

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное
учреждение высшего образования
«КРАСНОЯРСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ
УНИВЕРСИТЕТ им. В.П. Астафьева» (КГПУ им. В.П. Астафьева)

Институт математики, физики и информатики (ИМФИ)
Кафедра информатики и информационных технологий в образовании (ИИТО)

Юдина Юлия Ивановна

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**ИНДИВИДУАЛИЗАЦИЯ ДОПОЛНИТЕЛЬНОЙ ПОДГОТОВКИ
СТАРШЕКЛАССНИКОВ ПО ПРОГРАММИРОВАНИЮ**

Направление подготовки: 44.03.05 Педагогическое образование

Направленность (профиль) образовательной программы:
Математика и информатика

ДОПУСКАЮ К ЗАЩИТЕ

Зав. кафедрой ИИТО:
д-р пед. наук, профессор
Пак Н.И.

_____ (дата, подпись)

Руководитель:
канд. физ.-мат. наук, доцент каф.
ИИТО
Романов Д.В.

_____ (дата, подпись)

Обучающаяся:
Юдина Ю.И.

_____ (дата, подпись)

Дата защиты _____

Оценка _____

Красноярск 2023

Оглавление

Введение.....	3
Глава 1. Теоретические основы индивидуализации электронной образовательной среды для дополнительной подготовки старшеклассников.....	8
1.1. Общие аспекты индивидуализации обучения в электронной образовательной среде.....	8
1.2. Характеристика особенностей дополнительной подготовки старшеклассников по программированию.....	15
Выводы по первой главе.....	19
Глава 2. Разработка технологии индивидуализации обучения программированию при дополнительной подготовке старшеклассников..	21
2.1. Электронная среда обучения и ее наполнение.....	21
2.2. Методические рекомендации по наполнению и использованию среды..	33
2.3. Кейсы.....	35
Выводы по второй главе.....	39
Заключение.....	41
Библиографический список.....	43

Введение

Характерная черта нашей жизни – нарастание темпа изменений. Мы живем в мире, который совсем не похож на тот, в котором мы родились. И темп изменений продолжает нарастать. Чтобы получить 50 миллионов пользователей, радио потребовалось 38 лет, телевидению втрое меньше – 13 лет, Интернету еще в три раза меньше времени (4 года), а ChatGPT собрал первый миллион пользователей за 5 дней. Сегодняшним школьникам предстоит работать по профессиям, которых пока нет, использовать технологии, которые еще не созданы, решать задачи, о которых мы можем лишь догадываться [5]. Специалистам предстоит непрерывно учиться, оттачивая уникальные сочетания компетенций и поддерживая широкий кругозор.

В настоящее время, массовое школьное обучение ориентировано в основном на усредненного обучающегося. Индивидуальные особенности и дополнительные запросы обучающихся практически не учитываются, тем самым затрудняя индивидуализацию образования, учет мотивации, уровней знания и потребностей каждого обучающегося.

Чем полнее учитываются особенности каждого учащегося, тем выше эффективность школьного обучения. Требования жизни возрастают, поэтому увеличивается объем и усложняется содержание знаний, которые подлежат усвоению детьми в школе. Между тем, разнообразие задач и нагрузка приводит к тому, что многие учителя ориентируются в первую очередь на обязательные, базовые требования, что демотивирует более способных учеников, снижая их учебную активность [30].

На удовлетворение индивидуальных потребностей, обучающихся в интеллектуальном, нравственном и физическом совершенствовании ориентировано дополнительное образование, обеспечивающее адаптацию учащихся к жизни в обществе, профессиональную ориентацию, а также

выявление и поддержку детей, проявивших выдающиеся способности, или, напротив, отстающих [8].

Задача оптимизации обучения с учётом индивидуальных особенностей учеников в настоящее время исследуется авторами MOOK и проектировщиками обучающих онлайн-сред. Адаптивные учебники позволяют индивидуально настроить курсы под особенности и потребности каждого конкретного обучающегося. Разные авторы используют разные идеи, от машинного обучения до применения различных алгоритмов ИИ и целочисленной оптимизации [16].

