким качеством.

Конечно, предлагаемая выше технология выполнения учебных проектов может быть осуществлена и с помощью общедоступных бесплатных облачных технологий, в частности, на базе Google-документов, которые являются онлайн-версией проприетарного офисного пакета Microsoft Office. Однако данный способ будет иметь следующие недостатки:

- использование Google-документов предполагает наличие постоянного подключения к сети Интернет, которое может пропасть в ходе урока или домашней работы учащегося, что сделает невозможным продолжение работы над проектом;
- облачное хранилище Google-документов находится на иностранных серверах, доступ к которым потенциально возможен со стороны третьих лиц;
- в настоящее время в сервисе Google-документы доступны только текстовые документы, электронные таблицы и презентации. В случае необходимости создания, например, растрового или векторного графического изображения воспользоваться данным сервисом не удастся в связи с отсутствием необходимых программных инструментов, таких как растровый графический редактор и векторный графический редактор.

Делая промежуточный вывод, можно отметить, что **компонент методической системы обучения информатике «методы обучения» является частично программно зависимым, особенно в плане использования современных и интегрированных методов обучения.**

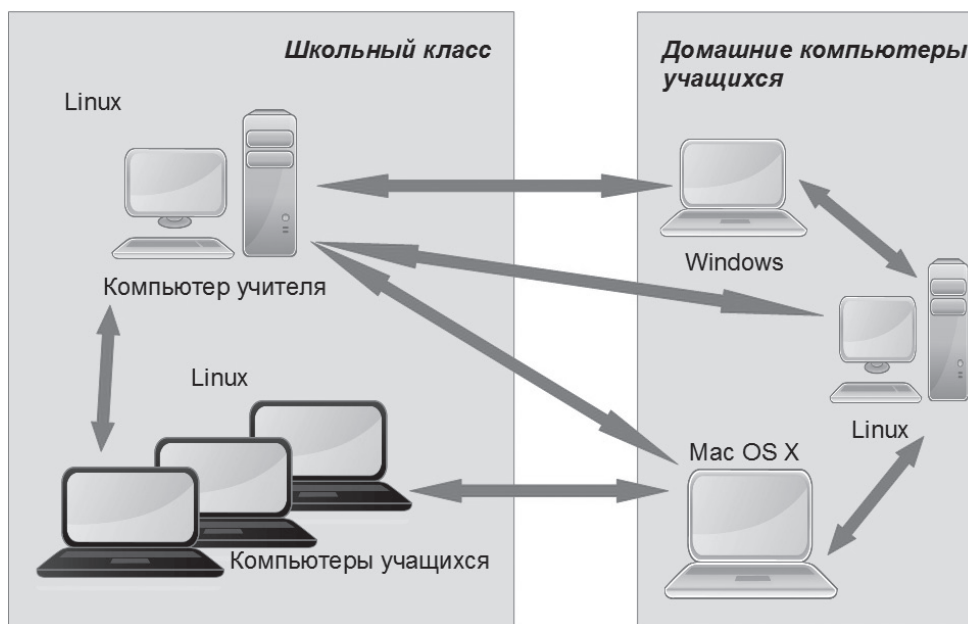


Рис. 2. Передача файла с компьютерной моделью на промежуточной стадии исследования между компьютерами участников учебного процесса

4. Формы обучения

Согласно обобщенной классификации А. А. Кузнецова, Т. Б. Захаровой, А. С. Захарова [11], формы обучения подразделяются на:

- урочные (лекции, семинары, практические занятия и др.);
- внеурочные:
 - регулярные (домашняя работа, курсы по выбору и др.);
 - эпизодические (тематические конференции, олимпиады, реферативные работы и др.);
- вспомогательные (индивидуальные консультации, групповые занятия выравнивания и др.).

На первый взгляд, можно предположить, что применяемый тип программного обеспечения никаким образом не повлияет на приведенные выше формы обучения информатике в школе, но тем не менее, с точки зрения автора, именно миграция учебного процесса с проприетарного на свободное программное обеспечение дает возможность несколько выйти за пределы часто критикуемой в наше время классно-урочной системы.

В частности, описанная выше в настоящей работе технология выполнения проекта по теме «Компьютерное моделирование и исследование движения тела, брошенного горизонтально» предполагает работу учащихся над одним учебным объектом (проектом), но в условиях постоянно меняющихся форм обучения, представленных в таблице 4.

Как видно из таблицы 4, учебный проект включает в себя практически все возможные формы обучения, причем в некоторых случаях формы обучения интегриру-

ются друг с другом, так, например, семинар объединяется с практической работой на компьютерах. А в каких-то случаях рассматриваемый этап проекта является индифферентным к возможной форме обучения, так, например, этапы 4–6 проекта могут проводиться как в классе, так и в виде домашней работы.

Говоря о формах обучения, также важно определить, каким образом будет осуществляться контроль знаний учащихся в случае миграции учебного процесса на свободное программное обеспечение. Классический вариант фронтальной контрольной работы в случае современного компетентностного подхода к результатам обучения несколько теряет свою былую актуальность. Поскольку мы ставим целью диагностировать уровень ИКТ-компетенций учащихся в той или иной области школьного курса информатики, то нужно применять адекватную цели форму контроля.

В качестве такой формы контроля можно предложить **полипрограммный интегрированный урок**, технология проведения которого подробно описана нами ранее в [14].

Полипрограммным интегрированным уроком будем считать самостоятельную практическую работу, проводимую для диагностирования уровня сформированности ИКТ-компетенций учащихся по нескольким темам курса информатики, целью которой является создание достаточно сложного информационного объекта с использованием нескольких различных программных сред (например, текстового редактора, электронных таблиц, растрового графического редактора и системы компьютерного черчения).

При этом алгоритм создания искомого информационного объекта предполагает попеременное неодно-

Таблица 4

Формы обучения при выполнении различных стадий учебного проекта

№ п/п	Стадия выполнения проекта	Форма обучения
1	Постановка целей и задач проекта	Лекция
2	Актуализация знаний по физике в области кинематики тела, брошенного горизонтально	Лекция
3	Совместное с учителем начало создания компьютерной модели движения тела, брошенного горизонтально, в электронных таблицах, заключающееся в расчете значений координат x и y тела в произвольный момент времени и построении траектории движения тела	Семинар + практическая работа
4	Самостоятельный расчет значений модулей проекций скорости тела на оси Ox и Oy в произвольный момент времени и построение графиков зависимости данных проекций от времени	Домашнее задание в виде практической работы / практическая работа в классе
5	Обмен полученными результатами с одноклассниками с целью взаимопроверки правильности выполнения задания	Домашнее задание в виде практической работы / практическая работа в классе
6	Самостоятельный расчет модуля скорости тела, брошенного горизонтально, и угла между вектором скорости тела и осью Ox в произвольный момент времени, построение соответствующих графиков зависимости	Домашнее задание в виде практической работы / практическая работа в классе
7	Написание индивидуальных рефератов по теме исследования	Реферативная работа (в классе или в виде домашней работы)
8	Проведение итоговой конференции по результатам выполненного исследования	Конференция

кратное использование всех перечисленных программных сред, состоит из большого количества операций (не менее десяти) и неочевиден для учащихся.

Таким образом, можно отметить, что **миграция учебного процесса на свободное программное обеспечение приводит не к увеличению многообразия потенциально возможных форм обучения, а к их интеграции друг с другом и к большей индифферентности вида учебной деятельности к форме обучения, определяемой эквивалентностью школьного и домашнего программного обеспечения. Также увеличивается динамика сменяемости форм обучения в ходе учебного процесса. Однако сам компонент методической системы обучения «формы обучения» можно считать мало программно зависимым.**

5. Средства обучения

Программное обеспечение принято относить к техническим программным средствам обучения. При миграции учебного процесса на свободное программное обеспечение происходит в первую очередь замена операционной системы Windows или Mac OS X на свободную ОС семейства Linux, а во вторую очередь — тотальная замена проприетарных прикладных программ на их свободные аналоги, представленные в таблице 5.

Таким образом, **содержание подкомпонента методической системы обучения информатике «технические программные средства обучения» полностью зависит от выбираемой идеологии при-**

меняемого в учебно-воспитательном процессе программного обеспечения.

Что касается непосредственно самих компьютеров, используемых в ходе проведения уроков информатики, их hardware, или «технических компьютерных аппаратных средств обучения», то в случае миграции учебного процесса на свободное программное обеспечение минимальные системные требования к ним корректируются в сторону уменьшения. Это связано с наличием версий операционной системы Linux, малотребовательных к аппаратному обеспечению, которые можно установить на персональные компьютеры с объемом оперативной памяти в 512 мегабайт. К таковым, в частности, относятся:

- Ubuntu;
- Linux Lite;
- Manjaro Linux — XFCE Edition;
- Linux Mint.

Более того, существуют дистрибутивы Linux, которые могут быть установлены на персональные компьютеры с объемом оперативной памяти в 256 мегабайт (!), например, ArchBang и Debian [9], но под ними будут работать не все необходимые для учебного процесса прикладные программы.

Резюмируя вышесказанное, можно отметить, что **содержание подкомпонента методической системы обучения информатике «технические компьютерные аппаратные средства обучения» является частично программно-зависимым в части возможности использования в учебном процессе менее мощных компьютеров.**

Таблица 5

Соответствие между платными и свободными программами

№ п/п	Тип программного обеспечения	Проприетарное ПО	Свободное ПО
1	Операционная система	Windows 7, 8, 8.1, 10; Mac OS X	Linux Mint, Ubuntu, OpenSUSE, Elementary OS, Debian, ROSA и др.
2	Текстовый процессор	Microsoft Word	OpenOffice.org Writer
3	Электронные таблицы	Microsoft Excel	OpenOffice.org Calc
4	СУБД	Microsoft Access	OpenOffice.org Base
5	Программа для создания презентаций	Microsoft PowerPoint	OpenOffice.org Impress
6	Векторный графический редактор	Corel DRAW	OpenOffice.org Draw, Inkscape
7	Растровый графический редактор	Adobe Photoshop	GIMP
8	Простой растровый графический редактор	Paint	EasyPaint
9	Интернет-браузер	Internet Explorer	Mozilla, Google Chrome
10	Среда визуального программирования	Visual Studio	Lazarus
11	Визуальная событийно-ориентированная среда программирования для обучения школьников младших классов	ЛогоМиры	Scratch
12	Звуковой редактор	Sound Editor	Audacity
13	Медиапроигрыватель	Проигрыватель Windows Media	VLC
14	Антивирус	Антивирус Касперского, Dr. Web	Clam AV

Выводы

В результате поэлементного анализа методической системы обучения информатике в школе можно констатировать наличие зависимости всех компонентов данной методической системы в той или иной степени от типа или, иными словами, от идеологии используемого в учебном процессе программного обеспечения — свободного или проприетарного. Скорректированную схему методики обучения можно представить, как показано на рисунке 3.

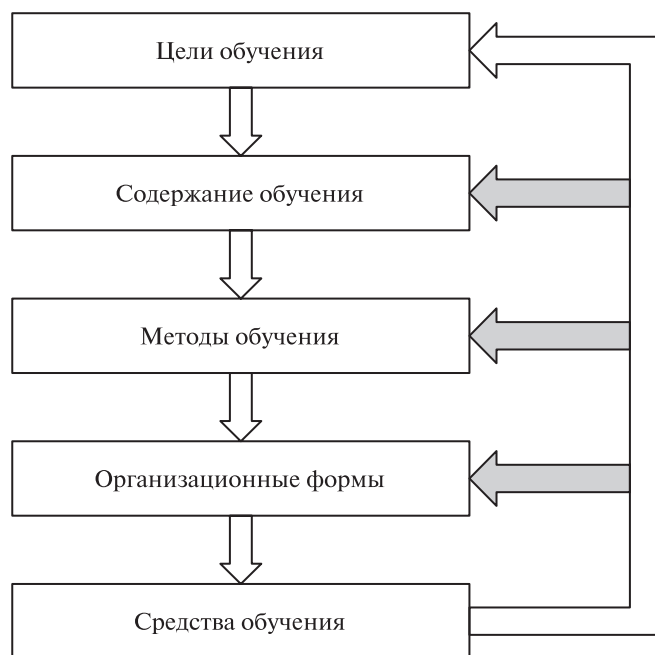


Рис. 3. Скорректированная схема методической системы обучения информатике

Принципиальное отличие идеологии свободного программного обеспечения от идеологии проприетарного ПО приводит, с нашей точки зрения, к необходимости разработки новой методики обучения информатике. О насущной необходимости создания новой методики преподавания информатики, опирающейся на идеологию свободного программного обеспечения можно говорить на основании анализа большинства статей, посвященных вопросу миграции учебного процесса общеобразовательной школы на СПО [2, 5–7, 10, 12, 13, 19]. Авторы данных статей, являющиеся практикующими преподавателями информатики в школе, колледже или вузе, как правило, пишут о трудностях перехода на свободное программное обеспечение, обусловленных именно отсутствием комплексных методик преподавания, а не какими-то другими обстоятельствами.

Таким образом, можно констатировать очевидную неизбежность в отношении создания методики преподавания информатики в школе, базирующейся только на СПО и имеющей следующие характерные черты, отличающие ее от существующих в настоящий момент методик, базирующихся на произвольном свободном/проприетарном/смешанном программном обеспечении, и обусловленные специфической идеологией СПО:

1. Вся методика обучения пронизана идеей воспитания личности школьников, связанной с положительным воздействием на духовно-нравственную сферу учащихся, определяемой свободным твор-

чеством сообществ программистов и пользователей СПО, обращенным к людям.

2. Причастность к комьюнотарному, по терминологии Н. А. Бердяева, сообществу пользователей и разработчиков свободного программного обеспечения может стать сильным дополнительным мотивационным аспектом для школьников, обучающихся по данной методике. Кроме того, учащиеся, обладающие к моменту окончания средней школы высоким уровнем ИКТ-компетенций в области свободного программного обеспечения, имеют хорошие перспективы на отечественном рынке труда.
3. Содержание обучения по данной методике дополняется модулями «Правовая информатика» и «Информационная этика», увеличивается и диверсифицируется объем модулей «Социальная информатика», «Информационная безопасность» и «Виды программного обеспечения».
4. Широко применяются современные и интегрированные методы обучения, в частности, такие как «метод проектов + компьютерное моделирование». При этом акцент делается на методе учебных проектов в связи с тем, что результат учебного проекта всегда «осязаем», он материализован в виде реферативной работы, компьютерной модели, распечатанной на 3D-принтере детали и т. п., что полностью соответствует компетентностному подходу в обучении. Метод проектов применяется повсеместно как в урочной, так и во внеурочной формах обучения благодаря использованию одного и того же программного обеспечения на школьных и домашних компьютерах учащихся.
5. В учебный процесс включается также и устаревшая компьютерная техника, на базе которой учащимися разворачиваются различные малотребовательные к hardware операционные системы семейства Linux. В результате этого учащиеся приобретают опыт установки операционной системы с носителей различных видов, разметки разделов жесткого диска и другие аналогичные навыки системного администрирования.

Список использованных источников

1. Бердяев Н. А. О свободе и рабстве человека. М.: Т8RUGRAM, 2018.
2. Бологова О. А. Современные информационные технологии высшей школы на базе свободного программного обеспечения // Психология и педагогика: методика и проблемы практического применения. 2010. № 12–2.
3. Бородин М. Н. Информатика. УМК для старшей школы: 10–11 классы. Углубленный уровень. Методическое пособие для учителя. Эл. изд. М.: БИНОМ. Лаборатория знаний, 2013. <http://files.lbz.ru/pdf/mpPolyakov10-11fgos.pdf>
4. Волох С. В. Ubuntu Linux с нуля. СПб.: БХВ-Петербург, 2018.
5. Герасимова О. А., Битюникова И. А. Свободное программное обеспечение в образовательном учреждении // Академический вестник. Вестник Санкт-Петербургской академии постдипломного педагогического образования. 2017. № 1 (35).
6. Глухих В. Р., Левкин Г. Г. Миграция на свободное программное обеспечение при обучении студентов // Экономические аспекты логистики и качества на транспорте. Межвузовский тематический сборник научных трудов. Министерство транспорта Российской Федерации, Федеральное агентство железнодорожного транспорта, Омский государственный

университет путей сообщения / под ред. Ю. А. Усманова. Омск, 2015.

7. *Зайцев А. Н.* Опыт внедрения свободного программного обеспечения в школе // Инновационные проекты и программы в образовании. 2008. № 2.

8. Идеология // Википедия. <https://ru.wikipedia.org/wiki/Идеология>

9. Как выбрать лучший легкий дистрибутив Linux // На неделе. <http://onweek.net/technologii/linux/luchshie-legkie-distributivy-linux-2018>

10. *Кочергина Л. А., Беленцов Д. С.* Свободное программное обеспечение в школе // Проблемы технологического образования в школе и вузе: Сборник статей по вопросам технологического образования / под ред. З. А. Литовой. Курск: Курский государственный университет, 2010.

11. *Кузнецов А. А., Захарова Т. Б., Захаров А. С.* Общая методика обучения информатике: учебное пособие для студентов педагогических вузов. Ч. I. М.: Прометей, 2016.

12. *Кулакова И. А.* Оценка уровня готовности учителей к использованию свободного программного обеспечения // Вестник Красноярского государственного педагогического университета им. В. П. Астафьева. 2015. № 1 (31).

13. *Мазурина Т. И.* Преподавание информатики на основе пакета свободного программного обеспечения (СПО) // Современная наука: тенденции развития. 2017. № 17.

14. *Маркушевич М. В.* Полипрограммные интегрированные уроки на базе свободного программного обеспечения как средство контроля ИКТ-компетенций старших школьников // Информатика в школе. 2017. № 4.

15. *Маркушевич М. В.* Проблема выбора между платным и свободным программным обеспечением для использования его в учебном процессе // Информатика и образование. 2016. № 10.

16. Офисное приложение Microsoft Office для дома и учебы 2016. https://www.citilink.ru/catalog/computers_and_notebooks/soft/soft_office/382626/

17. *Резвушкин С. В.* Особенности российского рынка образовательного программного обеспечения и предпосылки внедрения свободного программного обеспечения в школах // Ученые записки Российского государственного социального университета. 2010. № 9.

18. Сайт интернет-магазина allsoft.ru. <https://allsoft.ru/software/vendors/adobe/adobe-photoshop/>

19. *Сырцова А. О.* Использование свободного программного обеспечения в системе общего образования // Новые информационные технологии в автоматизированных системах. 2015. № 18.

20. *Угринович Н. Д., Самылкина Н. Н.* Информатика: 7–9 классы. Примерная рабочая программа. Эл. изд. М.: БИНОМ. Лаборатория знаний, 2016. <http://lbz.ru/metodist/iunk/informatics/files/ugrinovich-7-9-prog.pdf>

21. *Федосов А. Ю., Маркушевич М. В.* Использование интегрированных методов обучения предметам физико-математического цикла с применением свободного программного обеспечения // Информатика в школе. 2018. № 1.

22. *Федосов А. Ю., Маркушевич М. В.* Применение свободного программного обеспечения в учебном процессе как фактор духовно-нравственного воспитания учащихся // Информатика в школе. 2018. № 4.

НОВОСТИ

«Яндекс» запустил рейтинг популярных интернет-проектов

«Яндекс» объявил о том, что «Яндекс.Радар» стал показывать рейтинг самых популярных у россиян интернет-проектов. Это первый публичный рейтинг, который охватывает все наиболее посещаемые ресурсы и отражает интересы всей аудитории. Он будет полезен рекламодателям, маркетологам, медиа, владельцам сайтов — всем участникам рекламного рынка.

«Яндекс» рассчитывает их месячную аудиторию на основе обезличенных данных своих продуктов — Браузера, Элементов Яндекса и других. Каждый день одним только Браузером пользуются миллионы людей — в России его доля на компьютерах составляет 25 %. Для расчета аудитории применяется математическая модель, созданная с помощью машинного обучения. Она определяет диапазон посещаемости интернет-проектов.

Рядом с каждым из них в рейтинге указана минимальная оценка посещаемости, максимальную можно увидеть по наведению курсора. Для проектов, разрешивших использовать данные Метрики и AppMetrica, приведены точные значения, а также дневная посещаемость, доля аудитории приложений и среднее время, которое люди проводят на сайте.

Интернет-проекты можно фильтровать по типу и тематике — например, чтобы посмотреть список самых популярных у россиян магазинов или новостных сайтов.

Для каждой площадки Радар показывает две оценки аудитории — традиционную и полученную методом «кросс-девайс». Рейтинг автоматически формируется на основе второго метода: он гораздо точнее определяет реальный размер аудитории благодаря технологии Кримта, которая «узнает» пользователя в разных браузерах и на разных устройствах. При традиционном способе подсчета человек, открывавший сайт в двух разных браузерах, учитывается дважды.

«Любому рекламодателю важно знать, какую аудиторию он сможет привлечь и насколько эффективны будут его инвестиции. Около 80 % крупных рекламодателей и агентств планируют свой бюджет, опираясь на данные аналитических систем — обычно двух и более. С помощью нашего рейтинга они смогут получить полную картину, поскольку он учитывает посещаемость всех сайтов, — такой инструмент на рынке давно ждали», — отметил Леонид Савков, коммерческий директор «Яндекса».

«Яндекс.Радар» — это сервис аналитики, который позволяет следить за популярностью поисковых систем и браузеров в России, Беларуси, Казахстане и Турции, а также за посещаемостью сайтов в России. Сервис показывает ситуацию в целом, а также в различных срезах для конкретных платформ и типов устройств.

(По материалам CNews)

И. Е. Белоцерковская, М. Ю. Втюрин,
Нижегородский институт развития образования

ИЗУЧЕНИЕ ОСНОВ АЛГОРИТМИЗАЦИИ НА ПРИМЕРЕ МАШИНЫ ТЬЮРИНГА С ПРИМЕНЕНИЕМ ИГРОВОЙ ТЕХНОЛОГИИ

Аннотация

В статье представлены подробные решения нескольких алгоритмических задач на примере машины Тьюринга с подробным описанием формы работы на уроке. Особое внимание уделено описанию игрового способа обучения «Алгоритмизация по машине Тьюринга».

Ключевые слова: алгоритмизация, машина Тьюринга, групповая форма работы, урок информатики.

DOI: 10.32517/2221-1993-2018-17-9-25-27

Контактная информация

Белоцерковская Ирина Ефимовна, канд. физ.-мат. наук, доцент кафедры теории и методики обучения информатике Нижегородского института развития образования; адрес: 603122, г. Нижний Новгород, ул. Ванеева, д. 203; телефон: (831) 417-76-49; e-mail: miran_kaspir@mail.ru

Втюрин Максим Юрьевич, канд. физ.-мат. наук, зав. кафедрой теории и методики обучения информатике Нижегородского института развития образования; адрес: 603122, г. Нижний Новгород, ул. Ванеева, д. 203; телефон: (831) 417-76-49; e-mail: mvtyurin@yandex.ru

I. E. Belotserkovskaya, M. Yu. Vtyurin,
Nizhny Novgorod Institute of the Education
Development

STUDYING THE BASIS OF ALGORITHMIZATION BY THE EXAMPLE OF THE TURING MACHINE USING GAME TECHNOLOGY

Abstract

The article presents detailed solutions of a few algorithmic problems on the example of the Turing machine with a detailed description of the form of work at the lesson. Particular attention is paid to the description of the game learning method "Presenting algorithm as role game in Turing machine".

Keywords: algorithm, Turing machine, team learning activities, informatics lesson.

Федеральный государственный образовательный стандарт основного образования дает четкие указания учителю на темы и разделы учебного предмета, которые должны изучаться в школьном курсе. Предметные результаты освоения основной образовательной программы основного общего образования по предмету «Информатика», входящему в состав предметной области «Математика и информатика», включают в себя умение писать алгоритмы и решать задачи на составление алгоритмов [4]. Данные предметные результаты находятся на стыке предметов «Математика» и «Информатика», но анализ школьных УМК по информатике показывает, что теме «Основы алгоритмизации» уделяется значительное внимание именно в курсе информатики [3].

При решении любой алгоритмической задачи ученику приходится анализировать и выстраивать этапы решения задачи на компьютере. Для изучения процесса алгоритмизации прибегают к следующим методам:

- представляют алгоритм в виде блок-схемы;
- представляют алгоритм на формальном языке;
- выполняют пошаговое написание программы в режиме трассировки с применением трассировочных таблиц.

Описание алгоритма, поэтапное решение задачи могут вызывать затруднения у учащихся, поэтому применение интерактивных методов позволяет школьникам обучаться на собственном опыте путем специально организованного и регулируемого «проживания» определенной задачи (ситуации). В качестве такого интерактивного метода мы предлагаем использовать на уроке **ролевую игру «Алгоритмизация по машине Тьюринга»**.

Но, прежде чем переходить к описанию методики организации и проведения игры, необходимо познакомиться с машиной Тьюринга как уточненным понятием алгоритма. Машина Тьюринга с примерами хорошо описана в методическом пособии элективного курса «Математические основы информатики» [1].

Машина Тьюринга — это абстрактный исполнитель (абстрактная вычислительная машина), реализующий алгоритмический процесс, предназначенный для детализации представления алгоритма. Он был разработан в 1936 году Аланом Тьюрингом.

В состав машины Тьюринга входят:

- лента, разделенная на ячейки;
- каретка — считающая и записывающая головка;
- программируемый автомат, который представляет собой программу в виде таблицы.

Команды машины Тьюринга:

- запись символа в видимую ячейку;
- перемещение по ленте на один символ (R — вправо, L — влево, N — остаться на месте);
- переход в другое состояние.

Пример ленты машины Тьюринга представлен на рисунке 1. В каждой ячейке ленты может находиться только один символ. Будем считать, что Λ — пустая ячейка. Лента неограниченна, но в рамках рассмотрения нашей задачи мы будем использовать ограниченный участок бесконечной ленты, в случае необходимости записи новых символов можно дополнять ленту ячейками справа и слева.

Λ	a_{20}	a_{12}	a_3	a_5	Λ
-----------	----------	----------	-------	-------	-----------

Рис. 1. Бесконечная лента, разделенная на ячейки

На рисунке 2 представлен программируемый автомат машины Тьюринга, т. е. программа в виде таблицы.

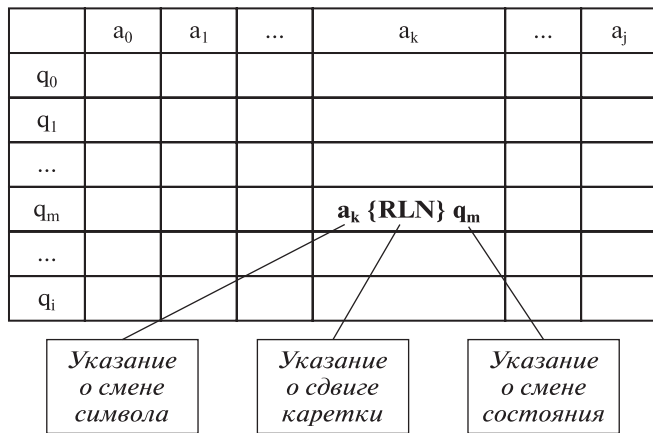


Рис. 2. Программируемый автомат машины Тьюринга

Разберем методику организации и проведения ролевой игры «Алгоритмизация по машине Тьюринга».

У нас будут задействованы следующие роли:

- оператор каретки;
- контролер состояний;
- оператор стирания;
- оператор записи (ввода).

Все остальные участники игры будут являться программой машины Тьюринга.

Обязанности каждой заявленной роли:

- оператор каретки должен перемещать каретку в необходимое место ленты, по определению машины Тьюринга он может двигаться вправо, влево или стоять на месте;
- контролеру состояний доступна для чтения только та ячейка, над которой находится каретка, и он должен оглашать аудитории текущий символ и состояние машины;
- оператор стирания в случае получения от программы команды «стереть» обязан удалить символ с ленты;
- оператор записи (ввода) в случае получения от программы команды «запись символа» обязан взять маркер и написать требуемое значение в той позиции, где находится каретка.

Прежде чем начинать игру, необходимо:

- на доске нарисовать ленту с исходными значениями (символами), а также все состояния машины;
- указать место на ленте, где стоит каретка;
- перечислить на доске состояния машины Тьюринга (для конкретного примера);
- обеспечить наличие у оператора стирания тряпки или губки;
- обеспечить маркером оператора записи;
- раздать аудитории подготовленные карточки для конкретного примера программы машины Тьюринга;
- объяснить контролеру состояний, что ему необходимо оглашать аудитории состояние машины и символ на ленте, над которым стоит каретка.

Дидактическое обеспечение игры «Алгоритмизация по машине Тьюринга».

В качестве раздаточного материала используются карточки.

Общие правила построения карточки.

На карте состояний располагается набор карточек (ячейки, выделенные жирным контуром), которые необходимо нарезать. На каждой карточке по вертикали

напечатаны состояние машины и символ на ленте, над которым стоит каретка; а по горизонтали — программа машины Тьюринга (что делает машина при данном символе и состоянии машины).

В начале игры:

- оператору каретки выдаются два магнита: один магнит он будет перемещать в необходимое место ленты, второй — в нужное состояние машины;
- контролер состояний должен оглашать аудитории текущий символ и состояние машины;
- оператору записи (ввода) предоставляется маркер;
- оператору стирания предоставляется тряпка или губка;
- группа учеников (остальные учащиеся) получает карточку с программой машины Тьюринга.

Контролер состояний зачитывает аудитории текущий символ и состояние машины. Ученик из аудитории, которому попала карточка с объявленным символом и состоянием машины (смотреть на вертикальный текст), оглашает, что необходимо сделать машине Тьюринга (читает текст по горизонтали).

Далее в работу включаются:

- оператор каретки — в случае, если необходимо переместить каретку вправо, влево, а также изменить состояние;
- оператор записи — если необходимо записать символ на ленте;
- оператор стирания — если необходимо стереть символ.

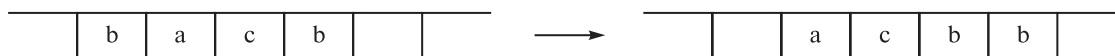
Такой подход позволяет в игре понять написание программы машины Тьюринга.

Рассмотрим формирование карточки на примере задачи 1 (рис. 3).

Условия задач 1 и 2 взяты из учебно-методического пособия [2].

$0, q_1$	0, R, q_1	$1, q_1$	1, R, q_1
$3, q_1$	3, R, q_1	$4, q_1$	4, R, q_1
$6, q_1$	6, R, q_1	$7, q_1$	7, R, q_1
$9, q_1$	9, R, q_1	Λ, q_2	Λ, L, q_2
$1, q_2$	2, N, !	$3, q_2$	3, N, !
$4, q_2$	5, N, !	$5, q_2$	6, N, !
$7, q_2$	8, N, !	$8, q_2$	9, N, !
$0, q_2$	1, N, !	$5, q_1$	5, R, q_1
$2, q_1$	2, R, q_1	$0, q_2$	1, N, !
$8, q_1$	8, R, q_1	$6, q_2$	7, N, !
$3, q_2$	4, N, !	$9, q_2$	0, L, q_2

Рис. 3. Карта состояний для задачи 1

Рис. 4. Пример переноса первого символа непустого слова P в его конец**Задача 1.**

Слово P — это последовательность из десятичных цифр, т. е. запись неотрицательного целого числа в десятичной системе. $A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ — внешний алфавит (из чего может состоять лента). Требуется получить на ленте запись числа, которое на 1 больше числа P .

Решение.

Алгоритм решения задачи:

1. Перемещаем автомат вправо под последний символ. Когда автомат оказывается на пустой ячейке, возвращаем его назад (перемещаем автомат влево) под последнюю цифру и переводим в состояние q_2 .
2. Прибавляем 1 к той цифре, которую видим в данный момент. Если цифра от 0 до 8, то прибавляем 1 и останавливаемся. Для цифры 9 заменяем на 0 и перемещаем автомат влево, оставаясь в том же состоянии, так как должны выполнить то же самое действие — увеличить на 1 видимую цифру. В случае отсутствия цифры в ячейке при осуществлении сдвига влево записываем в эту ячейку 1 и останавливаемся.

Раздаточный материал для задачи 1 представлен на рисунке 3.

Задача 2.

$A = \{a, b, c\}$. Перенести первый символ непустого слова P в его конец. Пример показан на рисунке 4.

Решение.

Алгоритм решения задачи:

1. В пустом слове P запомнить первый символ, а потом удалить его.
2. Переместить автомат вправо под первую пустую ячейку после слова и вписать в эту ячейку запомненный символ.

Раздаточный материал для задачи 2 представлен на рисунке 5.

Проводя урок по данной теме, учитель подбирает оптимальную, на его взгляд, форму работы с учащимися, и очень часто выбор делается в пользу организации групповой формы работы на уроке информатики. Применяя такую форму работы, учитель ориентирует учащихся на достижение цели посредством сотрудничества.

Многими учителями групповая деятельность на уроке позиционируется как средство естественного распределения ролей в группе, что дает возможность практически каждому учащемуся успешно проявить себя в ходе обсуждений и выработки совместных ре-

a, q_1	Λ, R, q_2	b, q_1	Λ, R, q_3
Λ, q_1	Λ, R, q_1	a, q_2	a, R, q_2
c, q_2	c, R, q_2	Λ, q_2	$a, N, !$
b, q_3	b, R, q_3	c, q_3	c, R, q_3
a, q_4	a, R, q_4	b, q_4	b, R, q_4
Λ, q_4	$c, N, !$	b, q_2	b, R, q_2
c, q_1	Λ, R, q_4	a, q_3	a, R, q_3
Λ, q_3	$b, N, !$	c, q_4	c, R, q_4

Рис. 5. Карта состояний для задачи 2

шений. Для «Алгоритмизации по машине Тьюринга» наиболее эффективной является работа в группах по 15–20 учеников. Закрепление ролей в группе (функций, обязанностей) в рамках поиска решения позволяет снижать уровень тревожности обучающихся, страха оказаться неуспешным, а также способствует развитию опыта ведения диалога, умения аргументировать свою точку зрения, что также регламентируется требованиями ФГОС ООО.

Рассмотренный в статье подход к организации групповой работы учащихся для изучения основ алгоритмизации был апробирован в рамках курсов повышения квалификации в Нижегородском институте развития образования и получил положительный отзыв учителей информатики.

Список использованных источников

1. Андреева Е. В., Босова Л. Л., Фалина И. Н. Математические основы информатики. Элективный курс: методическое пособие. М.: БИНОМ. Лаборатория знаний, 2007.
2. Пильщиков В. Н., Абрамов В. Г., Вылиток А. А., Горячая И. В. Машина Тьюринга и алгоритмы Маркова. Решение задач: учебно-методическое пособие. 2-е изд., испр. и доп. М.: МГУ, 2016.
3. Семакин И. Г. Информатика. Программа для старшей школы: 10–11 классы. Углубленный уровень. М.: БИНОМ. Лаборатория знаний, 2015.
4. 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%20утверждении%201897.pdf)

А. М. Абухба,

Абхазский государственный университет, г. Сухум, Абхазия

НЕКОТОРЫЕ ОСОБЕННОСТИ ПРОГРАММИРОВАНИЯ В MATHCAD PRIME

Аннотация

В статье рассмотрены некоторые отличительные особенности программирования в системе компьютерной математики Mathcad Prime. Приведены примеры различных программ, способствующих пониманию сути этих особенностей.

Ключевые слова: Mathcad, программирование, подпрограмма, переменная, операторы, функция, вектор, матрица, массив, ветвление, циклы.

DOI: 10.32517/2221-1993-2018-17-9-28-33

Контактная информация

Абухба Андрей Мустафаевич, доцент кафедры прикладной математики и информатики Абхазского государственного университета, г. Сухум, Абхазия; *адрес:* 384900, Республика Абхазия, г. Сухум, ул. Университетская, д. 1; *телефон:* +7 (840) 223-10-19; *e-mail:* aabuchba@mail.ru

A. M. Abukhba,

Abkhaz State University, Sukhum, Abkhazia

SOME FEATURES OF PROGRAMMING IN MATHCAD PRIME

Abstract

Some distinctive features of programming in the system of computer mathematics Mathcad Prime are considered in the article. Examples of various programs that contribute to understanding the essence of these features are given.

Keywords: Mathcad, programming, subroutine, variables, operators, functions, vector, matrix, array, branching, cycles.

Система Mathcad — одна из самых мощных и эффективных систем компьютерной математики, которая занимает особое место среди подобных других систем (Matlab, Maple, Mathematica и др.).

Mathcad — это приложение для математических и инженерных вычислений, промышленный стандарт проведения, распространения и хранения расчетов [5]. Задумывался Mathcad как средство «программирования без программирования», но, если возникает такая потребность, Mathcad имеет довольно простые для усвоения инструменты программирования, позволяющие строить весьма сложные алгоритмы, к чему прибегают, когда встроенных средств решения задачи не хватает, а также когда необходимо выполнять серийные расчеты [6]. Начиная с версии Mathcad Prime 3.0 появилась возможность внешнего программирования: теперь можно создавать программы на C++ или FORTRAN за пределами Mathcad (например, в Microsoft Visual Studio). Внутри Mathcad эти программы будут работать как пользовательские функции.

Основное отличие Mathcad от аналогичных программ — это графический, а не текстовый режим ввода выражений. В любом случае формулы будут иметь привычный, аналогичный книжному, вид. Любые переменные, формулы, параметры можно изменять, наблюдая воочию соответствующие изменения результата [6]. Также имеется возможность использования различных единиц измерения при проведении расчетов. Mathcad предоставляет большой набор единиц измерения, соответствующих международной системе единиц измерения СИ, американскому аналогу USCS и традиционной системе типа СГС (сантиметр — грамм — секунда).

Совместное использование функций и операторов Mathcad позволяет составлять довольно сложные программы. Если учесть, что в Mathcad содержится более 700 встроенных функций, то становится очевидным, насколько упрощается процесс составления программ для решения сложных задач. При составлении программ можно организовать вызов встроенных констант и функций, а также обращение к определенным ранее переменным и функциям пользователя. В этом смысле программирование на встроенном в систему языке программирования можно назвать **укрупненным программированием**.

На простых примерах, не отвлекаясь на решение сложных задач, рассмотрим некоторые особенности использования встроенного языка программирования.

Для составления программ (подпрограмм-функций) в Mathcad Prime имеется специальная вкладка *Программирование* (рис. 1):

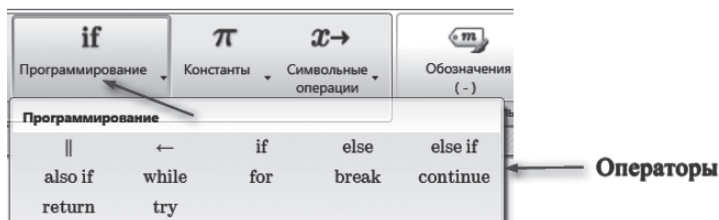


Рис. 1

Алгоритмические конструкции (операторы) в среде Mathcad вводятся не традиционным набором через клавиатуру служебных слов if, then, else, while и т. д., а нажатием одной из соответствующих кнопок панели инструментов или с использованием сочетания клавиш, предусмотренных для этих целей.

Необходимо отметить, что использование в программах обычного оператора присваивания ($:=$) вызовет ошибку. Вместо него необходимо использовать оператор локального присваивания $\leftarrow$. Также в программах

Программа	Результаты работы программы
$Kvur(a, b, c) := \left\ \begin{array}{l} D \leftarrow b^2 - 4 \cdot a \cdot c \\ x_1 \leftarrow \frac{-b + \sqrt{D}}{2 \cdot a} \\ x_2 \leftarrow \frac{-b - \sqrt{D}}{2 \cdot a} \\ \text{if } D < 0 \\ \quad \ Res \leftarrow \text{“Комплексные корни”} \\ \text{else} \\ \quad \ Res \leftarrow \text{“Действительные корни”} \\ \left[\begin{array}{l} Res \\ x_1 \\ x_2 \end{array} \right] \end{array} \right\ $	$Kvur(1, 2, 3) = \left[\begin{array}{l} \text{“Комплексные корни”} \\ \left[\begin{array}{l} -1 + 1.414i \\ -1 - 1.414i \end{array} \right] \end{array} \right]$ $Kvur(1, 2, -3) = \left[\begin{array}{l} \text{“Действительные корни”} \\ \left[\begin{array}{l} 1 \\ -3 \end{array} \right] \end{array} \right]$

Рис. 2

в определениях условия (логическое выражение) нельзя использовать обычный знак равенства $=$, вместо него необходимо использовать оператор сравнения $\equiv$ (жирный знак равенства).

Создание программ осуществляется с помощью оператора «программа — $\|$ ». Этот оператор является многоступенчатым контейнером для управляющих операторов Mathcad.

Общий вид программы:

$$Name(Param) := \left\| \begin{array}{l} S \\ Res \end{array} \right\|$$

Здесь:

Name — имя программы;

Param — формальные параметры (могут отсутствовать);

S — оператор(ы);

Res — результат.

Mathcad-программа возвращает результат на *последнем шаге*. Причем в качестве результата может возвращаться *единственный «объект»*. «Объект» может представлять собой:

- любое число, в том числе комплексное;
- текстовое сообщение, заключенное в кавычки (“ ”);
- вектор или матрицу (допускается вложенность).

Аналогами одномерного массива являются вектор-строка и вектор-столбец, а двумерного массива — матрица.

Пример 1.*

Найти корни квадратного уравнения:

$$ax^2 + bx + c = 0, a \neq 0.$$

Решение представлено на рисунке 2. В программе на этом рисунке:

Kvur — имя программы;

a, b, c — формальные параметры;

(1, 2, 3) и (1, 2, -3) — фактические параметры, подставляемые вместо формальных.

Вывод полученных результатов записан в виде вектора-столбца, элементами которого являются текстовое сообщение и вектор-столбец, содержащий корни уравнения x_1 и x_2 (вложенность массивов).

Для организации разветвлений в программах на Mathcad Prime используются операторы if, else, else if и also if. Первым оператором должен быть if, который является «заголовком» начала разветвляющегося процесса. Организовать ветвление можно и без использования вкладки *Программирование*. Для этого используется управляющий оператор if, общий вид которого: if (*B*, *S1*, *S2*), что соответствует конструкции: if *B* then *S1* else *S2* (если *B*, то *S1*, иначе *S2*), где *B* — логическое выражение (условие), *S1* и *S2* — произвольные числа, массивы или строки либо функции, при вычислении которых получается результат одного из этих типов. Управляющий оператор if необходимо набирать с помощью клавиатуры.

Пример 2.

Найти значение функции:

$$y = \begin{cases} \sqrt{x}, & \text{при } x \geq 3; \\ x^2, & \text{при } x < 3. \end{cases}$$

Решение.

$$\begin{aligned} y(x) &:= \text{if}(x \geq 3, \sqrt{x}, x^2) \\ y(9) &= 3 \\ y(-5) &= 25 \end{aligned}$$

Рис. 3

Пример 3.

Рекурсивное вычисление факториала $n!$:

$$\begin{aligned} Fakt(n) &:= \text{if}(n = 0, 1, n \cdot Fakt(n - 1)) \\ Fakt(0) &= 1 \\ Fakt(10) &= 3628800 \end{aligned}$$

Рис. 4

* Этот и все рассмотренные в дальнейшем примеры решены в системе Mathcad Prime 4.0.

Программа	Результат работы программы
<pre> ORIGIN := 1 F(n) := $\left\ \begin{array}{l} k \leftarrow 1 \\ \text{for } i \in 1..n \\ \quad \left\ \begin{array}{l} \text{for } j \in 1..n \\ \quad \left\ \begin{array}{l} M1_{i,j} \leftarrow \text{trunc}(\text{rnd}(10)) \\ M2_{i,j} \leftarrow \text{trunc}(\text{rnd}(20)) \\ M3_{i,j} \leftarrow \text{trunc}(\text{rnd}(15)) \end{array} \right. \\ \quad \text{for } i \in [M1 \ M2 \ M3] \\ \quad \quad \left\ \begin{array}{l} MAX_k \leftarrow \max(i) \\ \Delta_k \leftarrow i \\ SrAr_k \leftarrow \text{mean}(i) \\ k \leftarrow k+1 \end{array} \right. \\ \quad \left[\begin{array}{l} M1 \\ M2 \\ M3 \end{array} \right] \quad MAX \ \Delta \ SrAr \end{array} \right\$ </pre>	$F(5) = \left[\begin{array}{c} \left[\begin{array}{ccccc} 0 & 9 & 2 & 2 & 0 \\ 2 & 3 & 1 & 6 & 1 \\ 5 & 7 & 1 & 4 & 9 \\ 5 & 2 & 3 & 3 & 2 \\ 1 & 6 & 3 & 7 & 0 \end{array} \right] \\ \left[\begin{array}{ccccc} 12 & 5 & 7 & 18 & 3 \\ 0 & 0 & 17 & 18 & 11 \\ 5 & 1 & 12 & 7 & 7 \\ 12 & 17 & 18 & 14 & 13 \\ 4 & 1 & 2 & 13 & 0 \\ 1 & 14 & 3 & 13 & 11 \\ 1 & 2 & 2 & 13 & 14 \\ 5 & 8 & 9 & 8 & 2 \\ 0 & 10 & 5 & 11 & 5 \\ 9 & 0 & 1 & 9 & 6 \end{array} \right] \\ \left[\begin{array}{ccc} 9 & \begin{bmatrix} -2302 \\ 15511 \\ 39966 \end{bmatrix} & \begin{bmatrix} 3.36 \\ 8.68 \\ 6.48 \end{bmatrix} \end{array} \right] \end{array} \right]$

Рис. 5

Для организации циклов в программах Mathcad Prime используются операторы цикла с предусловием **while** и с параметром **for**. При этом если количество повторений тела цикла известно или можно вычислить, то используют цикл с параметром, в противном случае — цикл с предусловием.

Цикл с параметром **for** в среде Mathcad намного более гибок, чем его аналоги в языках программирования Pascal, C и т. д. [1].

Общий вид оператора **for**:

$$\left\| \begin{array}{l} \text{for } i \in D \\ S \end{array} \right\|$$

Здесь:

i — любое допустимое имя переменной или параметр цикла;

D — значение или последовательность значений. Чаще всего D — переменная-диапазон, но можно также использовать вектор или список из скаляров или векторов, задающих последовательность скаляров, которые по очереди становятся значениями переменной i . Чтобы выполнить один и тот же расчет над несколькими матрицами, можно определить D как последовательность матриц;

S — тело цикла, любое допустимое выражение или последовательность выражений.

Пример 4.

Создать три квадратные матрицы случайных целых чисел. Для каждой из них найти:

- максимальный элемент;
- определитель;
- среднее арифметическое.

Решение представлено на рисунке 5. В программе на этом рисунке: параметр i в цикле **for** $i \in [M1, M2, M3]$ будет поочередно принимать значения элементов сформированных матриц.

Вывод полученных результатов записан в виде вектора-строки, элементами которого являются вектор-столбец, содержащий исходные матрицы, и векторы-столбцы, содержащие искомые значения для каждого из матриц (вложенность).

Для случая, когда D — переменная-диапазон, в Mathcad Prime предусмотрены две разновидности данного оператора: обычный и модифицированный.

Обычный оператор:

$$\left\| \begin{array}{l} \text{for } i \in N..K \\ S \end{array} \right\|$$

В обычном операторе:

i — параметр цикла;

N, K — начальное и конечное значения параметра цикла (границы диапазона);

S — оператор(ы), или тело цикла.

Параметр цикла i после каждого выполнения S меняется с шагом, равным 1, причем если начальное значение параметра меньше конечного, то увеличивается на 1, в противном случае уменьшается на 1.

Модифицированный оператор:

$$\left\| \begin{array}{l} \text{for } i \in N, N+h..K \\ S \end{array} \right\|$$

В модифицированном операторе h — шаг изменения параметра i : параметр цикла i после каждого выполнения S меняется на величину h .

Необходимо отметить, что параметр цикла может принимать не только целое, но и дробное (десятичные и обыкновенные дроби) значение.

Пример 5.

Вычислить: $S = 1 + 2 + \dots + n$.

Решение.

$$\begin{array}{l}
 \text{Sum}(n) := \left\| \begin{array}{l} S \leftarrow 0 \\ \text{for } i \in 1..n \\ \left\| \begin{array}{l} S \leftarrow S+i \\ S \end{array} \right\| \end{array} \right\| \\
 \text{Sum}(40) = 820
 \end{array}$$

Рис. 6

Пример 6.Вычислить: $S = n + (n - 1) + \dots + 1$.