До сих пор публикуется много работ и исследований в этой области, однако однозначного, практичного и доступного решения всё ещё не найдено. Заметная часть работ носит теоретический характер и не воплощена в продукте. Следует отметить цифровой репетитор по математике Plario - первая в России онлайн-система адаптивного обучения математике [40]. Plario создан на базе Томского государственного университета, однако доступ к нему до сих пор является платным и не общедоступным.

Многие авторы не учитывают эволюцию рынка труда. Анализ литературы и требований работодателей показал, что прогресс в области информатики и искусственного интеллекта трансформирует высокотехнологический сектор экономики. Наряду со старыми профессиями непрерывно появляются новые: профессия “веб-программист” 2001-го года превратилась в семейство из десятков переплетённых профессий: фронт и бэкэнд разработчики разного уровня, использующих десятки языков программирования; зачастую разработка ведётся на множестве фреймворков разных поколений. Для работы команд среднего уровня и выше уже нужны DevOps инженеры, тестировщики, UI/UX дизайнеры, программисты баз данных, верстальщики, и многие другие. Такая же картина и в ряде других, критически важных для инновационного потенциала страны, областях.

Машинное обучение, искусственный интеллект и большие данные на наших глазах уже превратились в промышленно устоявшиеся дисциплины, в рамках которых требуются ML-инженеры, аналитики, и множество других профессий. Появление генеративных ИИ на наших глазах порождает профессию prompt engineer.

Таким образом, содержание курсов и(или) их структура постоянно устаревают, что дополнительно затрудняет набор статистики при построении адаптации методами машинного обучения или ИИ на элементах курса.

Входной уровень обучающегося, между тем, не растёт так же быстро, и обусловлен психофизиологическими характеристиками ребёнка, который заканчивает начальную школу, и которому только предстоит подготовка к выбору будущей профессии и поступлению в ВУЗ. И, как правило, учебная программа большинства курсов пишется преподавателями и работниками министерств вручную, исходя из личного опыта, представлений и предпочтений авторского коллектива/школы.

Таким образом, мы получаем острое **противоречие** между быстро и непрерывно меняющимися требованиями к результатам обучения, к содержанию и структуре образовательной программы, требованиями профессиональных стандартов будущих профессий, огромным объёмом актуальной, постоянно пополняемой и непрерывно устаревающей информации, - и ручным характером проектирования образовательных курсов и программ обучения. Полная автоматизация упирается в недостаток данных, которых для алгоритмов ИИ и машинного обучения требуется очень много, и игнорирует самого педагога, активно задействованного в процессе обучения и непосредственно заинтересованного в результате.

Объект исследования работы - обучение программированию обучающегося средней и старшей школы.

Предмет исследования - технология построения индивидуальной образовательной траектории минимальной трудоемкости с учётом пожеланий обучающегося в отношении будущей профессии и его текущего уровня владения нужными компетенциями.

Проблема исследования заключается в информатизации процесса подбора оптимального индивидуального плана обучающим педагогом.

Цель работы – научно-методически исследовать доступные технологии адаптивности электронных сред обучения, обосновать и разработать программный продукт для помощи в проектировании детального индивидуального учебного плана минимальной трудоемкости с учётом персональных целей ученика и опыта педагога.

Гипотеза: деление на задачи с минимальным объемом вспомогательного аннотирования, в общий банк задач, и автоматизация процесса учёта зависимостей даст основу для моделирования образовательной траектории к выбранному образовательному результату в полуавтоматическом режиме, позволяя педагогу учитывать особенности, пожелания и возможности обучающегося, временные и прочие ограничения курса.

Для достижения цели были поставлены следующие **задачи**:

1. Определить теоретические основы подготовки старшеклассников в области программирования на внеурочных занятиях по информатике с использованием современного оборудования.

2. Охарактеризовать внеурочную деятельность по информатике в старшей школе как составляющую образовательного процесса.