Решение.

$$\begin{array}{l}
 \text{Sum}(n) := \left\| \begin{array}{l} S \leftarrow 0 \\ \text{for } i \in n..1 \\ \left\| \begin{array}{l} S \leftarrow S+i \\ S \end{array} \right\| \end{array} \right\| \\
 \text{Sum}(40) = 820
 \end{array}$$

Рис. 7

Пример 7.Вычислить: $S = \frac{1}{10} + \frac{2}{10} + \dots + \frac{n}{10}$.

Решение.

$$\begin{array}{l}
 \text{Sum}(n) := \left\| \begin{array}{l} S \leftarrow 0 \\ \text{for } i \in \frac{1}{10}, \frac{2}{10} \dots \frac{n}{10} \\ \left\| \begin{array}{l} S \leftarrow S+i \\ S \end{array} \right\| \end{array} \right\| \\
 \text{Sum}(50) = 127.5
 \end{array}$$

Рис. 8

В примере 5 происходит автоматическое увеличение параметра цикла i на 1, в примере 6 — уменьшение на 1, в примере 7 — увеличение на $\frac{1}{10}$.

Встроенный в Mathcad язык программирования можно использовать и как язык для первоначального ознакомления с азами программирования, так как в нем реализованы все базовые алгоритмические конструкции: следование, ветвление и циклы. Также имеются развитые средства для обработки одномерных и двумерных массивов, предусмотрены возможности создания подпрограмм-функций для их использования (вызова) в других программах и реализации рекурсивных алгоритмов.

В случае использования языка для начального ознакомления или закрепления ранее полученных знаний при изучении других языков программирования желательно составлять развернутые программы, даже тогда, когда имеются встроенные функции и операторы, по-

Программа	Замена	Программа	Замена
$P = x!$, где $x \in N$ $\text{Fakt}(x) := \left\ \begin{array}{l} \text{if } x = 0 \\ \left\ \begin{array}{l} P \leftarrow 1 \end{array} \right\ \\ \text{else} \\ \left\ \begin{array}{l} P \leftarrow 1 \\ \text{for } i \in 1..x \\ \left\ \begin{array}{l} P \leftarrow P \cdot i \end{array} \right\ \end{array} \right\ \\ P \end{array} \right\ $	$x!$	$P = x^y$, где $y \in N$ $\text{Stepen}(x, y) := \left\ \begin{array}{l} \text{if } y = 0 \\ \left\ \begin{array}{l} P \leftarrow 1 \end{array} \right\ \\ \text{else} \\ \left\ \begin{array}{l} P \leftarrow 1 \\ \text{for } i \in 1..y \\ \left\ \begin{array}{l} P \leftarrow P \cdot x \end{array} \right\ \end{array} \right\ \\ P \end{array} \right\ $	x^y
$\text{НОД}(a, b)$ $\text{NOD}(a, b) := \left\ \begin{array}{l} \text{while } a \neq b \\ \left\ \begin{array}{l} \text{if } a > b \\ \left\ \begin{array}{l} a \leftarrow a - b \end{array} \right\ \\ \text{else} \\ \left\ \begin{array}{l} b \leftarrow b - a \end{array} \right\ \end{array} \right\ \\ a \end{array} \right\ $	$\text{gcd}(a, b)$	$\text{НОК}(a, b)$ $\text{NOK}(a, b) := \left\ \begin{array}{l} a1 \leftarrow a \\ b1 \leftarrow b \\ \text{while } a1 \neq b1 \\ \left\ \begin{array}{l} \text{if } a1 > b1 \\ \left\ \begin{array}{l} a1 \leftarrow a1 - b1 \end{array} \right\ \\ \text{else} \\ \left\ \begin{array}{l} b1 \leftarrow b1 - a1 \end{array} \right\ \end{array} \right\ \\ \text{RES} \leftarrow \frac{a \cdot b}{a1} \\ \text{RES} \end{array} \right\ $	$\text{lcm}(a, b)$

Рис. 9

Программа	Замена	Программа	Замена
Наибольшее из двух $Max(a, b) := \begin{cases} \text{if } a > b \\ \quad \parallel Max \leftarrow a \\ \text{else} \\ \quad \parallel Max \leftarrow b \\ \parallel Max \end{cases}$	$\max(a, b)$	Наименьшее из двух $Min(a, b) := \begin{cases} \text{if } a < b \\ \quad \parallel Min \leftarrow a \\ \text{else} \\ \quad \parallel Min \leftarrow b \\ \parallel Min \end{cases}$	$\min(a, b)$
$C_n^k = \frac{n!}{k!(n-k)!}$ $P(m) := \begin{cases} \text{if } m = 0 \\ \quad \parallel P \leftarrow 1 \\ \text{else} \\ \quad \parallel P \leftarrow 1 \\ \quad \text{for } i \in 1..m \\ \quad \quad \parallel P \leftarrow P \cdot i \\ \parallel P \end{cases}$ $C(n, k) := \begin{cases} P1 \leftarrow P(n) \\ P2 \leftarrow P(k) \\ P3 \leftarrow P(n-k) \\ \parallel Res \leftarrow \frac{P1}{P2 \cdot P3} \\ \parallel Res \end{cases}$	$\text{combin}(n, k)$	$A_n^k = \frac{n!}{(n-k)!}$ $P(m) := \begin{cases} \text{if } m = 0 \\ \quad \parallel P \leftarrow 1 \\ \text{else} \\ \quad \parallel P \leftarrow 1 \\ \quad \text{for } i \in 1..m \\ \quad \quad \parallel P \leftarrow P \cdot i \\ \parallel P \end{cases}$ $A(n, k) := \begin{cases} P1 \leftarrow P(n) \\ P2 \leftarrow P(n-k) \\ \parallel Res \leftarrow \frac{P1}{P2} \\ \parallel Res \end{cases}$	$\text{permut}(n, k)$
Максимальный элемент одномерного массива ORIGIN := 1 $Max(n) := \begin{cases} \text{for } i \in 1..n \\ \quad \parallel A_i \leftarrow \text{trunc}(\text{rnd}(50)) \\ \parallel Max \leftarrow A_1 \\ \text{for } i \in 2..n \\ \quad \text{if } Max < A_i \\ \quad \quad \parallel Max \leftarrow A_i \\ \parallel [A \quad Max] \end{cases}$	$\max(A)$	Минимальный элемент одномерного массива ORIGIN := 1 $Min(n) := \begin{cases} \text{for } i \in 1..n \\ \quad \parallel A_i \leftarrow \text{trunc}(\text{rnd}(50)) \\ \parallel Min \leftarrow A_1 \\ \text{for } i \in 2..n \\ \quad \text{if } Min > A_i \\ \quad \quad \parallel Min \leftarrow A_i \\ \parallel [A \quad Min] \end{cases}$	$\min(A)$

Рис. 10

звolyающие без какого-либо программирования получать необходимые результаты.

Некоторые программы и заменяющие их операторы и функции представлены на рисунках 9, 10.

Практически каждая составленная (автономная) программа, может быть использована при составлении более сложных программ в качестве подпрограммы-функции.

Пример 8.

Вычислить сумму:

$$S = 1! + 2! + \dots + n!, \quad n \geq 1.$$

Решение представлено на рисунке 11.

Переменные, значения которых определены в программе, являются локальными, и их значения недоступны вне этой программы.

Подпрограмма	Основная программа	Результат работы
$P(k) := \begin{array}{ l} P \leftarrow 1 \\ \text{for } i \in 1..k \\ \quad P \leftarrow P \cdot i \\ P \end{array}$	$S(n) := \begin{array}{ l} S \leftarrow 0 \\ \text{for } i \in 1..n \\ \quad S \leftarrow S + P(i) \\ S \end{array}$	$S(5) = 153$

Рис. 11

Пример 9.

$$f := \begin{array}{|l} i \leftarrow 5 \\ i = ? \end{array}$$

Рис. 12

Переменные, значения которых определены вне программы, являются глобальными, и их значения доступны этой программе. При этом переменная должна получить свое значение до его использования в программе с помощью обычного оператора присваивания $\leftarrow$ или после записанной программы с помощью оператора глобального определения $\equiv$. Вышесказанное касается также определения функции пользователя.

Пример 10.

$i := 5$ $Prog := \begin{array}{ l} i \leftarrow i + 1 \\ Prog = 6 \end{array}$	$Prog := \begin{array}{ l} i \leftarrow i + 1 \\ Prog = 6 \\ i \equiv 5 \end{array}$
---	--

Рис. 13

Пример 11.

$f(x) := x^4$ $Prog := \begin{array}{ l} I \leftarrow \int_1^3 f(x) dx \\ Prog = 48.4 \end{array}$	$Prog := \begin{array}{ l} I \leftarrow \int_1^3 f(x) dx \\ Prog = 48.4 \\ f(x) \equiv x^4 \end{array}$
--	---

Рис. 14

* * *

В данной статье рассмотрены те особенности программирования в Mathcad Prime, с которыми изучающий его так или иначе будет сталкиваться на начальном этапе. Понимание сути этих особенностей существенно облегчит процесс начала составления различных программ в этой системе. Хорошими помощниками в этом являются также встроенные в систему русифицированные справочник и учебник.

Mathcad, помимо своего главного предназначения — выполнения разнообразных научных и инженерных расчетов, также прекрасно подходит для организации обучения основам программирования.

Mathcad является эффективным средством в сфере образования, поскольку он прост в использовании, широко применим во многих областях науки и техники.

Список использованных источников

1. Воскобойников Ю. Е., Задорожный А. Ф. Основы вычислений и программирования в пакете Mathcad PRIME: учебное пособие. СПб.: Лань, 2016.
2. Кирьянов Д. В. Mathcad 15/Mathcad Prime 1.0. СПб.: БХВ-Петербург, 2012.
3. Основы вычислений и программирования в пакете Mathcad: учебное пособие / под ред. Ю. Е. Воскобойникова. Новосибирск: НГАСУ (Сибстрин), 2012.
4. Уроки Mathcad // Archicad-autocad.com — Ваш графический софт. <http://archicad-autocad.com/uroki-mathcad.html>
5. Mathcad // ПТС — Официальный дистрибьютер Creo, Windchill, Mathcad, Vericut в России. <http://pts-russia.com/products/Mathcad/>
6. Mathcad // Википедия, Свободная энциклопедия. <https://ru.wikipedia.org/wiki/Mathcad>

НОВОСТИ

Amazon запустил сервис для создания и тестирования софта в области робототехники

Компания Amazon запустила облачный сервис RoboMaker, с помощью которого разработчики в сфере робототехники смогут создавать дополнительный софт, тестировать его и внедрять в свою продукцию. Об этом говорится на сайте компании.

С помощью инструментов RoboMaker разработчики смогут довольно просто создавать приложения для работы своей техники, а также тестировать ее внутри

виртуальной среды Amazon и пользоваться базовой инфраструктурой компании.

Операционная система роботов (ROS), с которой и работает RoboMaker, является наиболее известным софтом для робототехники с открытым кодом. По данным Amazon, RoboMaker позволит не только разрабатывать софт, но и быстро его тестировать благодаря облачным технологиям компании.

(По материалам «Хайтек»)



В. Ф. Очков, А. О. Иванова, М. Д. Алексеев,
Национальный исследовательский университет «МЭИ», г. Москва

ТРИ «ЖАДНЫХ» АЛГОРИТМА

Аннотация

В статье рассмотрены три логистические задачи (транспортная задача, задача коммивояжера, задача о погоне), на примере которых показаны суть и особенности «жадных» алгоритмов. Впервые дано решение транспортной задачи в среде Mathcad Prime матричным способом с использованием единиц измерения. Предложены два новых приложения задачи коммивояжера. Описана разностная схема решения задачи погони.

Ключевые слова: «жадные» алгоритмы, транспортная задача, задача коммивояжера, задача погони, Mathcad.

DOI: 10.32517/2221-1993-2018-17-9-34-42

Контактная информация

Очков Валерий Федорович, доктор тех. наук, профессор, профессор кафедры теоретических основ теплотехники Национального исследовательского университета «МЭИ», г. Москва; адрес: 111250, г. Москва, Красноказарменная ул., д. 14; телефон: (495) 362-71-71; e-mail: ochkov@twi.mpei.ac.ru

Иванова Александра Олеговна, студентка Национального исследовательского университета «МЭИ», г. Москва; адрес: 111250, г. Москва, Красноказарменная ул., д. 14; e-mail: IvanovaAO@mpei.ru

Алексеев Михаил Дмитриевич, студент Национального исследовательского университета «МЭИ», г. Москва; адрес: 111250, г. Москва, Красноказарменная ул., д. 14; e-mail: AlexeevMD@mpei.ru

V. F. Ochkov, A. O. Ivanova, M. D. Alexeev,
National Research University MPEI, Moscow

THREE "GREEDY" ALGORITHMS

Abstract

The article considers three logistic tasks (transportation problem, traveling salesman problem, pursuit problem), by the example of which the essence and features of "greedy" algorithms are shown. For the first time, a solution was given to a transportation problem in the Mathcad Prime environment using the matrix method using units of measure. Two new applications of the traveling salesman problem have been proposed. The difference scheme for solving the pursuit problem is described.

Keywords: greedy algorithms, transportation problem, traveling salesman problem, pursuit problem, Mathcad.

Жадность фраера сгубила!

Грубая народная поговорка

Решать задачи оптимизации можно разными способами, среди которых особо выделяются так называемые «жадные» алгоритмы, когда принимается локально оптимальное решение («тактика») в надежде, что конечное решение («стратегия») также будет оптимальным.

Примечание. В военном искусстве кроме тактики и стратегии еще есть и «оперативное искусство». Тактика и оперативное искусство часто требуют не продвижения войск вперед, а обороны или даже отступления. У Н. В. Гоголя в «Мертвых душах» это хорошо написано про Ноздрева:

Бейте его! — кричал он таким же голосом, как во время великого приступа кричит своему взводу: «Ребята, вперед!» — какой-нибудь отчаянный поручик, которого взбалмошная храбрость уже приобрела такую известность, что дается нарочный приказ держать его за руки во время горячих дел. Но поручик уже почувствовал бранный задор, все пошло кругом в голове его; перед ним носится Суворов, он лезет на великое дело. «Ребята, вперед!» — кричит он, порываясь, не помышляя, что вредит уже обдуманному плану общего приступа, что миллионы ружейных дул выставлялись в амбразуры неприступных, уходящих за облака крепостных стен, что взлетит, как пух, на воздух его бесильный взвод и что уже свищет роковая пуля, готовясь захлопнуть его крикливую глотку.

Давайте попытаемся раскрыть смысл «жадной» алгоритмической «тактики» на трех несложных, но интересных и поучительных примерах, связанных с логистикой — наукой об управлении материальными, информационными, людскими и прочими потоками с целью их оптимизации.

Пример 1. Транспортная задача

Скупой платит дважды!

Вежливый вариант вышеприведенной поговорки

Постановка задачи.

Имеются две шахты, где добывают уголь, и три теплоэлектроцентрали (ТЭЦ), где этот уголь сжигают «с колес», т. е. без складирования. Первая шахта выдает на-гора 70 т угля в сутки, а вторая — 80. Первая электростанция сжигает 40 т угля в сутки, вторая — 85, а третья — 25. Затраты на перевозку угля с первой шахты на первую электростанцию составляют 120 руб. за тонну, на вторую — 160, на третью — 150. По второй шахте цифры такие: с нее на первую электростанцию завести уголь стоит 80 руб. за тонну, на вторую — 100, а на третью — 170.

Нужно так запланировать перевозку угля, чтобы ее суммарная стоимость была минимальна.

На рисунке 1 показано решение этой задачи в среде Mathcad Prime: в расчет вводится матрица *Data* с вышеприведенными

$$\begin{aligned}
 \text{Data} &:= \begin{bmatrix} \text{“Шахта\ТЭЦ”} & 40 \frac{\text{т}}{\text{сут}} & 85 \frac{\text{т}}{\text{сут}} & 25 \frac{\text{т}}{\text{сут}} \\ 70 \frac{\text{т}}{\text{сут}} & 120 \frac{\text{руб}}{\text{т}} & 160 \frac{\text{руб}}{\text{т}} & 150 \frac{\text{руб}}{\text{т}} \\ 80 \frac{\text{т}}{\text{сут}} & 80 \frac{\text{руб}}{\text{т}} & 100 \frac{\text{руб}}{\text{т}} & 170 \frac{\text{руб}}{\text{т}} \end{bmatrix} \\
 \text{Шахта} &:= \text{submatrix}(\text{Data}, 2, \text{rows}(\text{Data}), 1, 1) \\
 \text{ТЭЦ} &:= \text{submatrix}(\text{Data}, 1, 1, 2, \text{cols}(\text{Data}))^T \\
 \text{Стоимость} &:= \text{submatrix}(\text{Data}, 2, \text{rows}(\text{Data}), 2, \text{cols}(\text{Data})) \\
 i &:= 1 \dots \text{last}(\text{Шахта}) \quad j := 1 \dots \text{last}(\text{ТЭЦ}) \\
 \sum_i \text{Шахта}_i &= 150 \frac{\text{т}}{\text{сут}} \quad \sum_j \text{ТЭЦ}_j = 150 \frac{\text{т}}{\text{сут}} \\
 \text{Цена}(\text{План}) &:= \sum_i \sum_j \text{План}_{i,j} \cdot \text{Стоимость}_{i,j} \\
 \text{План} &:= \begin{bmatrix} 0 & 70 & 0 \\ 40 & 15 & 25 \end{bmatrix} \frac{\text{т}}{\text{сут}} \quad \text{Цена}(\text{План}) = 20150 \frac{\text{руб}}{\text{сут}} \\
 m_i &:= 1 \quad p_j := 1 \quad \text{План} \cdot p = \begin{bmatrix} 70 \\ 80 \end{bmatrix} \frac{\text{т}}{\text{сут}} \quad \text{План}^T \cdot m = \begin{bmatrix} 40 \\ 85 \\ 25 \end{bmatrix} \frac{\text{т}}{\text{сут}}
 \end{aligned}$$

Рис. 1. Решение транспортной задачи по «жадному» алгоритму

цифрами. Этот массив данных по своей сути является матрицей, а по форме — таблицей с шапкой (перечисление ТЭЦ) и боковиком (перечисление шахт). Далее с помощью функции *submatrix* эта матрица расчленяется на вектор *Шахта* с двумя элементами, хранящими суточную производительность шахт, на вектор *ТЭЦ*, хранящий суточный расход угля на трех ТЭЦ, и на матрицу *Стоимость* с соответствующими значениями затрат на перевозку угля. Выделение вектора *ТЭЦ* ведется с транспонированием ($\blacksquare^T$) матрицы с одной строкой в вектор — матрицу с одним столбцом.

Примечание 1. Использование матриц и векторов вместо скалярных величин существенно расширяет возможности расчета. В задаче несложно увеличить число шахт и/или ТЭЦ и получать нужный ответ, не меняя блок решения задачи. Но при этом следует помнить, что чрезмерное увеличение размеров матрицы *Data* может привести к срыву решения из-за ограничений по точности применяемых в Mathcad численных методов минимизации. Кстати, изменение размера векторов и матриц в среде Mathcad Prime стало делать намного проще по сравнению с Mathcad 15 и ниже. В среде Mathcad Prime эта операция делается примерно так, как в среде табличного процессора Excel: кликаньем по специальным кнопкам.

Примечание 2. Решение нашей транспортной задачи ведется с использованием единиц массы (тонны), времени (сутки) и стоимости (рубли), что делает расчеты очень удобными и исключает некоторые типичные ошибки в них, связанные с пересчетами единиц. Это невозможно делать ни в электронных таблицах, ни в языках программирования, которые, увы, отучили нас работать с физическими и экономическими величинами [1, 2]. Пакет Mathcad исправляет это ненормальное положение вещей. В среде Mathcad Prime стало также возможным решать и дифференциальные уравнения с единицами измерения, что нельзя делать ни в Mathcad 15, ни в средах любых других программ.

Примечание 3. Матрица *Data* хранит величины разной размерности. Так стало возможным поступать только в среде Mathcad Prime. В старом пакете Mathcad (Mathcad 15 и ниже) этого делать было нельзя, что являлось самым существенным недостатком этого компьютерного инструмента решения задач (см. также конец примечания 2).

После ввода исходных данных и раскладывания их на два вектора и одну матрицу в расчет вводятся целочисленные переменные Range variables (диапазон) *i* и *j*, ведущие нумерацию шахт и электростанций от первой до последней (функция *last*). Далее двумя операторами суммы проверяется материальный баланс: вся суточная выработка шахт (150 тонн угля) сжигается на электростанциях. Такая автоматическая проверка не будет лишней, когда в задаче число шахт и ТЭЦ увеличится и ручную такой контроль станет вести довольно сложно.

Центральным элементом расчета является функция пользователя с именем *Цена* и с аргументом *План*, которая возвращает суммарную стоимость перевозки угля с шахт на ТЭЦ. Функция получилась очень компактной за счет использования двойной суммы элементов двух матриц. Это *целевая функция* нашего оптимизационного расчета! Ее аргумент — *параметры* оптимизации, представленные в матричной форме. Третий элемент оптимизации (*ограничения*) будет введен в расчет чуть позже — в блоке *Решить*.

Есть хорошее программистское правило: ввел в расчет функцию пользователя — тут же проверь ее работоспособность. Сейчас мы сделаем это, заодно решив задачу в полуавтоматическом режиме! На рисунке 1 матрица *План* вручную формируется пользователем с «жадными» замашками: по самому дешевому маршруту со второй шахты на первую ТЭЦ (80 рублей за тонну) перевозится максимально возможное количество угля — 40 тонн в сутки, то есть ровно столько, сколько

Решить

План $\geq 0 \frac{\text{т}}{\text{сут}}$ Шахта = План $\cdot$ р ТЭЦ = План^T $\cdot$ m

План := Minimize(Цена, План) = $\begin{bmatrix} 40 & 5 & 25 \\ 0 & 80 & 0 \end{bmatrix} \frac{\text{т}}{\text{сут}}$

Цена(План) = 17350 $\frac{\text{руб}}{\text{сут}}$

Рис. 2. Решение транспортной задачи по умному алгоритму

сжигает первая ТЭЦ. Следовательно, с первой шахты первая ТЭЦ уголь не получает, а 85 тонн для второй ТЭЦ поступает частично с первой шахты (70 тонн в сутки), а частично со второй шахты (15 тонн в сутки). Третья ТЭЦ получает уголь только со второй шахты (25 тонн в сутки). При таком раскладе, учитывающем топливный баланс, затраты на перевозку угля составят 20 150 рублей в сутки. Запомним эту цифру, которую вернула нам функция *Цена*!

Примечание 4. В конце рисунка 1 видно, что в расчет введены вспомогательные единичные векторы *m* (mines, шахты) и *p* (power plants, электростанции), которые позволяют «вытащить» из матрицы *План* векторы, хранящие производительности шахт и расходы угля на ТЭЦ. В матрицу *План* в ручном режиме можно вводить разные числа: главное тут, чтобы сохранялся баланс перевозок. Векторы *m* и *p* мы также будем использовать при записи ограничений в блоке *Решить*.

Теперь поручим работу поиска оптимального плана перевозки угля программе Mathcad, взяв за первое предположение (приближение к ответу) результаты ручной, «жадной», оптимизации (см. выше).

В среде Mathcad есть функция *Minimize*, которая в старых версиях пакета (Mathcad 15 и ниже) работает в паре с ключевым словом *Given*, а в новых версиях (Mathcad Prime) — в блоке *Решить* (*Solve*): см. рисунок 2. До вызова функции *Minimize* в специальной области записываются *ограничения*, которые наряду с целевой функцией являются неотъемлемым элементом нашей транспортной задачи. В этой области на рисунке 2 записаны одно неравенство (уголь, естественно, не возится с ТЭЦ обратно на шахты) и два равенства (два баланса по перевозкам угля: по шахтам и по электростанциям). Эти ограничения вытекают из условия задачи, и мы их уже учитывали, формируя матрицу *План*. В двух матричных равенствах-ограничениях задействованы уже упомянутые вспомогательные единичные векторы *m* и *p*. Это решение авторам статьи подсказал на сайте пользователей Mathcad (<https://community.ptc.com/t5/PTC-Mathcad/One-transport-problem-linear-programming-in-matrix-form-Why/m-p/485675>) Вальтер Эксингер (Walter Exinger) из Австрии. Однако это все важные, но детали. Главное тут то, что функция *Minimize* показала явную «жадность» нашего ручного расчета (см. рис. 1), основанного на «здравом смысле». Оказалось, что если по самому дешевому маршруту уголь совсем не возить, то суммарная стоимость его перевозки снизится с 20 150 до 17 350 рублей в сутки. Смотрим эпиграф к этой части статьи: поддавшись чувству жадности, мы бы заплатили за перевозки ну не в два раза больше, а на 14 % уж точно!