3. Выделить и описать дидактические, технологические и программно-технические особенности учёта индивидуальных особенностей учеников.

4. Спроектировать и разработать среду проектирования

дополнительной программы подготовки старшеклассников в области программирования для использования на внеурочных занятиях по информатике.

5. Провести оценку разработанных средств и проанализировать ее результаты.

Область применения полученных результатов: в работе описана электронная среда и рекомендации по ее применению в старшей школе для обеспечения реализации дополнительной подготовки по программированию. Указанные средства направлены на формирование навыков программирования, что может быть использовано учителями информатики общеобразовательной школы.

Глава 1. Теоретические основы индивидуализации электронной образовательной среды для дополнительной подготовки старшеклассников

1.1. Общие аспекты индивидуализации обучения в электронной образовательной среде

Индивидуализация образования - это процесс адаптации образовательных программ и методов обучения к индивидуальным потребностям каждого обучающегося, учёта его уникальных способностей, интересов и возможностей. Это понятие становится все более актуальным с развитием новых технологий и возрастанием количества детей с разнородной подготовкой и потребностями, с увеличением количества слушателей MOOK курсов или пользователей электронных университетов.

Индивидуализация образования является одним из важнейших направлений в современном образовании. Она имеет множество преимуществ и адресует множество актуальных задач.

Постоянно изменяющаяся профессия “учитель” ставит перед обладателем данной профессии задачу и необходимость непрерывного профессионального развития, и возрастания роли в этом процессе самого педагога. Это может быть наиболее эффективно осуществлено посредством индивидуализации и самопроектирования образовательных траекторий [38].

Включение идеи индивидуализации и технологии самопроектирования в систему образования позволяет в большей степени учесть особенности и индивидуальный запрос каждого обучающегося.

Значимость проблемы индивидуализации обучения подчеркивается многими авторами (Е.А. Азарова, Т.В. Ахутина, Т.М. Ковалёва, Т.М. Марютина, Р.М. Петрунёва, Б.А. Сазонов и др.). Исследователи акцентируют внимание на основных проблемах организации обучения, направленных на

учет и развитие индивидуальных особенностей обучающихся, развитие познавательной самостоятельности обучающихся.

Индивидуализация обучения подразумевает переход на субъект-субъектные отношения, в процессе которых обучающийся имеет права и возможности формировать и реализовывать индивидуальную образовательную траекторию [14].

Суть индивидуализации образовательного процесса в широком контексте заключается в существовании возможности для каждого обучающегося формулировать собственные учебные образовательные цели и задачи, привносить личностный смысл в свое образование, а также определять и видеть собственные учебные образовательные перспективы [17].

Важной вехой индивидуализации современного обучения является совместная деятельность учителя и ученика, при которой выбор способов, приемов и темпа учебной деятельности обусловлен уровнем потенциальных возможностей обучающегося.

Несмотря на относительно недавнее представление вопроса об индивидуализации образовательного процесса в обществе, данная научная проблема имеет глубокие философские корни и древнюю предысторию.

Возникновение и развитие понятия “индивидуализация” в русле философского знания уже являлось предметом исследования. Согласно различным источникам, можно выделить следующие стадии развития этого понятия, представленные на рисунке. 1.1 [14].



Рисунок 1.1 – Эволюция понятия «индивидуализация» в русле философского знания

Преимущества индивидуализации образования:

- Адаптация программы обучения к потребностям и возможностям обучающегося повышает его интерес к обучению, и мотивацию для достижения успеха.

- Ориентация на увлечения, интересы, способности, склонности и темп обучения каждого обучающегося. Это позволяет учителю оптимально организовать образовательный процесс, чтобы каждый обучающийся мог реализовать свой потенциал.

- Различные методы обучения и помощи, которые позволяют учителю эффективно и привлекательно обучать каждого обучающегося.

- Создание условий для свободного и самостоятельного выбора учебных траекторий, индивидуальных заданий и проектов, которые максимально соответствуют интересам и целям каждого обучающегося.