Примечание 5. Цифры для матрицы *План* на рисунке 1 были получены так. Там сначала были поставлены нули (*План* = 0 т/сут), а в расчете на рисунке 2 функция *Minimize* была временно заменена на функцию *Maximize*. Так был получен «жадный» ответ, который был перенесен в задачу на рисунке 1 и стал первым приближением к «нежадному» решению, показанному на рисунке 2.

Пример 2. Задача коммивояжера, или Игра в лото

Сообразив, куда прежде, куда после ехать, чтоб не возвращаться, Нехлюдов прежде всего направился в сенат.

Л. Н. Толстой «Воскресение»

Каждый из нас, отправляясь в путь по мало-мальски сложному маршруту, должен сообразить, «куда прежде, куда после ехать, чтоб не возвращаться». Многие из нас, если и не работали (подрабатывали) курьерами, то получали заказанный товар из рук этого человека. Курьеру же, «направляясь в путь», обязательно нужно сообразить, «куда прежде, куда после ехать», чтобы оптимизировать маршрут, т. е. попытаться решить *задачу коммивояжера* (странствующего торговца, корабейника): найти маршрут обхода *n* городов (адресатов в одном городе), побывав в каждом городе один раз, вернувшись в исходную точку и сведя к минимуму путь (время в пути, расходы на дорогу и т. д.).

Разработано множество алгоритмов различной сложности для решения этой задачи. Но сначала ее несколько упрощают: берут *n* точек на плоскости с координатами, хранящимися в векторах *X* и *Y*, и считают, что путь из *i*-й точки (*i* = 1..*n*) в *j*-ю точку (*j* = 1..*n*; *i* ≠ *j*) лежит по прямой и может быть пройден в двух направлениях: от *i* к *j* и от *j* к *i*.

Постановка задачи.

В квадрат 100 на 100 случайным образом «брошено» *n* точек, которые нужно соединить отрезками прямых — участками пути коммивояжера от точки к точке, которые нужно «посетить». Какая должна быть очередность «посещения» точек?

На рисунке 3 показана программа, написанная в среде Mathcad 15, реализующая *алгоритм ближайшего соседа*. Этот алгоритм, как и в предыдущей задаче, подсказывает «здравый смысл».

В программе, представленной на рисунке 3, сначала заполняется квадратная матрица *M*, хранящая расстояние от точки *i* до точки *j*. Если *i* = *j* (главная диагональ

```

n := 30      X := runif(n, 0, 100)  Y := runif(n, 0, 100)  ib := 3
Way :=      M ←      for i ∈ 1..n
               for j ∈ 1..n
                   Mi,j ← if [i = j, ∞, √(Xi - Xj)2 + (Yi - Yj)2]
               M
Way1 ← ib
for i ∈ 2..n
    for j ∈ 1..n
        sj ← M(Wayi-1, j)
    Wayi ← match(min(s), s)1
    for j ∈ 1..i - 1
        M(Wayj, Wayi) ← ∞
        M(Wayi, Wayj) ← ∞
return Way

```

Рис. 3. Программа решения задачи коммивояжера по «жадному» алгоритму

матрицы), то соответствующий элемент матрицы будет хранить значение бесконечности (в численной математике Mathcad она равна 10^{307}). Это будет знаком того, что по данному маршруту ходить не нужно. Значение бесконечности также будет записываться в элементы матрицы $M_{i,j}$ и $M_{j,i}$, если от точки i к точке j (или наоборот) путь был уже пройден. Но еще раньше нужно заполнить векторы X и Y (координаты точек на плоскости) и скалярную целочисленную переменную ib (номер первой точки маршрута).

Программа, реализующая метод ближайшего соседа (рис. 3), получилась очень компактной благодаря двум встроенным функциям Mathcad: $\min$ и match . Первая ищет минимальное значение в векторе или матрице, а вторая определяет координаты элемента вектора или матрицы с заданным значением. В программе в цикле с параметром i (перебор точек около очередной точки) формируется вектор S , который хранит расстояние от этой текущей i -й точки до всех остальных j -х точек. Далее с помощью функций $\min$ и match (см. выше) определяется точка (очередной элемент искомого вектора Way), ближайшая к нашей точке и в которой мы еще не были. После этого «сжигаются мосты»: элементам матрицы $M_{i,j}$ и $M_{j,i}$ присваиваются значения бесконечности, для того чтобы этот пройденный путь никогда больше не был минимальным и по нему больше не ходили.

В задачу коммивояжера, программа решения которой показана на рисунке 3, заложен, повторяем, алгоритм ближайшего соседа: из очередного города коммивояжер идет в ближайший город, который он еще не посещал. Алгоритм ближайшего соседа относится к разряду «жадных» алгоритмов. Из-за этой «жадности» нашему коммивояжеру приходится петлять, а в конце своего турне перескакивать в отдаленные города, ранее опрометчиво пропущенные. Все это увеличивает общую длину маршрута: неразумная «тактика» портит всю «стратегию» решения задачи — см. рисунок 4.

Есть и не вполне обычные приложения задачи коммивояжера. Так, например, расшифровать геном живого организма невозможно без решения довольно сложной задачи коммивояжера.

Мы же можем предложить еще одно «необычное» приложение задачи коммивояжера — ее решение в... игре в лото. В наш компьютерный век она может вестись без вытаскивания из мешка бочонков со случайными целыми числами от 1 до 90: играющие смотрят на экран дисплея, где показывается анимация пути коммивояжера через 90 населенных пунктов с показом номера этапа пути (очередность вытаскиваемых из мешка бочонков — переменная j) и номера очередного населенного пункта (числа, прописанного на бочонке, — переменная k) (рис. 4).

Играя в лото с таким виртуальным мешком с бочонками (с компьютерным генератором целых случайных неповторяющихся чисел от 1 до 90), можно заодно рассуждать о задаче коммивояжера, о других алгоритмах решения задачи (метод ветвей и границ, алгоритм отжига, муравьиный алгоритм, генетический алгоритм и др.). Оптимизировать можно и «жадный» алгоритм ближайшего соседа — выбирать город, из которого нужно начинать путь, развязывать петли маршрута и т. д.

В [3] описано еще одно необычное приложение задачи коммивояжера. На плоской светодиодной панели каждые сутки ровно в полночь световыми точками отмечаются 720 населенных пунктов нашей планеты или отдельной страны, выбранные случайным образом. Число 720 — это количество минут в полусутках — в 12 часах. Более крупными световыми точками выделяются 12 узловых населенных пункта («города»). Это будут оригинальные часы. Далее выбирается первый случайный город (12 ч 00 м — полдень), куда помещается «коммивояжер», которому дается задание обойти все населенные пункты. На цифровой панели таких часов помимо пути коммивояжера высвечиваются две цифры: номер города, из которого вышел наш коммивояжер (12, 1, ..., 11 — часы), и номер населенного пункта, который он только что прошел (0, 1, ..., 59 — минуты). Это будет текущим временем (рис. 5).

Подобные настенные часы будут уместны на вокзалах. Человек, приехавший на вокзал, не только может справиться о времени по таким часам, но и при желании будет следить за передвижением «коммивояжера»,

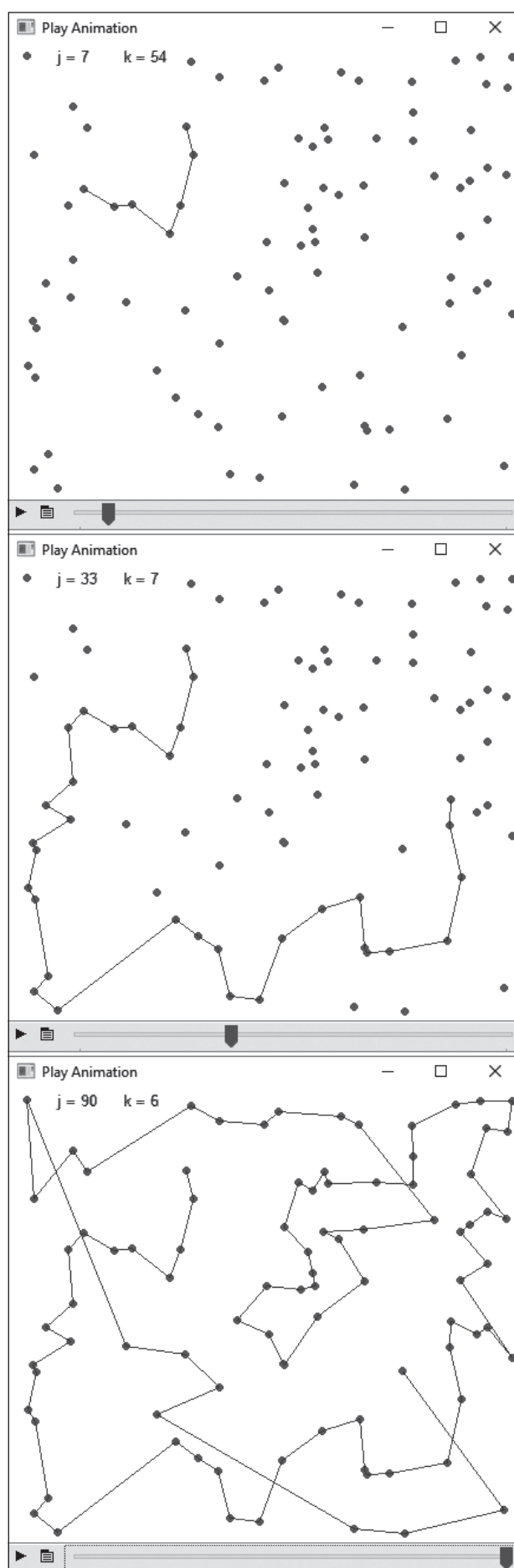


Рис. 4. Игра в лото в процессе решения задачи коммивояжера

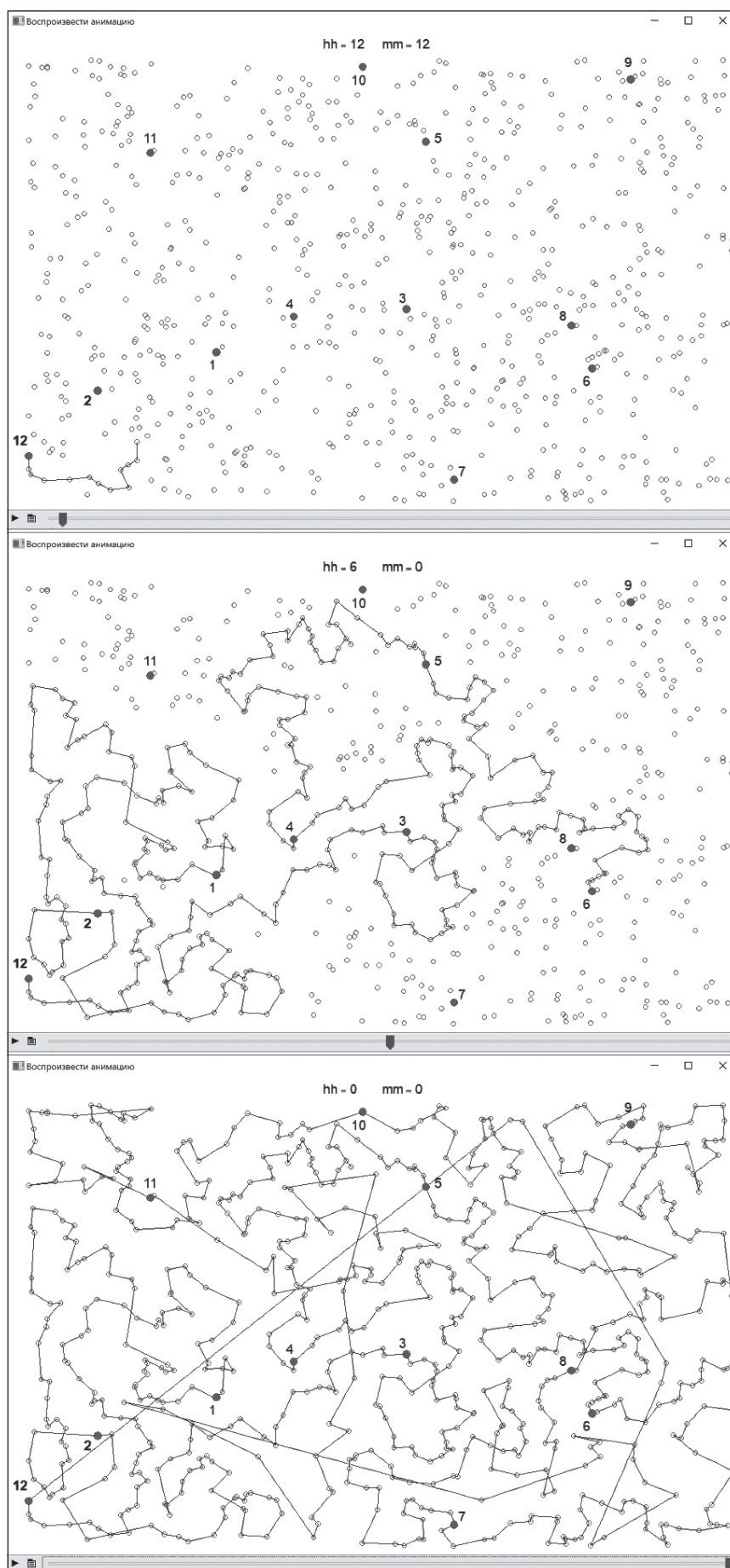


Рис. 5. Настенные часы с задачей коммивояжера
(анимация <https://community.ptc.com/t5/PTC-Mathcad/Math-cad-clock/td-p/583230>)

угадывать, куда он повернет в следующую минуту. Это скрасит ожидание поезда, заставит задуматься о задаче коммивояжера, об оптимизации собственного будущего путешествия...

Примечание. Циферблат часов на рисунке 5 — прямоугольный. Но его можно сделать с контуром, повторяющим контур конкретной страны. На таких часах часовые и минутные точки можно сделать не случайными (как на рисунке 5), а с соответствующими координатами городов и поселков данной страны.

Пример 3. Задача о погоне

«Погоня, погоня, погоня, погоня
в горячей крови!»

*Песня из к/ф «Новые приключения
неуловимых» (слова И. Гоффа)*

Выбор оптимального маршрута движения ведется в еще одной «логистической задаче» — задаче о погоне.

Постановка задачи.

Заяц бежит по полю, а за ним гонится волк. Какова будет траектория движения волка (кривая погони)? Как она будет связана с траекторией движения зайца?

Эта задача имеет важное практическое приложение: зенитная ракета или самолет-перехватчик летит к цели, выбирая некий оптимальный путь. Это приложение не просто важное — это вопрос жизни и смерти.

В интернете можно найти множество описаний постановки и решения этой задачи. При этом она обычно предельно упрощается: заяц бежит по прямой линии, а сбоку на него выбегает волк, ориентируясь не на свой интеллект или нюх, а на зрение — волк бежит строго

на зайца. Скорость волка, естественно, выше скорости зайца, и эти две величины постоянны во времени. Исследователи этой задачи, как правило, пытаются найти ее аналитическое решение — ищут математическое выражение для описания траектории бега волка, что выливается во множество сложных преобразований и пояснений к ним, в которых не посвященные в тонкости высшей математики быстро запутываются, теряют нить рассуждений и интерес к задаче.

Но в настоящее время мы все чаще и чаще используем не аналитические, а численные методы решения математических задач. Это, увы, несколько снижает изящество решения, но открывает другие интересные и не менее изящные возможности, в частности, для анимационного иллюстрирования решений, для их большей привязки к реальности.

Задачу о погоне волка за зайцем можно решить не только посредством составления «страшных» дифференциальных уравнений [4], но и другим путем — через реализацию несложной разностной схемы. А разность, как известно, это предтеча дифференциала — разности, стремящейся к нулю, но не достигающей нуля. Дифференциальные уравнения численно решаются через составление этих самых разностных схем. Так давайте же мы не будем составлять дифференциальное уравнение, описывающее бег волка за зайцем, которое аналитически решить невозможно, а вернемся к истокам — к этим самым разностным схемам. Обсуждение этой задачи можно посмотреть здесь: <https://community.ptc.com/t5/PTC-Mathcad/Is-it-my-own-or-Mathcad-15-error/m-p/576164>.

На рисунке 6 показана разностная схема решения задачи о волке и зайце. Центральным элементом этого численного метода является угол φ — угол направления, куда

$$\begin{pmatrix} x_{\text{wolf}_{i+1}} \\ y_{\text{wolf}_{i+1}} \\ \varphi_{i+1} \end{pmatrix} := \begin{pmatrix} x_{\text{wolf}_i} + v_{\text{wolf}} \cdot \Delta t \cdot \sin(\varphi_i) \\ y_{\text{wolf}_i} + v_{\text{wolf}} \cdot \Delta t \cdot \cos(\varphi_i) \\ \text{atan} \left(\frac{x_{\text{hare}_i} - x_{\text{wolf}_i}}{y_{\text{hare}_i} - y_{\text{wolf}_i}} \right) + \pi \cdot (y_{\text{wolf}_i} > y_{\text{hare}_i}) \end{pmatrix}$$

Рис. 6. Алгоритм «жадной» погони волка за зайцем

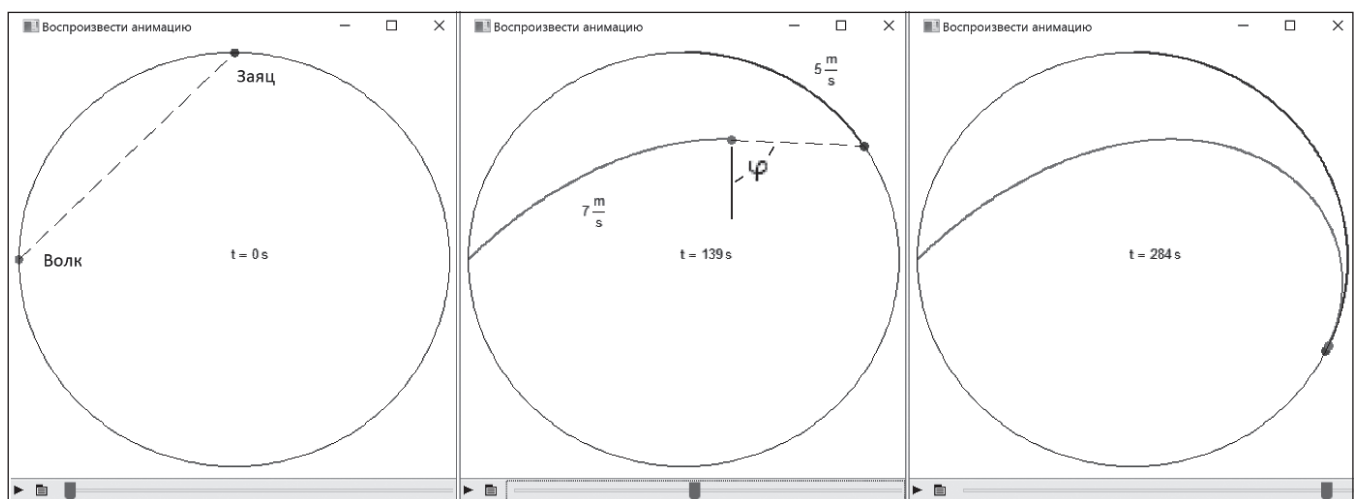


Рис. 7. Погоня волка за зайцем по «жадной» траектории

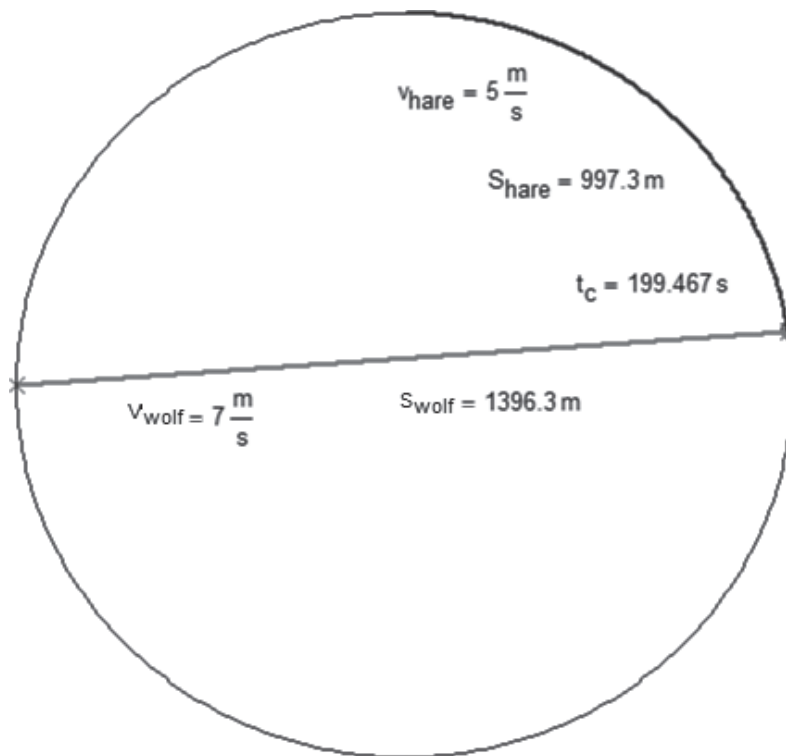


Рис. 8. Бег волка за зайцем по прямой

ориентируется волк, догоняя зайца (см. центральный кадр анимации на рисунке 7). По этому углу — через его синус и косинус — рассчитывается приращение пути волка в горизонтальном (точнее, левом) и вертикальном (правом) направлениях — значения векторов x_{wolf} и y_{wolf} . И все! И никаких головоломных дифференциальных уравнений и их решений — численных или аналитических!

На рисунке 7 можно видеть три кадра анимации такой погони: заяц бежит по кругу, стартуя из «двенадцати часов», если под кругом понимать циферблат часов. Из «девяяти часов» одновременно с зайцем выбегает волк, бежит строго на зайца и через 284 секунды ловит его по... «жадному» алгоритму без какой-либо оптимизации. А задача погони — это тоже задача оптимизации, где можно минимизировать длину пути погони, время погони или что-то другое.

Так вот, если минимизировать время бега волка за зайцем, то нужно выбирать прямолинейный маршрут (рис. 8), рассчитав заранее координаты точки, где произойдет роковая встреча. Такая погоня будет длиться не 284 секунды (см. рисунок 7 — третий кадр анимации), а 200 секунд.

Если же минимизировать не время, а длину пути погони, то волку нужно просто сидеть в засаде на «девяти часах» и ждать, когда заяц сам прибежит ему в пасть.

Но и бег по прямой (рис. 8) часто оказывается неоптимальным. Задача, решение которой показано на рисунке 9, доказывает это.

Кратчайшее расстояние между двумя точками, как известно, прямая линия. Но далеко не всегда можно и нужно перемещаться по прямой. Даже если нет явных препятствий в виде зданий, заборов и прочих преград, то перемещение по прямой даже в чистом поле не всегда оказывается оптимальным, если под оптимизацией понимать не сокращение расстояния, а минимизацию времени в пути [5]. Давайте не будем голословными и решим такую простенькую задачу:

Заяц сидит на открытой местности в точке *A* (рис. 9). Он чувствует волка. Зайцу нужно как можно быстрее добраться до точки *B*, где расположена его нора. По какой траектории бежать зайцу, если он сидит на газоне, перед норой асфальт, а путь пересекает вспаханная полоса? «Жадный» заяц побежит прямо к норе и...

Эту типичную задачу оптимизации, где целевой функцией будет время перемещения от точки *A* к точке *B*, а переменными оптимизации — абсциссы точек *C* и *D*, мы также решим опять же в среде пакета Mathcad. Ординаты точек *C* и *D* заданы и отмечают прямые линии раздела участков «газон — пашня» и «пашня — асфальт», которые параллельны оси абсцисс. Скорости бега по газону (*г*), вспаханному полю (*п*) и асфальту (*а*) тоже заданы, и они, естественно, разные: $v_a > v_p > v_g$. Заданы также координаты начальной и конечной точек *A* и *B*.

На рисунке 9 показано решение этой задачи с использованием уже известной нам встроенной в Mathcad функции *Minimize*. У нее переменное число аргументов. В нашей задаче их три: имя целевой функции *t* (time) и имена двух переменных оптимизации C_x и D_x . Функция *Minimize* по особому численному алгоритму будет автоматически менять значения переменных оптимизации C_x и D_x так, чтобы целевая функция приняла минимальное значение. Значения переменных C_x и D_x при проведении поиска изменяются от некоторых исходных (первое предположение — см. рис. 8), которые задаются самим пользователем, исходя из «физики» задачи. При значениях C_x и D_x , взятых для первого предположения, время бега от точки *A* к точке *B* равно 2 мин 50 с, а минимизация с помощью функции *Minimize* дает значение 2 мин 31 с.

Задача о беге зайца по газону, пашне и асфальту имеет прямую аналогию в... оптике. Принцип Ферма этого раздела физики гласит, что луч света, пробиваясь через прозрачные среды, выбирает такую траекторию, чтобы

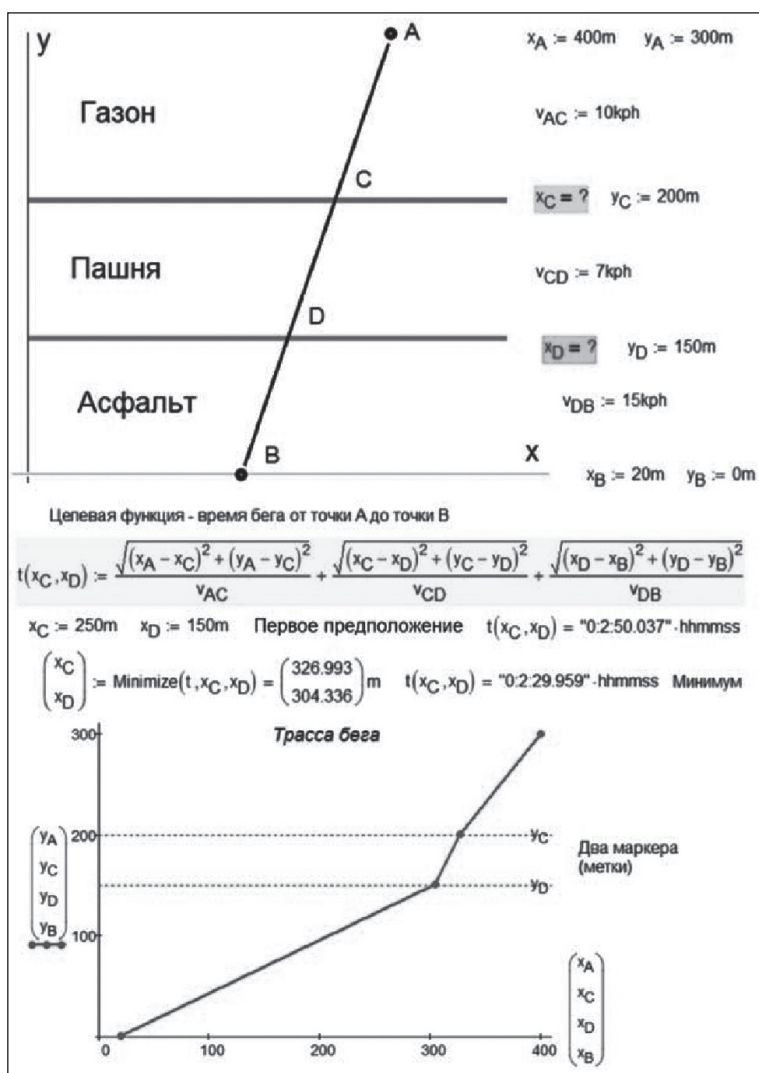


Рис. 9. Бег умного зайца по газону, пашне и асфальту

время движения было минимальным. Нижний график на рисунке 9 — это ломанная кривая прохода луча света через трехслойную прозрачную пластину с разными скоростями света в разных слоях.

Вышеприведенные примеры, конечно, слегка надуманные. Но вот реальный пример явной «жадности», с которой люди сталкиваются на эскалаторе. На этой движущейся лестнице обычно справа стоят, а слева проходят. Из-за такого способа перемещения часто возникают пробки при входе на эскалатор. Если же люди будут занимать правую и левую стороны эскалатора, то пробки на входе станут меньше и среднее время движения на эскалаторе снизится.

Вывод.

Решая сложную задачу со студентами или школьниками, можно начать с простейших «жадных» алгоритмов и программ, постепенно заменяя их на более сложные и более совершенные.

Список использованных источников

1. Очков В. Ф. Пойти туда, зная куда // Информатика в школе. 2014. № 10. С. 59–60. http://infojournal.ru/journals/school/school_10-2014/
2. Очков В. Ф. Физические и экономические величины в Mathcad и Maple. М.: Финансы и статистика, 2002. Серия «Диалог с компьютером». http://tw.t.mpei.ac.ru/ochkov/Units/Forward_book.htm
3. Очков В. Ф., Богомолова Е. П. Обратная задача коммивояжера, или Необычные математические часы // Открытое образование. 2014. № 2. 2014. С. 22–28. <http://tw.t.mpei.ac.ru/ochkov/SalesMan/index.html>
4. Очков В. Ф., Богомолова Е. П. Это страшное слово «диффуры»... // Информатика в школе. 2015. № 1. С. 55–58. http://infojournal.ru/journals/school/school_01-2015/
5. Очков В. Ф., Соколов А. В., Федорович С. Д., Мекес Л. Путешествие от дома в школу по маршруту Ферма, или Второе оптическое свойство гиперболы // Cloud of Science. 2016. Т. 3. № 4. С. 494–517. <http://tw.t.mpei.ac.ru/ochkov/Optic-Ochkov.pdf>

И. Ю. Воробьева,
АНОО ДО «Английский клуб», г. Челябинск,

О. А. Дмитриева,
Южно-Уральский государственный гуманитарно-педагогический университет, г. Челябинск

РЕШЕНИЕ ЗАГАДКИ ЭЙНШТЕЙНА НА УРОКАХ ИНФОРМАТИКИ

Аннотация

В статье предложена модификация текста загадки Эйнштейна для ее применения в школьном учебном процессе, раскрыты особенности решения данной задачи на уроках информатики в школе, рассмотрен таблично-графический метод решения загадки Эйнштейна.

Ключевые слова: загадка, Эйнштейн, загадка Эйнштейна, таблица, информатика.

DOI: 10.32517/2221-1993-2018-17-9-43-46

Контактная информация

Воробьева Ирина Юрьевна, преподаватель английского языка АНОО ДО «Английский клуб», г. Челябинск; *адрес:* 454003, г. Челябинск, ул. Героя России Е. Родионова, д. 13; *телефон:* (351) 776-01-08; *e-mail:* vorob.ir@gmail.com.

Дмитриева Ольга Александровна, канд. пед. наук, доцент кафедры информатики, информационных технологий и методики обучения информатике Южно-Уральского государственного гуманитарно-педагогического университета, г. Челябинск; *адрес:* 404580, г. Челябинск, ул. С. Кривой, д. 34; *телефон:* (351) 216-63-10; *e-mail:* dmitrievaoo@cspu.ru

I. Yu. Vorobeva,
The English Club, Chelyabinsk

O. A. Dmitrieva,
South Ural State Humanitarian Pedagogical University, Chelyabinsk

THE SOLUTION OF THE EINSTEIN'S RIDDLE ON INFORMATICS LESSONS

Abstract

The article proposes a modification of the text of the Einstein's riddle for its use in the school educational process, reveals the peculiarities of solving this riddle on informatics lessons at school, the tabular-graphical method for solving the Einstein's riddle is discussed.

Keywords: riddle, Einstein, Einstein's riddle, table, informatics.

Загадка Эйнштейна — одна из самых известных логических задач. Впервые задача была опубликована в журнале «Life International» в номере от 17 декабря 1962 года. Авторство данной загадки чаще всего приписывается Альберту Эйнштейну, хотя прямых подтверждений этому не существует.

Оригинальный вариант загадки состоит из 15 взаимосвязанных утверждений. Ниже приведен перевод оригинального условия задачи (оригинал можно посмотреть в [2]):

1. На улице стоят пять домов.
 2. Англичанин живет в красном доме.
 3. Испанца есть собака.
 4. В зеленом доме пьют кофе.
 5. Украинец пьет чай.
 6. Зеленый дом стоит сразу справа от белого дома.
 7. Тот, кто курит Old Gold, разводит улиток.
 8. В желтом доме курят Koal.
 9. В центральном доме пьют молоко.
 10. Норвежец живет в первом доме.
 11. Сосед того, кто курит Chesterfield, держит лису.
 12. В доме по соседству с тем, в котором держат лошадь, курят Koal.
 13. Тот, кто курит Lucky Strike, пьет апельсиновый сок.
 14. Японец курит Parliament.
 15. Норвежец живет рядом с синим домом.
- Кто пьет воду? Кто держит зебру?

В целях ясности следует добавить, что каждый из пяти домов окрашен в свой цвет, а их жители — разных национальностей, владеют разными животными, пьют разные напитки и курят разные марки американских сигарет. Еще одно замечание: в утверждении 6 «справа» означает «справа относительно вас» [1].

Изначально задача предполагает решение в уме без использования вспомогательных средств, в том числе без использования каких-либо записей.

Классический способ решения представляет собой решение табличным методом. Его суть заключается в заполнении таблицы данными друг за другом, исключая варианты, которые невозможны [1].

Загадка Эйнштейна является задачей повышенной сложности, которая требует достаточно много времени для ее решения. Основная сложность заключается в том, что эту задачу невозможно решить, просто оперируя исходными данными, — для решения необходимо сделать несколько предположений и, только исходя из этого, прийти к конечному решению. Все это представляет сложность при изучении загадки Эйнштейна в рамках школьных уроков. Кроме того, оригинальный текст задачи содержит названия марок сигарет, что является непедagogическим с точки зрения воспитания.

Для успешного изучения загадки Эйнштейна в рамках школьной программы задача требует изменений.

Во-первых, предлагается взять упрощенный вариант задачи, решение которого возможно без использования предположений. В интернете можно найти множество вариантов этой задачи, наиболее популярным из которых является следующий:

На одной улице подряд стоят пять домов, каждый своего цвета. В каждом живет человек, все пять — разных национальностей. Каждый человек предпочитает уникальную марку сигарет, напиток и домашнее животное. Кроме того:

1. Норвежец живет в первом доме.
2. Англичанин живет в красном доме.

3. Зеленый дом находится слева от белого, рядом с ним.
 4. Датчанин пьет чай.
 5. Тот, кто курит Marlboro, живет рядом с тем, кто выращивает кошек.
 6. Тот, кто живет в желтом доме, курит Dunhill.
 7. Немец курит Rothmans.
 8. Тот, кто живет в центре, пьет молоко.
 9. Сосед того, кто курит Marlboro, пьет воду.
 10. Тот, кто курит Pall Mall, выращивает птиц.
 11. Швед выращивает собак.
 12. Норвежец живет рядом с синим домом.
 13. Тот, кто выращивает лошадей, живет в синем доме.
 14. Тот, кто курит Winfield, пьет пиво.
 15. В зеленом доме пьют кофе.
- Вопрос: кто разводит рыбок? [1]

Во-вторых, в педагогических целях предлагается заменить марки сигарет на марки машин и заменить пиво на более подходящий напиток. Кроме того, для использования в качестве сокращенных названий начальных букв нужно изменить некоторые названия, чтобы начальные буквы всех данных одного типа были разными.

В-третьих, предлагается использовать иной метод решения задачи в целях экономии времени на уроке.

Таким образом, после всех изменений **предлагается использовать следующее условие задачи:**

На одной улице подряд стоят пять домов, каждый своего цвета. В каждом живет человек, все пять — разных национальностей. Каждый человек предпочитает уникальную марку автомобиля, свой напиток и свое домашнее животное. Кроме того:

1. Норвежец живет в первом доме.
 2. Англичанин живет в красном доме.
 3. Зеленый дом находится слева от белого.
 4. Датчанин пьет чай.
 5. Тот, кто водит Mercedes, живет рядом с тем, кто держит кошек.
 6. Тот, кто живет в желтом доме, водит BMW.
 7. У японца Ford.
 8. Тот, кто живет в центре, пьет молоко.
 9. Сосед того, у кого Mercedes, пьет воду.
 10. Тот, кто водит Audi, разводит птиц.
 11. Швед держит собак.
 12. Норвежец живет рядом с синим домом.
 13. Тот, кто держит лошадь, живет в синем доме.
 14. Тот, у кого Honda, пьет сок.
 15. В зеленом доме пьют кофе.
- Вопрос: кто разводит рыбок?

Для решения задачи воспользуемся следующим способом, назовем его условно «**таблично-графический**».

По условию задачи имеется:

- пять цветов, в которые окрашены дома;
- пять национальностей;
- пять видов напитков;
- пять видов животных;
- пять марок машин.

Для представления всех данных, возможных в одном доме, воспользуемся таблицей (сеткой) размером 5 × 5 клеток. Для заполнения данных воспользуемся начальными буквами всех названий:

цвет дома:

- К — красный;
- З — зеленый;
- Б — белый;
- С — синий;
- Ж — желтый;

национальность жителя:

- Н — немец;
- Д — датчанин;
- А — англичанин;
- Ш — швед;
- Я — японец;

марка машины:

- В — BMW;
- F — Ford;
- М — Mercedes;
- А — Audi;
- Н — Honda;

напиток:

- Ч — чай;
- С — сок;
- К — кофе;
- В — вода;
- М — молоко;

животное:

- С — собаки;
- Р — рыбки;
- П — птицы;
- К — кошки;
- Л — лошади.

Так как домов тоже пять, то и таких табличек подготовим пять. В результате таблица для решения задачи будет выглядеть, как представлено на рисунке 1.

Шаг 1.

После построения таблицы будем постепенно вычеркивать те данные, которые не удовлетворяют условию задачи. Соответственно, если условие истинно для одной из позиций, то данные следует вычеркнуть везде, кроме этой позиции. После первого прочтения условий можно сделать следующие выводы:

- из условия 1: норвежец живет в первом доме;
- из условия 2: первый дом не красный;
- из условия 3: первый дом не белый, а последний не зеленый;

	1					2					3					4					5				
цвет	К	З	Б	С	Ж	К	З	Б	С	Ж	К	З	Б	С	Ж	К	З	Б	С	Ж	К	З	Б	С	Ж
национальность	Н	Д	А	Ш	Я	Н	Д	А	Ш	Я	Н	Д	А	Ш	Я	Н	Д	А	Ш	Я	Н	Д	А	Ш	Я
машина	В	Ф	М	А	Н	В	Ф	М	А	Н	В	Ф	М	А	Н	В	Ф	М	А	Н	В	Ф	М	А	Н
напиток	Ч	С	К	В	М	Ч	С	К	В	М	Ч	С	К	В	М	Ч	С	К	В	М	Ч	С	К	В	М
животное	С	Р	П	К	Л	С	Р	П	К	Л	С	Р	П	К	Л	С	Р	П	К	Л	С	Р	П	К	Л

Рис. 1. Исходная таблица

	1	2	3	4	5
цвет	З		К	З	Б
национальность	Н	Д	Д	Д	Д
машина	В	В	В	В	В
напиток	С	Ч		Ч	Ч
животное	Р		С	С	С

Рис. 2. Вид таблицы после шага 1

- из условия 4: в первом доме не пьют чай;
- из условия 7: в первом доме не водят Ford;
- из условия 8: в третьем доме пьют молоко;
- из условия 11: в первом доме не держат собак;
- из условия 12: второй дом — синий;
- из условия 13: во втором доме держат лошадь;
- из условия 14: в третьем доме не водят автомобиль марки Honda;
- из условия 15: во втором доме не пьют кофе, а третий дом не зеленый.

На основе этих выводов вносим изменения в таблицу (рис. 2).

Шаг 2.

Снова обращаемся последовательно к условиям задачи и, учитывая изменения, выполненные в таблице, делаем следующие выводы:

- из условия 2: во втором доме живет не англичанин;
- из условия 3: первый дом не зеленый, третий дом не белый, значит, первый дом — желтый, следовательно, третий дом — красный, четвертый дом — зеленый, а пятый дом — белый;
- из условия 4: в третьем доме живет не датчанин;
- из условия 6: в первом доме водят автомобиль марки BMW;
- из условия 10: в первом доме не разводят птиц;
- из условия 11: во втором доме живет не швед;
- из условия 14: в первом доме не пьют сок;
- из условия 15: кофе пьют в четвертом доме, значит, в первом доме пьют воду.

Вид таблицы после внесения изменений представлен на рисунке 3.

	1	2	3	4	5
цвет	Ж	С	К	З	Б
национальность	Н	Д	Д	Д	Д
машина	В	В	В	В	В
напиток	С	Ч		Ч	Ч
животное	Р		С	С	С

Рис. 3. Вид таблицы после шага 2

	1	2	3	4	5
цвет	Ж	С	К	З	Б
национальность	Н	Д	Д	Д	Д
машина	В	В	В	В	В
напиток	С	Ч		Ч	Ч
животное	Р		С	С	С

Рис. 4. Вид таблицы после шага 3

Шаг 3.

Опять обращаемся последовательно к условиям задачи и, учитывая изменения, выполненные в таблице, делаем следующие выводы:

- из условия 2: в третьем доме живет англичанин;
- из условия 7: в третьем доме не водят автомобиль марки Ford;
- из условия 9: во втором доме водят Mercedes, значит, в третьем доме может быть только Audi;
- из условия 10: в третьем доме разводят птиц;
- из условия 14: во втором доме не Honda, значит, там не пьют сок, т. е. там пьют чай, следовательно, в пятом доме пьют сок и там же водят автомобиль марки Honda, а в четвертом доме водят Ford.

Вид таблицы после внесения изменений представлен на рисунке 4.

Шаг 4.

Снова обращаемся последовательно к условиям задачи и, учитывая изменения, выполненные в таблице, делаем следующие выводы:

- из условия 4: во втором доме живет датчанин;
- из условия 5: кошек могут держать только в первом доме;
- из условия 7: в четвертом доме живет японец, значит, швед живет в пятом доме;
- из условия 11: в пятом доме держат собак, значит, в четвертом доме разводят рыбок.

Вид таблицы после внесения изменений представлен на рисунке 5.

Задача решена, все соответствия установлены, ответом на главный вопрос задачи является то, что рыбок разводит японец.

	1	2	3	4	5
цвет					Ж
национальность	Н	Д	А	Я	Ш
машина	В	М	А	Ф	Н
напиток		Ч	М	К	
животное			П	Р	С

Рис. 5. Вид таблицы после шага 4

Данный способ решения задачи является более быстрым, чем обычный табличный метод.

Еще одним преимуществом данного метода является то, что учащимся нет необходимости самим заполнять таблицы и тратить на это время. Заполненную таблицу с исходными данными необходимо распечатать и выдавать ученикам в качестве карточек, на которых учащиеся будут вычеркивать ячейки, противоречащие условию.

Для организации успешной индивидуальной работы учащихся, а также для возможности продемонстрировать ход решения задачи можно подготовить несколько вариантов упрощенной загадки Эйнштейна. Для этого

достаточно изменить условия в задаче. Это можно сделать несколькими способами. Один из способов — переставить между собой данные одного типа в условии задачи, другой — заменить одни данные задачи другими, третий — поменять расположение элементов в исходной таблице.

Список использованных источников

1. Загадка Эйнштейна // Википедия, Свободная энциклопедия. http://ru.wikipedia.org/Загадка_Эйнштейна
2. Zebra_Puzzle // Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Zebra_Puzzle

НОВОСТИ

ИИ научился создавать репродукции картин на 3D-принтере

Инженеры из Массачусетского технологического института (MIT) создали систему RePaint — комбинацию искусственного интеллекта и трехмерной печати, которая позволит любому желающему создавать репродукции любимых картин. Об этом пишет «EurekAlert!».

Прежде чем начнется печать картины, ИИ раскладывает ее на слои. Затем 3D-принтер воссоздает изображение из нескольких слоев прозрачных чернил и под руководством нейросети раскрашивает изображение, учитывая освещенность помещения и другие факторы, чтобы наиболее точно передать цвет картины.

Пока RePaint может напечатать изображение размером с открытку и не способна воспроизвести фактуру работ, написанных маслом или обладающих матовым блеском. Однако в будущем люди смогут самостоятельно создавать копии своих любимых работ без существенных затрат, отмечается в сообщении института.

Ранее на аукционе Christie's купили картину «Портрет Эдмона Белами» из серии «Семья Белами», созданную искусственным интеллектом. Сумма сделки составила \$432 тыс., тогда как первоначально лот оценивали в \$7 тыс.

(По материалам «Хайтек»)

Соревнования по 3D-технологиям для школьников прошли в ТемоЦентре

Двухдневное практико-ориентированное соревнование для школьников в разрезе современных 3D-технологий прошло в рамках Московского технологического марафона.

Более 120 столичных школьников приняли участие в двухдневном 3D-мейкертоне, который проходил в Московском центре технологической модернизации образования с 27 по 28 ноября 2018 года. Юные москвичи познакомились с различными технологиями, научились собирать и настраивать 3D-принтеры и 3D-сканеры, а также создали собственные 3D-модели.

(По материалам, предоставленным пресс-службой Департамента образования города Москвы)

Перед школьными командами стояла задача в течение двух дней собрать три разных 3D-принтера и запустить на каждом из них тестовую печать. По завершению мейкертон все участники получили сертификаты 3D-мастеров.

Ребята, которые в прошлом году прошли все этапы Московского технологического марафона и набрали максимальное количество баллов, были приглашены на стажировку в одну из ведущих компаний по 3D-технологиям и на соревнованиях уже выполняли роль экспертов, помогая новичкам в сборке оборудования.



М. С. Долинский,

Гомельский государственный университет имени Франциска Скорины, Беларусь

ВВЕДЕНИЕ В РЕШЕНИЕ ЗАДАЧ НА ТЕМУ «ИССЛЕДОВАНИЕ»

Аннотация

В статье описана методика изучения темы «Исследование» при подготовке школьников к олимпиадам по информатике. Технической основой является разработанная под управлением автора инструментальная система дистанционного обучения (<http://dl.gsu.by>).

Ключевые слова: исследование, олимпиады по информатике, инструментальная система дистанционного обучения.

DOI: 10.32517/2221-1993-2018-17-9-47-51

Контактная информация

Долинский Михаил Семенович, канд. тех. наук, доцент, доцент кафедры математических проблем управления Гомельского государственного университета имени Франциска Скорины, Беларусь; *адрес:* 246000, Республика Беларусь, г. Гомель, ул. Советская, д. 104; *телефон:* (375-232) 77-70-69; *e-mail:* dolinsky@gsu.by

M. S. Dolinsky,
Gomel State University named after Francis Skorina, Belarus

INTRODUCTION TO RESEARCH PROBLEM SOLVING

Abstract

The article describes the methodology to teach research problem solving on Olympiads in informatics. Distance learning system DL.GSU.BY is the effective technical base for teaching.

Keywords: research problems, teaching for programming, Olympiads in informatics, distance learning tools.

Введение

С сентября 1996 года на базе средней школы № 27 г. Гомеля, Беларусь, а с сентября 1999 года дополнительно и на базе сайта дистанционного обучения DL.GSU.BY ведется работа по факультативному изучению информатики и программирования школьниками разных возрастов [1, 2]. Ключевыми особенностями этого обучения являются:

- раннее начало обучения (фактически с первого класса);
- система региональных олимпиад для учащихся трех возрастных дивизионов (I—IV классы, V—VIII классы, IX—XI классы);
- дифференцированное интернет-обучение, направленное на подготовку к указанным олимпиадам;
- собственное методическое обеспечение по изучению программирования и алгоритмизации.