- Улучшение качества обучения и повышение успеваемости каждого обучающегося за счет учета его потребностей и возможностей, поддержки и дополнительных ресурсов.

Один из практических методов индивидуализации образования – это диагностика потребностей учащихся с помощью тестов и опросов, интернет-анкет, которые позволяют определить интересы, способности, характер, проблемы и задачи различных категорий учеников, что затем

используется для адаптации педагогических методик к потребностям каждого обучающегося.

Инновационные процессы, происходящие в обществе, ставят новые задачи, стимулируя появление передовых, современных, форм и способов повышения качества образования с целью развития самостоятельности, предприимчивости, инициативы всех субъектов образовательной деятельности. Информатизация образовательного процесса, актуализация непрерывного, открытого образования, составляющего фундамент современного информационного общества, послужили резким толчком для внедрения электронного обучения [9].

Согласно ФГОС нового поколения, преподавание любых дисциплин должно вестись с использованием электронных ресурсов сети Интернет и информационных технологий, соответственно учителя должны существенно повышать ИКТ-компетенцию, так как это является требованием времени [21].

Применение технологий электронного обучения (ЭО) - дистанционного, гибридного (смешанного) или мобильного - при реализации образовательных программ становится законодательно обозначенной нормой для каждого образовательного учреждения. Это требует создания условий (технических, нормативных, педагогических, методических, пространственных, др.), обеспечивающих эффективное существование электронной образовательной среды (ЭОС) [32].

Электронная образовательная среда (ЭОС) - это совокупность средств, технологий и ресурсов, которые используются для организации образовательного процесса, предоставления обучения, оценки знаний и управления учебным процессом через интернет.

Индивидуализация образования заключается в создании уникальной образовательной программы для каждого обучающегося и предоставления

ресурсов, которые наилучшим образом подходят для этого обучающегося. В электронной образовательной среде (ЭОС) индивидуализация означает возможность создания гибкого образовательного пространства, которое учитывает различные особенности обучающихся и подстраивается под их индивидуальные потребности.

Одним из важных аспектов ЭОС является использование её возможностей для индивидуализация учебного плана, который можно адаптировать под потребности конкретных обучающихся. Возможно создание персонализированных путей обучения, которые могут определяться различными параметрами (например, уровнем подготовки, интересами, особенностями личности и т.д.). Анализ данных о прошлом успехе обучающегося, результаты тестов и заданий также могут послужить учителям опорой при создании персональных планов обучения.

Еще одним аспектом индивидуализации в ЭОС является возможность предоставления гибких методов оценки знаний и умений, которые учитывают индивидуальные потребности и способности обучающихся. Например, использование различных типов заданий (например, тесты, задачи, проекты, дискуссии, и т.д.) и методов оценки (например, письменные ответы, устные ответы, и т.д.) могут помочь учителям лучше оценить индивидуальные потребности обучающихся.

Возможности индивидуализации в ЭОС проявляются в использовании гибких методов обратной связи, которые могут помочь обучающимся понять свои индивидуальные сильные и слабые стороны и улучшить свои результаты. Например, эффективным инструментом обратной связи являются виртуальные обучающие модули и упражнения, которые могут анализировать и оценивать ответы учеников и давать им рекомендации по их развитию.

Индивидуализация в ЭОС также включает в себя использование различных методов обучения, которые оптимальны для каждого

обучающегося. Например, это могут быть интерактивные игры, видеоуроки, виртуальные лаборатории, презентации и т.д. Важно подбирать и комбинировать эти методы, чтобы наилучшим образом соответствовать индивидуальным потребностям и интересам каждого обучающегося.

В целом, индивидуализация ЭОС представляет собой процесс создания гибкой и персонализированной образовательной среды для каждого обучающегося. Она позволяет создать условия для более эффективного обучения и развития личности каждого обучающегося. Важно, чтобы реализация индивидуализации в ЭОС была осуществлена с систематическим подходом и учетом всех необходимых аспектов.

Электронная образовательная среда, или ЭОС, играет важную роль в индивидуализации протекания образовательного процесса.