Сайт дистанционного обучения DL.GSU.BY обеспечивает высокотехнологичную поддержку учебного процесса, в том числе:

- возможность выбора задач;
- автоматическую проверку решений (в том числе всех задач, приведенных в данной статье);
- представление необходимой теории;
- форум для интерактивного взаимодействия «ученик — тьютор» при дистанционном обучении;
- таблицы результатов для повышения мотивации.

Разработка собственного методического обеспечения связана с тем, что традиционные подходы рассчитаны на обучение как минимум старшеклассников, автору же приходится модифицировать эти подходы в сторону более простого и наглядного объяснения, а также медленного продвижения по учебному материалу, явно выделяя и обозначая все этапы этого продвижения.

В данной статье приводится пример такого методического материала для обучения школьников решению задач по информатике на тему «Исследование». Задача на эту тему является восьмой из десяти задач гомельских региональных олимпиад для учащихся V—VIII классов. Первые шесть задач предлагаются на темы: «Введение в программирование», «Одномерный массив», «Двумерный массив», «Геометрия», «Строки», «Сортировки». Седьмая задача — текстовая, девятая — на тему «Жадный алгоритм» и десятая — на тему «Очередь».

Очень важное для участника олимпиад по информатике и программированию умение — *исследовать пространство решений задачи*. Условие в таком случае может быть предельно кратким, но в то же время оно должно быть понятным. Основная проблема в задаче на исследование — придумать, как же можно получить результат. На сегодняшний день для постановки таких задач мы используем задания международного математического конкурса «Кенгуру» для учащихся V—VI классов, переформулированные в задания

на разработку программы. С одной стороны, для решения задачи необходимо будет провести исследование, а с другой — реализовать придуманное решение на языке программирования. Как правило, наши задачи обобщают «Кенгуру»-задания, поскольку вводится параметризация — ведь входные данные являются переменными, а не константами.

Материал может быть интересен для учителей — как в качестве иллюстрации методики обучения, так и по содержанию, — и в то же время он может оказаться

полезным и интересным для школьников и студентов, занимающихся самообучением.

Читателям предлагается следующий порядок работы: откладывать статью в сторону и пытаться самостоятельно выполнить предлагаемое задание: первый раз — после прочтения условия задачи, второй раз — после прочтения указаний к решению.

В таблице представлены темы и названия всех задач на исследование, предложенных в рамках гомельских региональных олимпиад по информатике и программированию.

Таблица

Оператор присваивания		13_гАпр*. Решить систему
Оператор условия If		12_оМай. Часы
Цикл For	Генерация повторений	12_шАпр. Разность
	Перебор значения переменной	12_гАпр. Монеты 15_гНоя. Максимальный остаток 18_гАпр. Двухзначное число
	Перебор диапазона чисел	14_оМай. Сумма трех цифр 13_оМай. Теми же цифрами 15_шМар. Насколько больше 16_шОкт. Считаем цифры
Вложенные циклы For	Сортировка	14_шМар. Катины карточки
	Перебор значений двух переменных	13_шОкт. Наибольшая разность 16_гНоя. Сахар
	Перебор значений трех переменных	12_гНоя. Карточки
	Перебор сочетаний цифр	15_гАпр. 1-2-3 11_шОкт. Хитрые цифры 11_гНоя. Цифровые уравнения 17_гАпр. Разность 17_гНоя. $KEN + GYRY = x$ 13_гНоя. $KEN = GU * RU$ 15_оАпр. $KE + HГ = УРУ$ 12_шОкт. Головоломка
Цикл While		13_шМар. Воробей Джек 14_гАпр. Буратино 14_шОкт. Бабушкины яблоки 14_гНоя. Вставить цифру 16_гАпр. Извлекая корни 16_оАпр. Последняя цифра 17_оАпр. Сумма цифр
Элементы теории чисел	Проверка на простоту по определению	15_шОкт. Сколько простых
	Системы счисления	17_шОкт. Двоичные числа 18_оАпр. Несложное вычисление
	Битовая обработка	17_шМар. Максимальная сумма 18_шМар. Пицца
Стереометрия		16_шМар. Наибольший объем

* Цифры обозначают год проведения олимпиады, буквы ш, г, о обозначают, какая это олимпиада — школьная, городская, областная соответственно. Например: «18_гАпр. Двухзначное число» означает, что задача «Двухзначное число» была предложена на городской олимпиаде в апреле 2018 года.

Задача 1 «Двузначное число»

Лиза выбрала двузначное число, не делящееся на x , поменяла его цифры местами и вычислила разность полученных чисел.

Какое самое большое число она могла получить?

Формат ввода:

x — заданное число.

Формат вывода:

out — самое большое число.

Пример ввода:

10

Пример вывода:

72

Нужно исследовать все двузначные числа (в интервале от 10 до 99), не делящиеся на введенное число x . В каждом из них выделить цифры i_2 i_1 , составить число из этих цифр наоборот: i_1 i_2 и сравнить с текущим значением максимального max , запомнив большее из них.

Полный текст решения:

```
var
  x, max, i, i1, i2, i3, y: longint;
begin
  readln(x);
  max:=0;
  for i:=11 to 99 do
    if (i mod x) <> 0 then
      begin
        i1:=i mod 10;
        i2:=i div 10;
        i3:=i1*10+i2;
        y:=abs(i-i3);
        if y > max then max:=y;
      end;
  writeln(max);
end.
```

Задача 2 «Теми же цифрами»

2012 год записывается четырьмя цифрами: 2 0 1 2. В будущем этими же четырьмя цифрами год будет записываться еще пять раз: 2021, 2102, 2120, 2201, 2210.

Вам дается текущий год (ровно четыре цифры, первая цифра больше нуля). Вы должны указать, сколько раз в будущем год будет записываться точно теми же цифрами, что и текущий год.

Формат ввода:

Y — текущий год.

Формат вывода:

k — количество раз.

Пример ввода:

2012

Пример вывода:

5

Идея предлагаемого решения заключается в следующем:

- выделить цифры введенного числа и сохранить его в массиве $a[0..9]$.

$a[i]$ указывает, сколько раз встретилась цифра i во введенном году (x).

А затем в цикле по i от $x+1$ до 9999:

- выделять цифры числа i в массив $b[0..9]$;
- сравнивать массивы $a[0..9]$ и $b[0..9]$ и, если они равны, инкрементировать ответ.

Для выделения цифр используем процедуру *Split*, которая в качестве формальных параметров принимает целое число n и возвращает массив d цифр этого числа.

Для получения последней цифры числа n используется операция $n \bmod 10$.

Для сдвига числа вправо на одну цифру («вычеркивания» обработанной последней цифры) используется операция $n \div 10$.

Для сравнения массивов цифр $a[0..9]$ и $b[0..9]$ используем функцию *EQ*, которая осуществляет стандартный поиск пары неравных элементов $a[i]$ и $b[i]$. Если такая пара не найдена, то массивы совпадают и, значит, числа x и i состоят из одних и тех же цифр.

Полный текст решения:

```
type
  arrD=array[0..9] of longint;
var
  a, b: arrD;
  i, k, x: longint;

procedure Split(n: longint; var d: arrD);
var
  i: longint;
begin
  for i:=0 to 9 do d[i]:=0;
  for i:=1 to 4 do
    begin
      inc(d[n mod 10]);
      n:=n div 10;
    end;
end;

function EQ: boolean;
var
  i: longint;
begin
  i:=0;
  while (i<=9) and (a[i]=b[i]) do inc(i);
  EQ:= i>9;
end;

begin
  readln(x);
  Split(x, a);
  k:=0;
  for i:=x+1 to 9999 do
    begin
      Split(i, b);
      if EQ then inc(k);
    end;
  writeln(k);
end.
```

Задача 3 «Карточки»

На столе лежит много карточек. На каждой из них написано одно из трех чисел: 3, 13, 31. Какое самое маленькое количество карточек нужно взять, чтобы сумма всех чисел на них была равна N ? Гарантируется, что сумму N всегда можно набрать хотя бы одним способом.

Формат ввода:

N — требуемая сумма (целое, $0 < N < 10000$).

Формат вывода:

k — минимальное количество карточек.

Пример ввода:

47

Пример вывода:

3

Достаточно перебрать возможные количества карточек каждого вида от максимума к нулю и, как только набрана нужная сумма, вывести ответ и завершить выполнение программы.

Полный текст решения:

```
var
  n, i, j, k: longint;
begin
  readln(n);
  for i:=n div 31 downto 0 do
    for j:=n div 13 downto 0 do
      for k:=n div 3 downto 0 do
        if 31*i+13*j+3*k=n then
          begin
            writeln(i+j+k);
            exit;
          end;
        end;
      end;
    end;
  end.
```

Задача 4 «KEN = GU * RU»

В равенстве $KEN = GU * RU$ разными буквами обозначены разные ненулевые цифры, а одинаковыми буквами — одинаковые цифры. Найдите значение $X(K, E, N, G, U, R)$, если известно, что число KEN — самое маленькое из возможных.

Формат ввода:

X — один из символов K, E, N, G, U, R.

Формат вывода:

v — цифра, соответствующая символу X .

Пример ввода:

K

Пример вывода:

4

Перебираем число KEN от 123 до 999.

Перебираем оставшиеся цифры от 0 до 9.

Проверяем выполнение условия задачи. Когда находим, выводим нужную (указанную во вводе) цифру.

Полный текст решения:

```
var
  i, N, E, K, U, R, G: longint;
  C: char;
begin
  readln(C);
  for i:=123 to 999 do
    begin
      N:=i mod 10;
      E:=(i div 10) mod 10;
      K:= i div 100;
      if (N<>0) and (E<>0) and (K<>0) and
        (E<>N) and (K<>N) and (K<>E)
      then
        for U:=1 to 9 do
          for R:=1 to 9 do
            for G:=1 to 9 do
              if (U<>N) and (U<>E) and (U<>K) and
                (R<>N) and (R<>E) and (R<>K) and
```

```
(R<>U) and (G<>N) and (G<>E) and
(G<>K) and (G<>U) and (G<>R) and
(i=(10*G+U)*(10*R+U))
```

```
then
```

```
begin
```

```
{writeln(i, '=', G, U, '*', R, U);}
```

```
case C of
```

```
'N': writeln(N);
```

```
'E': writeln(E);
```

```
'K': writeln(K);
```

```
'G': writeln(G);
```

```
'U': writeln(U);
```

```
'R': writeln(R);
```

```
end;
```

```
halt;
```

```
end;
```

```
end;
```

```
end.
```

Заметим, что отладочная печать в комментариях намеренно оставлена в тексте программы в качестве подсказки о том, как проверять корректность решения во время отладки.

Задача 5 «Воробей Джек»

Воробей Джек прыгает по изгороди с колышка на соседний колышек. Каждый прыжок занимает у него одну секунду. Он делает A прыжков вперед, потом B прыжков назад ($B < A$), потом опять A вперед и B назад, и т. д.

За сколько секунд Джек доберется от колышка 1 до колышка N ? ($0 < N, A, B \leq 100$)

Формат ввода:

$N A B$

Формат вывода:

K — минимальное количество секунд.

Пример ввода:

11 4 1

Пример вывода:

14

Для решения задачи достаточно просто посекундно промоделировать процесс прыжков воробья:

k — номер колышка, на котором находится воробей;

t — номер секунды;

a — сколько прыжков осталось сделать вперед в текущем цикле;

b — сколько прыжков осталось сделать назад в текущем цикле.

Полный текст решения:

```
var
  n, a0, b0, a, b, t, k: longint;
begin
  readln(n, a0, b0);
  t:=0;
  k:=1;
  a:=a0;
  b:=b0;
  while k<n do
    begin
      if a>0 then
        begin
          inc(t);
          dec(a);
          inc(k);
        end
```

```
else
begin
  if b>0 then
  begin
    inc(t);
    dec(b);
    dec(k)
  end
else
begin
  a:=a0;
  b:=b0;
end
```

```
end;
end;
writeln(t);
end.
```

Список использованных источников

1. Долинский М. С. Гомельская школа олимпиадного программирования // Информатика и образование. 2015. № 7.
2. Долинский М. С., Кугейко М. А. Гомельская инструментальная система дистанционного обучения // Информатика и образование. 2010. № 11.

НОВОСТИ

В России будут развиваться онлайн-курсы для студентов и преподавателей

13 ноября 2018 года в МИА «Россия сегодня» состоялась пресс-конференция, посвященная началу работы Национальной электронной платформы педагогического образования (НЭППО).

Современное образование невозможно без высокого уровня компьютерной грамотности, включения всех информационных возможностей в учебный процесс, отметил заместитель директора Департамента управления имуществом комплексом и конкурсных процедур Минпросвещения России Борис Лосиков.

«Будут созданы не менее четырех тысяч онлайн-курсов, обеспечивающих освоение дисциплин, модулей, образовательных программ среднего, высшего и дополнительного образования», — отметил спикер. Он уточнил, что всего в России уже в текущем году планируется, что все заинтересованные смогут освоить 1500 онлайн-курсов.

Представляя «индивидуальность и особенности» НЭППО, интегрированной в 26 платформ приоритетного проекта «Современная цифровая образовательная среда в Российской Федерации», вице-президент Союза «Профессионалы в сфере образовательных инноваций» Дмитрий Коноплянский заверил, что «она обладает всеми элементами открытости: мы никого ничем не ограничиваем».

Платформа разработана так, чтобы, повышая квалификацию, преподаватели могли разрабатывать собственные курсы. «Задача наша состояла не в том, чтобы сделать какой-то обособленный продукт, а в том, чтобы создать платформу, которая жила бы своей жизнью», — подчеркнул он, выразив надежду, что «это будет лучшая платформа в части педагогического образования».

Вуз давно обратил внимание на эксперименты с онлайн-образованием, отметил проректор по учебной работе Московского городского педагогического университета (МГПУ) Дмитрий Агранат.

«Студенты сегодня экономят время, многие подрабатывают. И если информацию можно получить вне аудитории, они это делают», — сказал он, признав, что цифровой контент позволяет и педагогу более рацио-

нально распорядиться временем. В частности, уделить в аудитории «больше внимания развитию определенных навыков».

В Московском государственном психолого-педагогическом университете (МГППУ) существует целый факультет дистанционного обучения. «Опыт многолетний. В рамках государственных заданий апробировано порядка сорока дистанционных курсов для студентов с инвалидностью», — рассказала заместитель директора ресурсного центра учебно-методического сопровождения региональных ресурсных центров МГППУ Светлана Панюкова.

По ее мнению, инструменты платформы позволяют осуществить разработки, очень важные для данной категории студентов. Но также и «для педагогов, которые по тем или иным причинам лишились возможности приходить на занятия, но могут обучать дистанционно».

Развитие такого рода платформ вносит огромный вклад в процессы социализации этих ребят, в выборку лучших онлайн-курсов и обеспечение ими в том числе лиц с инвалидностью, подчеркнула она.

Участники пресс-конференции полагают, что традиционная философия «от цели через содержание к результату» не всегда выдерживает конкуренцию с новыми подходами. Однако важно понимать, что мы не можем заменить все «цифрой».

«Никто не заменит профессионального общения... Не «чему учить?», а «как учить?» — здесь очень много помощи должно быть создано в рамках профессиональной и методической поддержки именно преподавателя», — подчеркнула профессор кафедры технологии и профессионального обучения Института физики, технологии и информационных систем Московского педагогического государственного университета (МПГУ) Марина Вайндорф-Сысоева.

Эксперты рассказали, что по итогам освоения модулей в рамках НЭППО будут выдаваться удостоверения, которые, конечно, не должны стать главной целью обучения. «Надо менять свою ментальность так, чтобы учиться не ради удостоверения», — отметили они.

(По материалам «РИА Новости»)



Е. И. Мичкасская,

Центр мониторинга и оценки качества образования, г. Саранск, Республика Мордовия,

Н. В. Вознесенская,

Мордовский государственный педагогический институт им. М. Е. Евсевьева, г. Саранск, Республика Мордовия

ОСНОВЫ РАЗРАБОТКИ ЭФФЕКТИВНЫХ ОНЛАЙН-КУРСОВ

Аннотация

В статье анализируется актуальная проблема разработки онлайн-курсов. Рассмотрены основные требования, на которые следует обратить внимание разработчику курса. Описаны современные методы обучения, способствующие эффективной организации процесса онлайн-обучения. Приводятся примеры внедрения данных методов в онлайн-курс, базовые советы и примеры актуальных приемов, используемых при наполнении онлайн-курса.

Ключевые слова: онлайн-обучение, онлайн-курс, модуль, сторителлинг, дизайн-паттерн, нормализованный контент, активные методы обучения, интерактивные методы обучения, баскет-метод.

DOI: 10.32517/2221-1993-2018-17-9-52-56

Контактная информация

Мичкасская Елена Ивановна, методист Центра мониторинга и оценки качества образования, г. Саранск, Республика Мордовия; *адрес:* 4300001, Республика Мордовия, г. Саранск, ул. Федосеенко, д. 19; *телефон:* (834) 224-87-76; *e-mail:* gia@stoko.ru

Вознесенская Наталья Владимировна, канд. пед. наук, доцент, зав. кафедрой информатики и вычислительной техники Мордовского государственного педагогического института им. М. Е. Евсевьева, г. Саранск, Республика Мордовия; *адрес:* 430007, Республика Мордовия, г. Саранск, ул. Студенческая, д. 11а; *телефон:* (834) 233-92-84; *e-mail:* voznensenskaya.n@gmail.com

E. I. Michkasskaya,

Center for Monitoring and Assessing the Quality of Education, Saransk, The Republic of Mordovia,

N. V. Voznesenskaya,

Mordovian State Pedagogical Institute Named after M. E. Evseviev, Saransk, The Republic of Mordovia

THE BASICS OF DEVELOPING EFFECTIVE ONLINE COURSES

Abstract

The actual problem of developing online courses is analyzed in the article. The basic requirements that should be paid attention to the developer of the course are considered. The modern teaching methods that contribute to the effective organization of the online learning process are described. Examples of the implementation of these methods in the online course, basic tips and examples of current techniques used when filling the online course are given.

Keywords: online learning, online course, module, storytelling, design pattern, normalized content, active teaching methods, interactive teaching methods, basket method.

В настоящее время в России существует достаточно много платформ для онлайн-обучения, но большинство из них реализованы в электронной образовательной среде вузов и недоступны слушателю, который не является обучающимся данной образовательной организации. Для решения этой проблемы на федеральном уровне реализован и введен в опытную эксплуатацию ресурс «Современная цифровая образовательная среда в РФ»: <https://online.edu.ru/>. Концепция ресурса предполагает предоставление доступа к онлайн-курсам, разработанным и реализуемым разными организациями на разных платформах онлайн-обучения, всем категориям граждан и образовательным организациям всех уровней образования по так называемому принципу одного окна.

В соответствии с Приказом Министерства образования и науки РФ от 23 августа 2017 года № 816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ» [4] организации вправе осуществлять реализацию образовательных программ или их частей с применением исключительно электронного обучения, дистанционных образовательных технологий и организацией учебных занятий в виде онлайн-курсов. Это позволяет обучающимся независимо от их места нахождения и организации, в которой они осваивают учебную программу, достижение и оценку результатов обучения путем формирования образовательной деятельности в электронной информационно-образовательной среде, к которой предоставляется открытый доступ с помощью сети Интернет.

Онлайн-курс — это организованная и целенаправленная совокупность форм, видов, средств учебной деятельности, обеспечивающая достижение конкретных целей и формирование компетенций, прописанных в образовательных программах высшего образования, реализуемая с использованием только электронного обучения, дистанционных образовательных технологий на основе комплекса связанных между собой в рамках единого педагогического сценария электронных образовательных ресурсов.

Онлайн-обучение сегодня реализуется во всех ведущих вузах России, а в остальных идет активное внедрение предоставления такой услуги [1]. Отличительная черта онлайн-обучения, реализуемого в современной образовательной организации, — создание общей ресурсной базы на единой технологической платформе, формируемой для различных форм организации процесса обучения, направленного на различные целевые группы. При этом эффективность управления онлайн-обучением зависит от возможности оптимизировать расходы на его реализацию, не понижая качества обучения.

Наиболее распространенные в нашей стране платформы, используемые для разработки и внедрения онлайн-курсов:

- Udemу;
- Courson;
- Uniweb;
- «Универсарииум»;
- HTML Academy;
- Национальный Открытый Университет «ИНТУ-ИТ»;
- «Открытое образование».

Каждая платформа обладает рядом преимуществ по сравнению с другой.

Так, в *Udemу* реализуется принципиально новый вид онлайн-курса — по запросу слушателя. Платформа синхронизируется с мобильными персональными устройствами iPhone, iPad и Android, поддерживает русский язык, позволяет найти узкоспециализированный курс, особенно в области разработки программного обеспечения, познакомиться с обучающимися из разных стран.

Courson — одна из популярнейших российских платформ в сфере организации онлайн-обучения. Предлагает для изучения профессионально снятые платные и бесплатные видеокурсы с картинкой в высоком разрешении, текстовым сопровождением, возможностью выдачи сертификата.

Особенности организации онлайн-курса в *Uniweb*:

- выбор для изучения курсов, созданных крупнейшими вузами страны (МГИМО, МГУ и т. д.);
- гибкий график обучения;
- индивидуальные консультации с преподавателями;
- сертификат государственного образца.

Политика «*Универсарииума*» строится на построении единой сетевой межуниверситетской площадки с привлечением ведущих университетов, лучших преподавателей страны, на использовании современных технологий и методик, интерактивных домашних заданий, системы кросс-проверки домашних работ, разбиения курса на модули.

Онлайн-курсы в *HTML Academy* строятся на выполнении множества заданий, приближенных к реальным. Обучающийся в соответствии с указаниями системы изменяет HTML- и CSS-коды задачи в онлайн-редакторе. Итоговым испытанием для обучающегося является верстка макета, специально разработанного с учетом пройденного материала.

На сайте *Национального Открытого Университета «ИНТУИТ»* можно одновременно изучать несколько курсов, по результатам прохождения которых предусмотрена выдача бесплатных электронных сертификатов. В 2010 году НОУ «ИНТУИТ» получил лицензию на образовательную деятельность. Материалы «ИНТУИТ» активно используются в учебном процессе многих вузов РФ, а в некоторых результат прохождения курса может быть засчитан в качестве одного из вариантов отчетности по соответствующей дисциплине.

Принципиальной особенностью платформы «*Открытое образование*» является то, что все онлайн-курсы разрабатываются в соответствии с требованиями ФГОС к результатам изучения образовательных программ, реализуемых в вузах. «Открытое образование» сотрудничает с вузами страны, предоставляя им возможность получать полную информацию об успеваемости своих студентов.

Большая часть существующих онлайн-курсов базируется на **принципе ускоренного изучения материала**, и обучающимся предлагается для исследования *типовое* решение поставленной задачи. При таком подходе к организации процесса обучения любая *нестандартная* постановка вопроса (задания) приводит к тому, что обучающиеся теряются и не могут решить возникшую проблему. На данный момент можно утверждать, что *онлайн-курсы формируют у обучающихся шаблонное мышление*.

На сегодняшний день большинство слушателей при изучении онлайн-курса сталкиваются с **трудностями самоорганизации процесса обучения**, грамотного распределения учебной нагрузки. Планируя самостоятельное изучение учебного материала, обучающийся должен отводить на выполнение практических работ большее количество часов, чем на теоретическую часть, при этом выделять время на общение с другими участниками курса.

Обязательное условие **общения в онлайн-курсах** — это активность педагога и обучающихся, синхронизация (или асинхронизация), учет количества участников. Общение в онлайн-курсах — это особый жанр педагогического общения, феномен которого на данный момент остается малоизученным.

Неотъемлемая часть образовательного процесса в современных вузах — формирование электронной информационно-образовательной среды, компонентом которой являются онлайн-курсы.

Разработка качественного онлайн-курса и поддержание его функционирования требуют наличия определенных ИКТ-компетенций у его создателя-преподавателя, а также достаточного количества времени, которое он сможет уделять поддержке данного курса.

Разработчик собственного онлайн-курса должен:

- определить тематику и тип курса (информационный, мотивационный, тренинговый, с элементами геймификации);
- детально продумать структуру курса (структура — важнейший невидимый атрибут курса, правильная организация которого придает курсу логичность, читаемость, он легче воспринимается и запоминается) и его содержание;
- подобрать:
 - дизайн курса (ориентируясь на основное правило: 90 % — основной стиль, 10 % — акценты);
 - интерактивные ресурсы курса;
 - способы взаимодействия с обучающимися;
 - формы оценивания результатов их деятельности и т. д.

Следует отметить, что дублирование в онлайн-курсе содержания учебников и практикумов чаще всего дает отрицательный результат и формирует негативный опыт у слушателей.

Принято разделять учебные онлайн-курсы по сложности организации сценария и типу курса на несколько уровней [3]:

- **легкий уровень — информационные курсы:** изучение необходимой информации с целью дальнейшего использования полученных знаний на практике. Это основной способ организации изучения учебного материала многих курсов. Такие курсы можно применять при изучении новой темы, программы, нормативного документа и т. д. При разработке сценария такого курса до-

статочно описать, о чем и в какой последовательности будет сообщаться на конкретной странице (слайде);

- **средний уровень — мотивационные курсы:** основная задача таких курсов — заинтересовать слушателя с целью изменения его отношения к чему-либо — к учебной или творческой деятельности, к работе. Вовлечение слушателя в процесс обучения возможно благодаря созданию главного персонажа, от лица которого происходит повествование, моделированию множества практических заданий;
- **сложный уровень — тренинговые курсы:** на этих курсах обучающиеся улучшают полученные навыки или развивают новые.

Начальный этап создания любого онлайн-курса — составление краткого описания концепции курса.

Разработчику следует:

- продумать:
 - форму представления курса;
 - формы взаимодействия пользователей;
 - формы отчетности;
- согласно тематике курса подобрать и адаптировать релевантный, аутентичный и интересный учебный контент, внешние и внутренние электронные образовательные ресурсы;
- подготовить практические задания.

Желательно перед каждым набором слушателей на курсы производить реконструирование порядка изучения теоретического материала, выполнения практических работ, проектов, т. е. **адаптацию курса под конкретную аудиторию**. Это необходимо потому, что уровень овладения знаниями разных целевых аудиторий может существенно отличаться. В этом случае идеальным решением является создание онлайн-курсов по запросу слушателей.

Структура и построение сценария онлайн-курса определяются его типом. Например, для курса с элементами геймификации **рациональное построение сюжета курса** достигается путем использования классической трехактной структуры повествования:

- погружение в конфликт;
- развитие событий;
- развязка.

Слушатель получает заведомо невыполнимое на первоначальном этапе задание. Для разрешения проблемы ему приходится изучить и проанализировать учебный материал, а в конце — пройти очередное испытание, в случае успешного прохождения которого обучающийся подтверждает усвоение полученных знаний.

Тематический план и учебное расписание онлайн-курса практически не отличаются от привычного обучающемуся расписания занятий. Преподаватель, в зависимости от определенной структуры курса, выстраивает изучение тем в нужной последовательности, разбивает онлайн-курс на модули.

Модуль — достаточно большой логически связанный блок информации, мини-курс. В модуле создается внутренняя логика курса: последовательная подача информации, линейное изучение взаимосвязанных тем. Сам модуль разбивается на части:

- введение;
- основная часть;
- заключение.

В заключение кроме краткого резюме и послесловия желательно добавлять тест, проект, кейс или диалоговый тренажер для практики.

Для закрепления изученного материала следует предлагать **обучающие практические задания**. Преподавателем в настройках соответствующего задания указывается количество попыток ответа на вопрос (одна и более), неограниченное время прохождения и низкий проходной балл.

Проверяющие задания необходимы для осуществления контроля усвоения материала и оценки знаний слушателей. Для таких заданий преподаватель устанавливает различные ограничения, например, конкретизирует время выполнения задания (сколько минут), повышает проходной балл. Принципиально учитывать, что практическое задание может присутствовать в каждом модуле онлайн-курса, в то время как контрольный тест размещается в самом конце отдельных разделов или всего курса.

Распространенной ошибкой при наполнении онлайн-курса является перенос в него традиционного учебного материала и традиционного учебного процесса, т. е. организация изучения материала онлайн-курса по аналогии с офлайн-занятием. В современных условиях всеобщей цифровизации помимо традиционных методов подачи учебного материала и оценивания знаний обучающихся **эффективно использовать современные нестандартные способы подачи учебного контента**, формат представления которого зависит от содержания лекций [3].

Необходимо строить структуру материала, проектирующую несколько слоев чтения, т. е. заменять традиционный вариант последовательного чтения текста на **кросс-чтение (скимминг)**: чтение идет только по заголовкам, врезам и т. д. В этом случае, если в тексте встречается что-то интересное, внимание акцентируется именно на этом.

Результативно внедрять в онлайн-курс **дизайн-паттерн** — какой-либо элемент или совокупность элементов (объектов, слов, фигур и пр.), повторяющихся на страницах курса с определенной целью. С помощью паттерна можно придать индивидуальность и динамику курсу, что позволит выделить его из огромного количества существующих онлайн-курсов.

При наполнении контентом онлайн-курса актуально использовать картинки, скринкасты и видеолекции. В технологии создания скринкастов нет ничего сложного, но предпочтительнее записывать видеолекции с преподавателем, так как они позволяют создать эффект соучастия, более интересны для обучающихся. Предоставляемый таким образом материал лучше воспринимается и с удовольствием изучается обучающимися.

Изучение хронологии исторических событий всегда сложно для восприятия человеком. Поэтому **при изучении тем, связанных с хронологией, рекомендуется использовать инфографику** — наглядный способ подачи информации с помощью текста и изображений. Благодаря инфографике повышается возможность запоминания важных дат, становятся доступными для понимания самые сложные концепции. Данный способ подходит и для отображения статистических данных или наглядных советов. Однако требуется наличие базовых знаний для работы с программами построения инфографики.

Чтобы слушатели не сомневались в качестве размещенного в онлайн-курсе контента, приводя какие-то факты, всегда следует ссылаться на авторитетные источники. Например, в онлайн-курсе для будущих учителей (или для системы повышения квалификации) в теме, посвященной формированию информационно-образовательной среды образовательной организации или электронного портфолио, в достоверных источниках стоит найти статьи, в которых рассматриваются трактовки соответствующих понятий.

Онлайн-обучение предполагает использование современных информационных технологий и методов обучения — активных и интерактивных, основой функционирования которых является организация групповой работы, обмен знаниями, взаимодействие обучающихся между собой, обучающихся и преподавателя. Не исключается применение и классических методов обучения.

Метод поиска необходимой информации в сети Интернет — классический метод обучения в онлайн-курсах. Практический пример использования данного метода при подготовке/переподготовке педагогов — поиск информации на информационно-правовых порталах («Гарант», «КонсультантПлюс» и т. п.), в специализированных системах организации и сопровождения профессиональной деятельности учителей. Этот метод также включает просмотр познавательных или образовательных видеороликов, фильмов, затрагивающих конкретные темы или фрагменты учебного материала. В этом помогут широко известные видеохостинги, например, Викисклад, YouTube, RuTube и пр. Просмотр видео, новостей, телепередач, ток-шоу позволяет привлечь внимание обучающихся, максимально погрузить их в процесс изучения учебного материала. Главное условие — все это должно быть привлекательно для обучающихся.

Выполнение заданий по образцу — не менее распространенный метод обучения при организации самостоятельной работы обучающихся. Особенно эффективен данный метод при изучении нового программного продукта, когда обучающиеся посредством повторения действий преподавателя знакомятся с интерфейсом некоторого приложения и с основными правилами работы в рассматриваемом продукте. Этот метод позволяет выработать у обучающихся практические навыки.

«Естественный» метод обучения основан на непосредственной практике обучающегося в использовании полученных знаний в определенных учебных ситуациях. Для осуществления данного метода необходимо создание небольших групп обучающихся, причем у них должна присутствовать высокая мотивация, так как в искусственной среде довольно тяжело генерировать естественные ситуации, а также гарантировать понимание и достаточную практику для всех участников. Этот метод предусматривает использование вебинаров, онлайн-конференций.

Коммуникативный метод обучения основывается на идее изучения материала в реальных ситуациях. В этом методе преподаватель содействует процессу коммуникации между всеми участниками, выступая руководителем группы, контролируя процесс обучения, задавая мотивацию обучающимся, вовлекая их в активное обсуждение рассматриваемой проблемы. В данном методе используются такие средства, как Skype, социальные сети, чаты, форумы.

Примерами заданий на использование описанных методов в онлайн-курсе, предназначенном для подготовки/переподготовки педагогов, могут служить:

- поиск основных законов, нормативных документов, законодательных актов в сфере образования;
- поиск электронно-образовательных ресурсов на информационно-образовательных порталах;
- просмотр видеоресурсов на видеохостинге по образовательным темам;
- создание электронного журнала в системе ведения учета успеваемости в электронном виде по образцу;
- участие в вебинаре, ответы на вопросы;
- обсуждение проблемной ситуации, предложенной преподавателем в форуме.

Повысить эффективность онлайн-курса, заинтересовать слушателей можно с помощью **организации взаимной проверки выполненных заданий**. Например, обучающемуся предлагается проверить пять работ своих сокурсников, высказать замечания, написать отзыв о работах. Это позволяет слушателю работать с учебным материалом, осваивая его через *продуктивную оценочную деятельность*. Эффективно использование чек-листов, когда составленный перечень контрольных вопросов/заданий/критериев позволяет понять последовательность действий для выполнения какой-либо задачи. Пример реализации данного метода в онлайн-курсе — составление обучающимися чек-листа после изучения некоторой темы.

Для обеспечения высокой мотивации и вовлечения в учебный процесс можно использовать **баскет-метод**, который является методом моделирования коллективного результата. Суть данного метода состоит в том, что обучающийся не только учится самостоятельно, но и обучает своих одноклассников. Для реализации этого метода в онлайн-курсы встраиваются элементы, предполагающие совместную работу (электронные рабочие тетради, чаты, форумы, онлайн-встречи и др.).

Удерживать внимание слушателя — целая наука. Практическим путем доказано, что через 20 минут после начала занятия внимание человека ослабевает. Поэтому мудрый преподаватель обязательно организует небольшой перерыв или сменит вид деятельности. Внимание интернет-пользователя еще более расфокусировано. Постоянно возникают какие-то отвлекающие факторы внешнего мира, появляется желание открыть сторонний сайт, зайти в социальную сеть или заняться интернет-серфингом. По этим причинам **контент онлайн-курса должен быть максимально вовлекающим**, чтобы даже пятиминутное изучение материала оставляло какую-то полезную информацию и желание вернуться к нему снова.

Нестандартным способом привлечения внимания слушателей является **применение технологии сторителлинга** — рассказывания историй с использованием метафор, а также сюжетов, лежащих в основе сказок и литературных произведений. Характерная черта сторителлинга — красочное визуальное оформление и постоянное добавление поучительных историй в онлайн-курс. Такой подход позволяет организовать комплексное воздействие на восприятие: интересный рассказ дополняется визуальным образом, музыкой, интерактивными элементами (3D-книги, шкала времени, творческие задания). Внедрение в курс данного подхода весьма сложно, так как добиться интерпретации

известных произведений в соответствии с тематикой создаваемого курса — нетривиальная задача. Однако если данная задача решена, то созданный на основе сторителлинга онлайн-курс обладает рядом преимуществ перед другими курсами: он актуальный, современный, позволяет получить позитивный опыт, усвоить учебный материал на качественно новом уровне.

Эффективное управление образовательным процессом во многом определяется знанием **приемов привлечения внимания слушателей**.

Практика разработки и внедрения онлайн-курсов позволила нам выделить наиболее важные способы удержания внимания слушателей:

- запись вебинара продолжительностью не более 1–1,5 часа;
- запись коротких видеоуроков примерно по 8–13 минут;
- оптимальный объем текста — три страницы Word, которые при необходимости снабжаются иллюстрациями, графиками, мини-заданиями;
- продолжительность изучения видеокурса или курса из текстовых заданий — две-три недели, с онлайн-занятиями — до 45 дней;
- выполнение домашних заданий — 10 минут;
- занятия проводятся ежедневно или два-три раза в неделю по будням;
- идеальные дни для проведения онлайн-занятий — суббота до обеда или вечер четверга.

Важно помнить, что самостоятельное изучение онлайн-курса обучающимися — без положительных отзывов, советов от преподавателя — сводит к нулю их первоначальную мотивацию. Такой онлайн-курс не получает необходимого количества откликов, рекомендаций и остается невостребованным. Поэтому **взаимодействие преподавателя со слушателями**, постоянная поддержка с его стороны — одно из важнейших условий успешного функционирования онлайн-курса.

При организации такого взаимодействия следует [3]:

- внедрить элементы геймификации — активно раздавать баллы, бонусы, дополнительный материал за выполненные задания и прогресс, лучших слушателей поощрять скидками (например, продлить срок сдачи проекта), индивидуальными консультациями;
- установить дисциплину — за задержку изучения материала, невыполнение самостоятельных работ исключать из группы;
- проводить онлайн-встречи примерно один раз в неделю для совместного разбора сложных моментов, заданий из теории и практики;
- создать чат для активного общения слушателей;
- сопровождать слушателя — писать ему мотивационные сообщения, вовремя проверять домашние задания;

- мотивировать кейсами — демонстрировать успехи и результаты других участников, заинтересовывать в создании проекта.

* * *

Таким образом, онлайн-курс предполагает организацию контента в простой, понятной, легко воспринимаемой со стороны слушателя форме, с четкими смысловыми блоками, выделенными ключевыми понятиями, визуализацией данных. Преподаватель обязан детально прорабатывать каждую лекцию, практическое задание, описывать любое домашнее задание, подбирать соответствующий значимый дополнительный материал из достоверных источников.

Онлайн-курсы будущего — это курсы, формируемые по запросам клиента и реализующие индивидуальную образовательную траекторию, связанную с интересами и профессиональной областью деятельности. Для реализации такого решения требуются формирование кардинально обновленного, «нормализованного» контента (небольших уроков, минимально связанных друг с другом) [1], активное использование сервисов для проверки текста по различным критериям (грамотность, штампы, повторы; например, «Главред», Plainrussian).

Понятно, что невозможно разработать идеальный онлайн-курс. Однако поддержка обратной связи со слушателями позволяет не только исправить недостатки курса, но и усовершенствовать его, так как наиболее активные слушатели замечают недочеты и ошибки в созданном продукте. К тому же наличие активного сообщества в онлайн-курсе — один из вариантов продвижения курса. Если разработанный курс включает интересный и актуальный материал, представленный в непривычной для обучающихся форме, то уже сами слушатели будут привлекать дополнительную аудиторию к изучению такого курса.

Список использованных источников

1. Вознесенская Н. В. Смена парадигмы сайта современного вуза // Информатика и образование. 2013. № 9. С. 90–96.
2. Кирьянов Д. «Динамические» онлайн-курсы потребуют нового «нормализованного» контента // Хабр. <https://habr.com/company/nerepetitor/blog/300566/>
3. Мичкасская Е. И., Вознесенская Н. В. Особенности разработки онлайн-курсов для реализации обучения с применением дистанционных образовательных технологий // Учебный эксперимент в образовании (Саранск: МГПИ им. М. Е. Евсевьева). 2018. № 2 (86). С. 60–70.
5. Приказ Министерства образования и науки Российской Федерации от 23.08.2017 года № 816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ». <https://rg.ru/2017/09/21/minobr-prikaz816-site-dok.html>

Д. А. Чечко,

Инновационный образовательный центр «Мой университет», г. Петрозаводск, Республика Карелия,

А. В. Радинова,

Петрозаводский государственный университет, Республика Карелия

ВИДЕОКУРС «ЛИЦЕВАЯ АНИМАЦИЯ В BLENDER»

Аннотация

Одним из наиболее сложных направлений, связанных с трехмерной графикой, является анимация лица. Для ее создания необходимы знания анатомии лицевых мышц, методов и приемов лицевой анимации, навыки работы в трехмерном редакторе. В статье описан курс из шести видеоуроков по лицевой анимации в пакете обработки 3D-графики Blender. Курс знакомит с основными этапами двух методов создания лицевой анимации: использование ключей формы и трекинг. Разработанный видеокурс может применяться в образовательном процессе школы, но требует предварительного ознакомления с основами работы в Blender.

Ключевые слова: компьютерная графика, 3D-анимация, лицевая анимация, ключи формы, трекинг, Blender.

DOI: 10.32517/2221-1993-2018-17-9-57-60

Контактная информация

Чечко Денис Андреевич, программист
Инновационного образовательного центра
«Мой университет», г. Петрозаводск, Республика Карелия; *адрес:* 185035, Республика Карелия, г. Петрозаводск, ул. Фридриха Энгельса, д. 10, оф. 211; *телефон:* 8 (499) 685-10-45; *e-mail:* frequency@gmail.com

Радинова Анна Владимировна, преподаватель кафедры теории и методики обучения математике и ИКТ в образовании
Петрозаводского государственного университета, Республика Карелия; *адрес:* 185910, Республика Карелия, г. Петрозаводск, ул. Ленина, д. 33; *телефон:* (8142) 71-10-80; *e-mail:* radionovaann@gmail.com

D. A. Chechko,
Innovative Education Centre "My University", Petrozavodsk, The Republic of Karelia,

A. V. Radionova,
Petrozavodsk State University, The Republic of Karelia

THE videocourse "FACIAL ANIMATION IN BLENDER"

Abstract

One of the most difficult areas of 3D computer graphics is a facial animation. To create it, you need knowledge of facial muscles anatomy, methods and techniques of facial animation, skills in a 3D modeling software. The article describes facial animation tutorials in Blender. The tutorials introduce two methods of facial animation: using shape keys and motion tracking. The tutorials can be used at school, but requires basic Blender knowledge.

Keywords: computer graphics, 3D animation, facial animation, shape keys, motion tracking, Blender.

Сегодня 3D-анимация применяется все более широко и в самых разных сферах. Основные потребители 3D-анимации и визуализации — медиаиндустрия, строительство, архитектура, наука, образование. В связи с активным развитием 3D-технологий на рынке труда возрастает спрос на специалистов в этой области. Таким образом, большое значение имеет подготовка обучающихся, имеющих представление об областях применения 3D-графики, обладающих базовыми знаниями, умениями и навыками в этой сфере. Изучение 3D-моделирования в школьном курсе информатики предусматривает УМК К. Ю. Полякова, Е. А. Еремина для X—XI классов углубленного уровня [3, 4]. В качестве средства обучения авторы УМК используют свободно распространяемый редактор трехмерной графики Blender [2]. Изучение 3D-моделирования и анимации способствует развитию пространственного мышления, воображения, творческих способностей обучаемых, их профессиональной ориентации.

Одним из наиболее сложных направлений, связанных с трехмерной графикой, является анимация лица. Лицевая анимация — важная составляющая любого анимированного персонажа, она позволяет не только оживить, индивидуализировать создаваемый образ, но и придать ему нужный уровень реалистичности, естественности, максимизировать сходство с реальным человеком или, наоборот, добавить выразительности, преувеличить определенные эмоции. Современные технологии позволяют как создавать анимацию вручную, уделяя внимание мельчайшим деталям, так и превращать игру живых актеров в движения трехмерной модели, при необходимости дополняя или редактируя их.

Для создания реалистичной лицевой 3D-анимации необходимы знания анатомии лица человека, владение основными принципами анимации (в том числе лицевой анимации), навыки работы в трехмерном редакторе и пространственное воображение [1, 5].

Анализ существующих немногочисленных обучающих материалов по созданию лицевой анимации в Blender показал, что для доступа к большинству системно выстроенных курсов (в основном англоязычных) необходима платная подписка.

В рамках выпускной квалификационной работы одним из авторов данной статьи Д. А. Чечко был разработан видеокурс, обучающий созданию анимации лица, отражающего различные эмоции, с использованием таких методов лицевой анимации, как ключи формы и трекинг. Разработанные видеоуроки ориентированы на обучающихся, имеющих опыт работы в программе 3D-моделирования.

Один из способов создания лицевой анимации предполагает ручное создание анимации с использованием рига (управляющих конструкций) и ключей формы (shape keys). Другой способ основывается на получении информации о движении лица с помощью трекинга (tracking) — процесса определения местоположения и ориентации движущегося объекта в виртуальной среде [6]. Различные виды трекинга широко используются в киноиндустрии, рекламе, анимации. Трекинг, в частности, позволяет переносить движения лица актера из видеоряда на трехмерную модель персонажа. Данные два способа создания анимации обычно применяются в сочетании друг с другом.

Обучающимся предлагается видеокурс:
<https://www.youtube.com/playlist?list=PL7oY--vm2zBNp-L4HFfRfRagofANW9OF>
раскрывающий оба указанных способа анимации.

При выполнении работы можно использовать готовую 3D-модель головы персонажа (рис. 1), ссылка на необходимые для работы файлы размещена в блоке описания под первым видеоуроком курса.

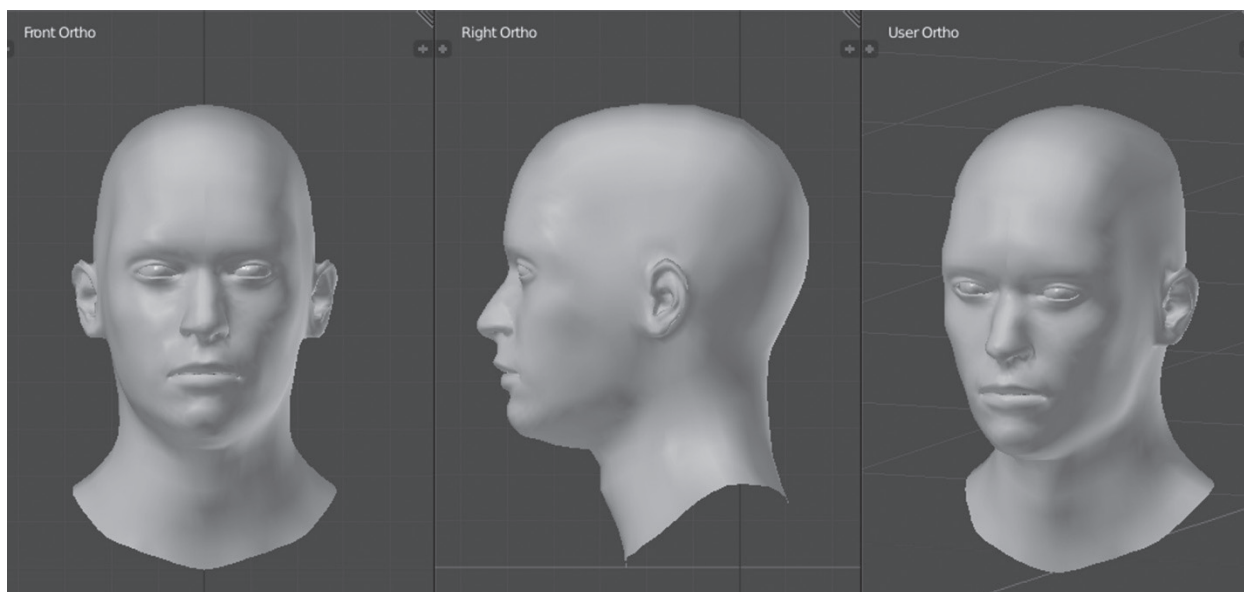


Рис. 1. 3D-модель головы персонажа



Рис. 2. Влияние изменения значений ключей формы на положение век

Рассмотрим краткое описание уроков по созданию анимации с использованием ключей формы.

Урок 1. Введение, ключи формы глаз.

В рамках урока дается общая информация о способах создания лицевой анимации, обсуждаются принципы лицевой анимации, вводится понятие «ключи формы», рассматриваются особенности, преимущества, ограничения относительных ключей формы. В резуль-

тате выполнения урока обучающиеся создадут четыре ключа для раздельного управления движениями верхнего и нижнего века.

Ключи формы представляют собой зафиксированное положение меша*, созданное путем перемещения

* *Меш* (англ. mesh — петля, ячейка сети, отверстие сита; полигональная сетка, полисетка) — 3D-объект, который представляет собой совокупность вершин, ребер и полигонов [6].