Во-первых, ЭОС позволяет организовать доступ к образовательным ресурсам в любое время и из любой точки мира. Это значит, что обучающиеся могут учиться в своем темпе и в соответствии с индивидуальным графиком занятий. Кроме того, ЭОС может быть настроена на предоставление обучающимся персонализированных заданий и материалов.

Во-вторых, ЭОС содержит инструменты, позволяющие автоматизировать процессы управления знаниями, учета успеваемости и общения между преподавателями и обучающимися. Таким образом, данные могут быть использованы для анализа прогресса каждого учащегося и разработки персональных образовательных планов.

В-третьих, снижение требований к уровню технической подготовки педагога позволяет последнему включать в работу различные формы квизов, помогая удерживать внимание слушателей и меняя характер взаимодействия с аудиторией.

Технологии индивидуализации образования – это методы и инструменты, позволяющие сделать образование более персонализированным и адаптированным к индивидуальным потребностям обучающегося.

Вот несколько технологий, которые могут помочь в индивидуализации образования:

- Адаптированные программы обучения – программы, которые адаптированы к возрасту, уровню знаний и способностям обучающегося.

- Разноуровневые группы – деление по группам, в которых собираются обучающиеся одинакового уровня знаний.

- Оценки на основе компетенций – оценки, которые сфокусированы на уровне усвоения конкретной компетенции, а не на общей оценке за дисциплину.

- Смешанное обучение (blended learning) – использование платформ для онлайн-обучения для достижения определенных учебных целей в удобное для обучающегося время и удобной форме.

- Интерактивные группы – группы, в которых среди обучающихся проводятся дискуссии и обсуждения, при этом учителю предоставляется роль модератора, а ЭОС служит площадкой и предоставляет средства общения.

- Разнообразные формы контроля знаний – включение в процесс обучения различных тестов, кейсов, домашних заданий и иных форм контроля, обеспечивающих учеников дополнительной информацией о собственном уровне знаний.

- Персональные учебные планы – индивидуальные программы обучения, разработанные учителем или специальным педагогом, учитывающие индивидуальные потребности и цели обучающегося.

Каждая из этих технологий может быть эффективным средством индивидуализации образования. Конечно, наилучший результат может быть

достигнут в том случае, если использовать не одну, а несколько технологий одновременно, а также при условии грамотного и профессионального применения технологий учителем.

Таким образом, можно сделать вывод о том, что электронная образовательная среда играет важную роль в поддержке индивидуализации образовательного процесса, предоставляя обучающимся инструменты для управления своим образованием, а также предоставляя возможности для преподавателей для разработки образовательных программ, ориентированных на индивидуальные потребности и цели каждого обучающегося.

1.2. Характеристика особенностей дополнительной подготовки старшекласников по программированию

Цель создания системы дополнительного образования детей - раннее обнаружение склонностей, проблем или талантов ребенка, формирование его интересов и помощь в профессиональном самоопределении.

Ценность дополнительного образования в том, что оно дает обучающимся почувствовать важность обучения в школе, побуждает их более ответственно относиться к занятиям и способствует реализации тех знаний, которые они получают на уроках.

Система дополнительного образования в школе побуждает учеников развиваться самостоятельно. Занятость ребенка вне школы формирует у него дисциплинированность, организованность и самоконтроль.

Для организации системы дополнительного образования в основном используется база учебного учреждения: кабинеты, литература, инвентарь. Не все заведения обладают достаточными ресурсами для углубленного изучения предметов и внеклассных занятий [15].

Таким образом, востребована помощь и поддержка учителям в

осуществлении ими своей профессиональной деятельности при ведении программ дополнительного образования.

Федеральными государственными образовательными стандартами основного общего и среднего (полного) общего образования продиктована необходимость изучения алгоритмизации и программирования в рамках предметной области «Математика и информатика». Такая необходимость оправдана прежде всего тем, что в век глобальной информатизации общества и стремительного совершенствования вычислительных систем, умение формализовывать алгоритмы является одной из важнейших компетенций современного человека. Изучение основ алгоритмизации и программирования в школьном курсе информатики на разных этапах обучения позволяет сформировать у обучающегося актуальные профессионально направленные умения и навыки, а также способствовать развитию личностных особенностей, к числу которых можно отнести, в частности, алгоритмическое мышление [32].

Истоки программирования связаны с развитием вычислительной техники и информатики. С самого начала появления первых электронных вычислительных машин их создание, программирование и использование ограничивалось небольшим числом специалистов и университетами, которые обладали высоким уровнем знаний в области математики и информатики.

В начале 1940-х годов первые вычислительные машины были построены в США военно-промышленным комплексом для военных. Эти машины были огромными, громоздкими и мало использовались. Однако они стали отправной точкой для будущих разработок в области электронных вычислительных машин.

К началу 1950-х годов разработка электронных компьютеров и программирование стали предметом академического и технического

интереса. В 1951 году Джон Маккарти из Массачусетского технологического института создал первый язык программирования - Short Code. Он был очень примитивным и имел низкую скорость.

В последующие годы появилось еще очень много языков, а Java, созданный в 1995 году Джеймсом Гослингом из Sun Microsystems, стал первым языком программирования, предназначенным для Интернета.

Так, программирование начало бурно развиваться, охватывая всё больше людей и все большие страны.

В настоящее время программирование является одной из самых популярных и востребованных областей в IT-индустрии, поэтому всё больше и больше школьников выражают желание изучать программирование уже на старшей ступени обучения в школе. Однако, общеобразовательная программа в школах не всегда может обеспечить полноценное изучение программирования, в связи с чем в школах проводят дополнительную подготовку старшеклассников по программированию.

Во-первых, дополнительная подготовка по программированию в школьных кружках или специализированных секциях отличается от обычной школьной программы тем, что она ориентирована на самостоятельное изучение материала обучающимся. В курсе обычной школы обучающиеся такого объема самостоятельной работы не получают, поэтому в подготовке по программированию приходится изучать материал вне школы.

Во-вторых, дополнительная подготовка по программированию ориентирована на практическую работу. Это означает, что на занятиях много времени уделяется написанию кода, решению задач и проектных работ. Большое внимание уделяется также использованию современных технологий, которые позволяют быстро и удобно создавать программные продукты.

В-третьих, подготовка по программированию носит открытый и гибкий

формат работы. Программа может быть комплексной и нацелена на изучение разных языков программирования, или же более узкоспециализированной, ориентированной на конкретную технологию или направление в IT-сфере. Программа также может меняться в зависимости от интересов и потребностей обучающихся, что позволяет индивидуализировать обучение.

Обучение программированию на среднем и высоком уровне сложности имеет и специфическую сложность - при изучении программирования крайне тяжело удерживать обучающегося в потоке, см. рисунок 1.2, поскольку разброс уровня сложности задач один из самых широких в сфере умственного труда. Программист собирает алгоритм, последовательно усложняя и комбинируя примитивы языка, и верхний уровень сложности ограничен только умственными способностями ученика. Тем не менее, как практика, программирование выработало ряд эффективных парадигм, позволяющих создавать промежуточные слои абстракции при конструировании программного обеспечения, которые крайне не очевидны для новичка. Именно поэтому в репертуаре опытного педагога собраны сотни примеров и задач на понимание, и он применяет их крайне осторожно, с поддержкой обратной связи с обучающимся, постоянной рефлексией учебного процесса в течение всего обучения. Непривычный характер трудностей, по сравнению с однородной сложностью школьных задач, часто демотивирует обучающихся, которые путают непривычность задач с отсутствием способностей (таланта) к предмету.

Поскольку способности и стиль программирования быстро становятся уникальными, набор задач для решения и разбора, требуемый для удержания ученика в потоке, также должен быть разнообразным. В пример можно привести содержание ресурса <https://acmp.ru/>, хороших учебников по программированию, содержание профиля ЕГЭ (сайт К.В. Полякова).