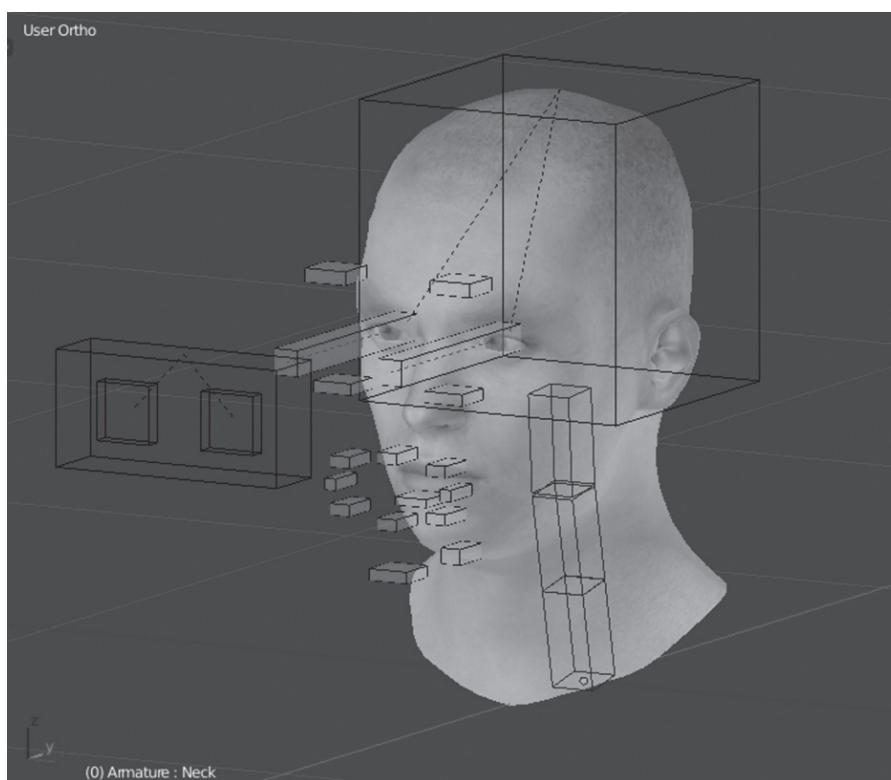


Рис. 3. Лицевой риг

вершин в режиме редактирования. В лицевой анимации используются относительные ключи формы, которые фиксируют изменение меша относительно его базового состояния (нейтрального, безэмоционального выражения лица). Слайдер ключа формы позволяет гибко управлять деформацией меша: значение «0» соответствует базовой форме меша, «1» — деформации меша, определяемой данным ключом. Кроме того, можно расширить диапазон возможных значений ключа формы, например, использовать отрицательные значения [7]. Таким образом, для формирования закрытого и удивленно распахнутого глаза достаточно создать по одному ключу формы для каждого века (рис. 2). Используя комбинацию ключей формы, можно конструировать разнообразные выражения лица.

Урок 2. Ключи формы бровей, редактирование модели.

На уроке рассматривается создание двух ключей, отвечающих за поднятие бровей при удивлении или недоумении, и двух ключей, отвечающих за нахмуривание.

Урок 3. Ключи формы челюсти и губ.

Урок состоит из трех частей, описывающих создание 15 ключей, отвечающих за движения нижней челюсти, губ, языка.

Урок 4. Риггинг*, настройка драйверов.

В данном уроке вводятся понятия «риг», «риггинг», описывается структура создаваемого рига и назначение костей, демонстрируются настройка степени влияния костей на меш и управление ключами формы с помощью драйверов.

Риг состоит из трех частей (рис. 3):

1. Кости шеи и головы, которые управляют положением головы, отвечают за повороты и наклоны головы.
2. Кости, управляющие направлением взгляда персонажа.
3. Кости, обеспечивающие управление ключами формы. Преобразование меша между конечными состояниями (выбранным ключом формы и базисным ключом) связывается с перемещением кости по одной из координат с помощью создания и настройки драйвера.

Урок 5. Анимация на основе ключевых кадров.

Анимация создается на основе ключевых кадров, на которых помещается информация о созданном ранее наборе управляющих элементов. В уроке описана технология рендеринга анимации. На рисунке 4 приведено несколько выражений лица, созданных в рамках описанного блока уроков.

Урок 6. Анимация на основе трекинга.

Анимация на основе трекинга рассмотрена в рамках одного урока.

Использование трекинга в Blender состоит из следующих этапов:

1. Обучающимся предлагается видео, где актер, на лице которого размещены маркеры, передает различные эмоции. Учащимся необходимо импортировать данное видео в видеоредактор Blender.
2. В режиме трекинга в видеоредактор добавляются маркеры, соответствующие маркерам на лице актера. Далее осуществляется автоматическое отслеживание траекторий маркеров в видеоряде, при необходимости их ручная корректировка.

* *Rigging (rigging)* — процесс создания и настройки виртуального скелета (системы костей, рига) персонажа, а также использования других управляющих элементов с целью подготовки персонажа к анимации [6].

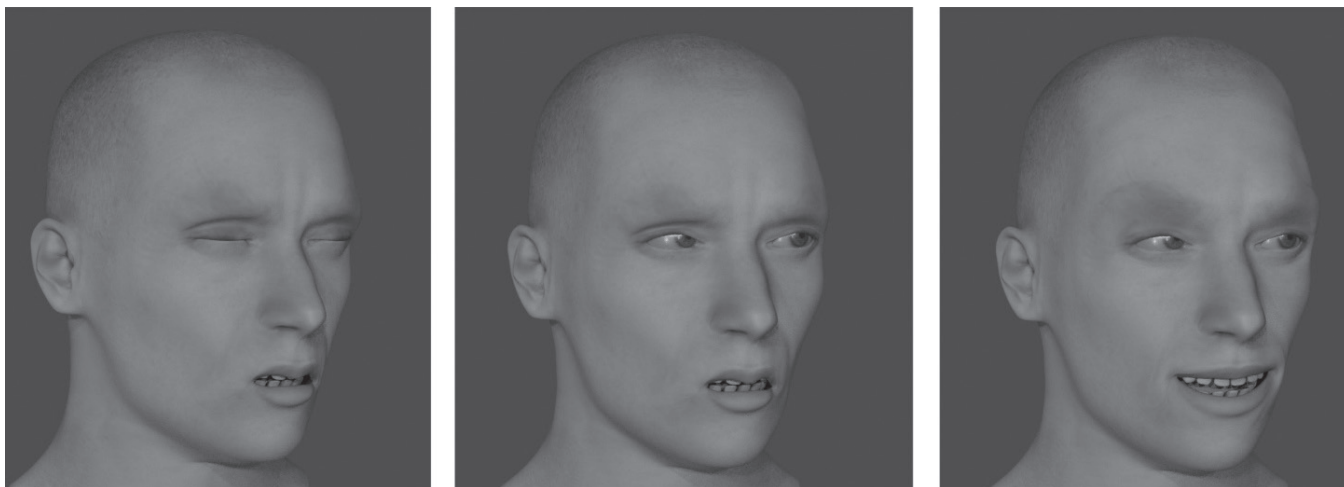


Рис. 4. Выражения лица, созданные путем комбинации различных ключей формы



Рис. 5. Процесс трекинга

3. Производится расчет ошибки трекинга, при необходимости маркеры добавляются, их траектории корректируются. При достижении приемлемой точности информация о параметрах камеры и положении маркеров добавляется на сцену.
4. Движение маркеров связывается с движением трехмерной модели персонажа: создается отдельная управляющая конструкция для модели, и движения маркеров преобразовываются в движения отдельных управляющих элементов (рис. 5).

Созданный видеокурс помимо описания технологии работы для создания анимации лица в конкретном программном средстве содержит теоретическое обоснование выполняемых действий. Таким образом, обучающиеся изучат теоретические принципы создания лицевой анимации, освоят основные этапы двух способов создания анимации, овладеют анимационной терминологией и особенностями работы в пакете Blender.

Список использованных источников

1. Осипа Дж. 3D-моделирование и анимация лица: методики для профессионалов. 2-е изд. / пер. с англ. М.: ИД «Вильямс», 2008.
2. Официальный сайт Blender. <https://www.blender.org/>
3. Поляков К. Ю., Еремин Е. А. Информатика. Углубленный уровень: учебник для 11 класса: в 2 ч. М.: БИНОМ. Лаборатория знаний, 2015.
4. Поляков К. Ю. Электронное приложение «Электронный практикум по информатике» к учебнику «Информатика. Углубленный уровень» для 10–11 классов // Преподавание, наука и жизнь: сайт Константина Полякова. <http://kpolyakov.spb.ru/school/probook/prakt.htm>
5. Флеминг Б., Доббс Д. Методы анимации лица. Мимика и артикуляция / пер. с англ. М.: ДМК Пресс, 2002.
6. Blender Manual. <https://docs.blender.org/manual/>
7. Chronister J. Blender basics. Classroom tutorial book. http://www.cdschools.org/cms/lib04/PA09000075/Centricity/Domain/81/BlenderBasics_4thEdition2011.pdf

Власти потратят на цифровую экономику 400 млрд руб. за три года

В бюджете на следующие три года, который одобрила Госдума, на реализацию национальной программы «Цифровая экономика» в 2019–2021 годах отведено порядка 403 млрд руб. Из них около 108 млрд руб. будет потрачено в 2019 году, чуть более 123 млрд руб. — в 2020 году и почти 172 млрд руб. — в 2021 году. Об этом свидетельствуют приложения к закону о бюджете, размещенные на сайте системы обеспечения законодательной деятельности (СОЗД).

Если говорить отдельно о шести направлениях, которые составляют нацпрограмму, то средства распределены следующим образом: на информационную инфраструктуру будет потрачено около 152 млрд руб. за три года, на цифровое госуправление — чуть более 101 млрд руб., на цифровые технологии и проекты — 88,5 млрд руб., на кадры для цифровой экономики — более 46 млрд руб., на информационную безопасность — более 15 млрд руб., на нормативное регулирование цифровой среды — 794 млн руб.

Если говорить конкретно о следующем 2019 году, то бюджетом предусмотрены такие расходы по данным шести направлениям: на информационную инфраструктуру будет потрачено около 41,7 млрд руб., на цифровое госуправление — 29,2 млрд руб., на цифровые технологии и проекты — 21,5 млрд руб., на кадры для цифровой экономики — 10,5 млрд руб., на информационную безопасность — 4,8 млрд руб., на нормативное регулирование цифровой среды — 265 млн руб.

Что касается 2020–2021 годов, то на реализацию направлений нацпрограммы в бюджете определены следующие суммы: на информационную инфраструктуру зарезервированы 48 млрд руб. и 62 млрд руб. соответственно, на цифровое госуправление — 31 млрд руб. и 41 млрд руб., на цифровые технологии и проекты — 25,5 млрд руб. и 41,6 млрд руб., на кадры для цифровой экономики — 13,3 млрд руб. и 22,4 млрд руб., на информационную безопасность — 5,6 млрд руб. и 4,9 млрд руб., на нормативное регулирование цифровой среды — по 265 млн руб.

По направлению «Информационная инфраструктура» на обеспечение покрытия объектов транспортной инфраструктуры сетями связи с возможностью беспроводной передачи голоса и данных в 2019–2021 годах предусмотрено 500 млн руб., 2,8 млрд руб. и 4,4 млрд руб. соответственно.

В рамках направления «Цифровое госуправление» на обеспечение информационно-аналитического и экспертно-аналитического сопровождения в сфере контрольной и надзорной деятельности предусмотрено 153,4 млн руб. в год. На создание национальной системы управления данными в 2019–2021 годах зарезервировано 510,5 млн руб., 502,8 млн руб. и 509,8 млн руб. соответственно.

Направление «Кадры для цифровой экономики» включает в себя такое мероприятие, как проведение тематических смен в сезонных лагерях для школьников по передовым направлениям дискретной математики, информатики, цифровых технологий, на что в ближайшие три года планируется выделять более 470 млн руб. ежегодно.

В рамках этого же направления на развитие и распространение лучшего опыта в сфере формирования цифровых навыков в 2019 году и 2020 году предусмотрено около 1,8 млрд руб., а в 2021 году — 2,7 млрд руб. На предоставление грантов физическим лицам на реализацию проектов в области цифровой экономики зарезервировано 250 млн руб. ежегодно. На тиражирование лучших практик по развитию цифровой грамотности школьников на базе русских школ за рубежом планируется потратить 309 млн руб. в 2019 году и 340 млн руб. — в 2021 году. Сумма за 2020 год в приложении к закону о бюджете отсутствует.

По направлению «Информационная безопасность» на доведение уровня безопасности объектов критической информационной инфраструктуры до установленных законодательством зарезервировано 150 млн руб. в 2020 году и 100 млн руб. в 2021 году. Значение за 2019 год не указано. На создание и функционирование Центра мониторинга и управления сетью связи общего пользования, а также создание, эксплуатацию и развитие информационной системы мониторинга и управления сетью связи общего пользования предусмотрено в 2019–2021 годах 597 млн руб., 699 млн руб. и 546 млн руб. соответственно.

На обеспечение защиты прав и законных интересов личности, бизнеса и государства от угроз информационной безопасности в условиях цифровой экономики планируется выделить 834 млн руб., 454 млн руб. и 409 млн руб. в год. Создание условий для глобальной конкурентоспособности в области экспорта отечественных разработок и технологий обеспечения безопасности информации обойдется в течение трех лет в 104 млн руб., 119 млн руб. и 139 млн руб. в год.

Государственная поддержка автономной некоммерческой организации «Центр компетенций по импортозамещению в сфере информационно-коммуникационных технологий» составит 34,7 млн руб. ежегодно.

По направлению «Цифровые технологии и проекты» на реализацию комплекса мер по поддержке разработки и внедрения российскими организациями технологий, продуктов, сервисов и платформенных решений в целях повышения доступности российского ПО и программно-аппаратных комплексов для цифровой трансформации приоритетных отраслей экономики и социальной сферы в 2019–2021 годах предусмотрено 2 млрд руб. ежегодно.

На разработку и реализацию дорожных карт развития перспективных «сквозных» цифровых технологий зарезервировано 2,1 млрд руб., 2,8 млрд руб. и 6 млрд руб. в год. Государственная поддержка компаний — лидеров по разработке продуктов, сервисов и платформенных решений на базе «сквозных» цифровых технологий обойдется бюджету в 3 млрд руб., 2,5 млрд руб. и 5,5 млрд руб. в год.

Поддержка проектов по преобразованию приоритетных отраслей экономики и социальной сферы на основе внедрения отечественных продуктов, сервисов и платформенных решений, созданных на базе «сквозных» цифровых технологий составит в 2019–2021 годах 11,7 млрд руб., 15,7 млрд руб. и 25,6 млрд руб. в год.

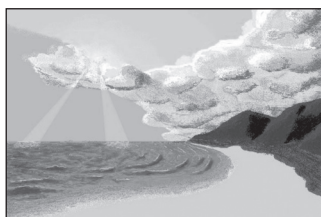
(По материалам CNews)

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» в первом полугодии 2019 года.

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

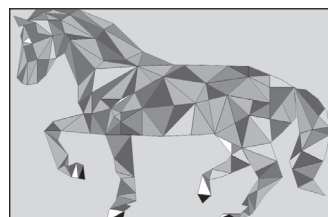
В оформлении обложки данного номера журнала «Информатика в школе» использованы работы следующих авторов — победителей конкурса цифровых изображений и фотографий:



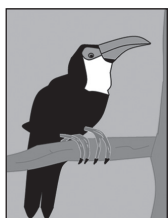
Анна Дроздова,
студентка Оренбургского
государственного колледжа



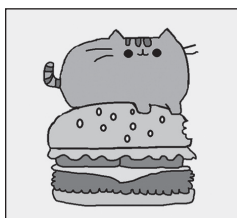
Алексей Шахаев,
студент Мамадышского
политехнического колледжа,
Республика Татарстан



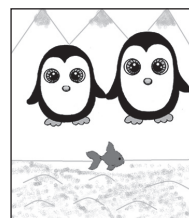
Константин Малков,
ученик Наро-Фоминской средней
общеобразовательной школы № 6
с углубленным изучением отдельных
предметов, Московская область



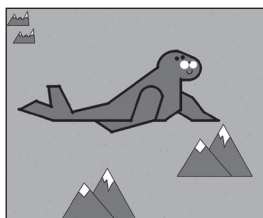
Наиль Нигматзянов,
студент Мамадышского
политехнического колледжа,
Республика Татарстан



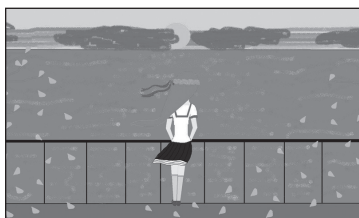
Эрика Байрамова,
ученица средней общеобразовательной
школы № 1699 Управления делами
Президента Российской Федерации, г. Москва



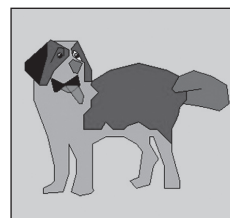
Ульяна Трескунова,
ученица Павловской гимназии,
Истринский район,
Московская область



Анна Яшина,
ученица Наро-Фоминской средней
общеобразовательной школы № 1,
Московская область



Дарья Климова,
ученица средней общеобразовательной школы
№ 2, г. Невельск, Сахалинская область



Даниил Казаков,
ученик Наро-Фоминской средней
общеобразовательной школы № 1,
Московская область



Максим Трегубов,
ученик средней общеобразовательной
школы № 8, г. Ноябрьск,
Ямало-Ненецкий автономный округ



Юлия Трухина,
ученица средней общеобразовательной школы
сельского поселения Поселок Молодежный
Комсомольского муниципального района Хабаровского края

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 1-е полугодие 2019 года

- 81407 — для индивидуальных подписчиков
- 81408 — для организаций

Периодичность выхода: 5 номеров в полугодие (в январе 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		 (индекс издания)			
Информатика в школе						Количество комплектов					
(наименование издания)											
На 2019 год по месяцам											
1	2	3	4	5	6	7	8	9	10	11	12
Куда											
(почтовый индекс)				(адрес)							
Кому _____											
----- Линия отреза -----											
ПВ			место			литер			ДОСТАВОЧНАЯ КАРТОЧКА		
									(индекс издания)		
На газету журнал						Информатика в школе					
(наименование издания)											
Стои- мость	подписки		руб.		Количество комплектов						
	каталож- ная		руб.								
	переадре- совки		руб.								
На 2019 год по месяцам											
1	2	3	4	5	6	7	8	9	10	11	12
Город											
село											
почтовый индекс			область								
код улицы			Район								
дом			улица								
корпус			квартира								
Фамилия И.О.											

Электронная подписка на журналы ИНФО

Журналы по методике
обучения информатике
и информатизации образования



- ✓ Доступ к журналам не дожидаясь печати типографии
- ✓ С любого устройства, подключенного к Интернет
- ✓ Возможность сохранить файл в формате PDF
- ✓ В два раза дешевле печатной подписки
- ✓ Скидки при оформлении подписки на комплект журналов
- ✓ Оплата на сайте издательства в Интернет-магазине

Информатика и образование

ИЗДАЕТСЯ С 1986 ГОДА • ОТ 64 СТРАНИЦ • ВЫХОДИТ 10 РАЗ В ГОД

Научно-методический журнал по методике преподавания информатики и информатизации образования

Информатика в школе

ИЗДАЕТСЯ С 2002 ГОДА • ОТ 64 СТРАНИЦ • ВЫХОДИТ 10 РАЗ В ГОД

Научно-практический журнал для учителей информатики, методистов, преподавателей вузов и колледжей

Подробную информацию о подписке на наши издания вы можете найти на сайте

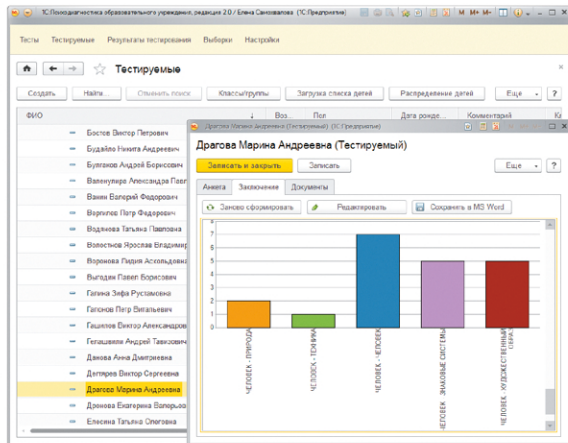
<http://infojournal.ru/subscribe/>



1С:ПСИХОДИАГНОСТИКА

Программно-методические комплексы линейки «1С:Психодиагностика» представляют собой инструментарий для проведения компьютерной психодиагностики детей и подростков, для сбора и консолидации результатов тестирования. Программы разработаны при поддержке группы ведущих психологов МГУ им. М.В. Ломоносова под общим руководством доктора психологических наук, профессора А.Н. Гусева. Программы линейки «1С:Психодиагностика»

- одобрены ФГАУ «Федеральный институт развития образования» в качестве программного обеспечения для использования психологами образовательных учреждений;
- включены в Единый реестр российских программ для электронных вычислительных машин и баз данных.



Функциональные возможности

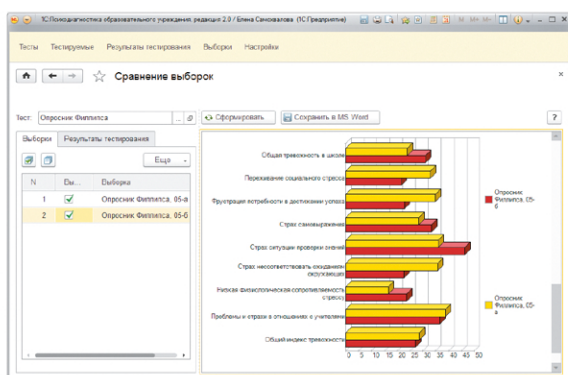
- Хранение информации о тестируемых, их родителях, учителях в единой базе данных.
- Хранение результатов тестирования.
- Ведение истории работы психолога с тестируемым.
- Удаленное и массовое тестирование при помощи проекторов. Поддерживаются батареи тестов.
- Ввод и обработка данных с бумажных бланков, сформированных в программе.
- Сравнение результатов тестирования отдельных тестируемых.
- Автоматический расчет результатов тестирования.
- Формирование выборок результатов тестирования: по классам (группам), полу, возрасту и т.д.

The screenshot shows the 'Тесты' (Tests) window with a table of test results. The table has columns for 'Наименование' (Name), 'Возраст от' (Age from), 'Возраст до' (Age to), 'Время тестирования' (Testing time), and 'Для младших групп' (For younger groups).

Наименование	Возраст от	Возраст до	Время тестирования	Для младших групп
Опросник жалоб ребенка	Адаптация в коллективе	5	14	15
Опросник Кателла. Подрост...	Общее	12	15	40
Опросник креативности Дики...	Креативность	7	13	15
Опросник Стигбегера - Хан...	Общее	15	99	15
Опросник темперамента Тома...	Общее	3	7	15
Опросник толерантности	Толерантность	15	99	15
Опросник Томаса	Общее	14	99	15
Опросник Флиссаса	Адаптация в коллективе	7	17	13
Опросник Шварца	Мотивация	11	99	15
Оценочник к самооценке	Адаптация в коллективе	3	7	5
Оценочник к чужому взросло...	Адаптация в коллективе	3	7	5
Оценка наркотизации	Адаптация в коллективе	13	99	20
ТБО	Общее	14	13	40
Польский	Исходный	4	7	20
Пословный	Мотивация	11	99	15
Проба на произвольность	Исходный	5	7	15
Проективная агрессия	Адаптация в коллективе	3	7	5
Психолого-педагогическая к...	Общее	3	13	5
Расшифровка кружков	Общее	5	9	30
Расшифровка	Психическая в школе	5	7	15
САН	Общее	7	13	5

Блоки методик

- Профориентация.
- Индивидуально-психологические особенности:
 - Оценка уровня тревожности,
 - Оценка уровня агрессии,
 - Исследование самооценки,
 - Исследование темперамента,
 - Исследование креативности,
 - Оценка познавательной сферы
 - Оценка ценностных ориентаций.
- Адаптация в коллективе.
- Детско-родительские отношения.
- Готовность к школе.



Преимущества использования

- Улучшение качества психологического сопровождения воспитательного процесса.
- Повышение производительности труда психологов.
- Соблюдение конфиденциальности психологической информации.
- Оценивание динамики психического развития детей.
- Формирование отчетов о проделанной работе.
- Снижение вероятности ошибок в результатах расчета психодиагностического исследования.
- Автоматизация процесса написания заключений.