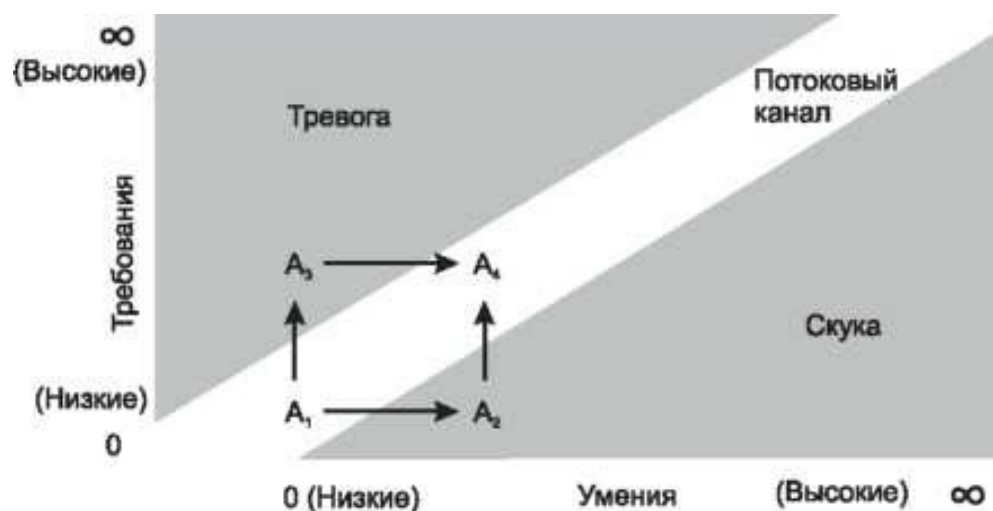


Рисунок 1.2 - Поток

Для удержания состояния потока требуются [43]:

Ясные цели: чем яснее цели, тем проще на них сконцентрировать внимание, и тем легче достичь состояния потока.

Отсутствие отвлекающих факторов: наличие отвлекающих факторов приводит к рассеиванию внимания. Для полной концентрации внимания необходимо задействовать как сознание, так и тело (это к вопросу важности безупречного управления в игре).

Мгновенная обратная связь: запаздывание обратной связи выбрасывает из потока, в то время как мгновенная обратная связь помогает оставаться в состоянии потока.

Непрерывный вызов: люди (как мы выяснили в предыдущей главе, в основном мужчины) любят вызов.

Выводы по первой главе

В данной главе были определены теоретические основы индивидуализации электронной образовательной среды для дополнительной подготовки старшеклассников. А также рассмотрены общие аспекты индивидуализации электронной образовательной среды и особенности

дополнительной подготовки старшеклассников по программированию.

В настоящее время важной вехой является индивидуализация, ориентиры на запросы и потребности конкретного человека. Это отражается и на образовании. Чтобы обучение прошло наиболее полно и эффективно, необходимо исходить из особенностей, целей и задач, которые ставит перед собой конкретный обучающийся.

Образование как и весь мир не стоит на месте и согласно ФГОС нового поколения преподавание любых дисциплин должно вестись с использованием электронных ресурсов сети Интернет и информационных технологий. Поэтому немаловажным является использование электронных образовательных сред. А именно индивидуализированных электронных образовательных сред. Индивидуализация означает возможность создания гибкого образовательного пространства, которое учитывает различные особенности обучающихся и подстраивается под их индивидуальные потребности.

Умение программировать, программное мышление - те навыки, которые необходимы практически каждому современному человеку. Однако, это достаточно сложная область, в которой много особенностей и нюансов. Программирование изучается от простого к сложному, но нет гарантий, что на каком-то из этапов не образуется трещина, по достижению определенного уровня программирования эти трещины становятся все больше и через них уже нельзя перепрыгнуть.

Таким образом, для современного учителя стоит задача учесть индивидуальные запросы обучающихся при том минимальном времени, выделенном на уроках и дополнительных занятиях.

Глава 2. Разработка технологии индивидуализации обучения программированию при дополнительной подготовке старшеклассников

2.1. Электронная среда обучения и ее наполнение

Обучение программированию - сложный и многогранный процесс, с которым справляются далеко не все учителя. Из-за этого у многих обучающихся формируются лишь минимальные навыки программирования, или не формируются вовсе.

С течением школьных лет все равно остаются некоторые пробелы в знаниях, даже иногда самых базовых, поэтому для решения более сложных задач необходимо вначале разобраться с решениями тех простых задач, которые постепенно подводят ученика к более сложным и “комом” нарастают на последующие задачи.

Особенно острая необходимость в закрывании пробелов возникает у обучающихся в 9 - 11 классах при подготовке к ОГЭ и ЕГЭ.

Основной государственный экзамен — итоговый экзамен за курс основного общего образования в России. Служит для контроля знаний, полученных учащимися за 9 лет, а также для приёма в учреждения среднего профессионального образования.

Единый государственный экзамен — централизованно проводимый в Российской Федерации экзамен в средних учебных заведениях — школах, лицеях и гимназиях, форма проведения ГИА по образовательным программам среднего общего образования. Служит одновременно выпускным экзаменом из школы и вступительным экзаменом в вузы.

При подготовке к ОГЭ и ЕГЭ по информатике важно учитывать, что задания не располагаются по нумерации от простых к сложным. Поэтому для многих обучающихся и учителей это оказывается проблемой, которую приходится решать вручную. Более того, иногда нужно затронуть те темы,

которые по отдельности не входят в структуру заданий ОГЭ/ЕГЭ.

Данная проблема возникает и при подготовке обучающихся к различным олимпиадам и конкурсам.

В помощь учителям была **разработана и усовершенствована электронная среда [6]**, которая подберет нужный набор задач для того, чтобы беспрепятственно решить задачу определенного уровня. При этом среда не только подберет нужные задачи, но и выстроит их в правильном направлении от простого к сложному. Данная среда универсальна и может совмещать в себе задачи не только различных уровней сложности, но и одновременно разных связанных областей знания - информатики, математики, физики.

Программный код среды написан на языке Python. Python — высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ.

Выделим требования, предъявляемые к электронной среде:

- допустимость разбиение материала на смысловые части, чтобы можно было его выдавать тематическими блоками, раздроблено;
- допустимость дифференциации заданий по уровню сложности для того, чтобы курс смогли пройти обучающиеся с разными уровнями знаний;
- допустимость создание индивидуальной траектории прохождения курса для некоторых обучающихся или малых групп;
- простота и понятность интерфейса среды для учителя, поскольку не все обладают достаточной компьютерной грамотностью;
- возможность непрерывного добавления в электронную среду нового контента.

Ключевые проектировочные решения, выбранные при создании ЭОС, следующие:

1. В качестве контрольных точек и единиц контента выбраны единичные задачи, набираемые из содержательной части курсов.
2. Каждая единица аннотируется метками (тегами, tags). В качестве метки преподаватель выбирает ключевые компетенции, которыми надо владеть для решения задачи.

Если единица контента - проект, то выбираются более высокоуровневые компетенции.

Пример трёх аннотированных единиц показан на рисунке 1.3.

3. Все единицы контента находятся в единой базе.
4. Для проектирования образовательного маршрута достаточно выделить набор задач финального уровня сложности. Это могут быть задачи профиля или базы ОГЭ и ЕГЭ, олимпиадные задачи, проект нужной специализации (проект блога или веб-портала, телеграм-бота, тестирующей среды).
5. После выбора, сделанного в п. 4, ЭОС строит граф освоения дисциплины, в котором гарантируется минимальное изменение набора тегов при движении по любому ребру. Таким образом, внизу оказываются самые простые задачи, и при движении вверх ученик учит на каждом шаге минимальный объём нового материала, опираясь на известный материал как на базу. Пример графа дан на рисунке 1.4.
6. Добавляя в набор п.4 дополнительные образовательные результаты, можно визуализировать набор дополнительных единиц контента, обязательных для освоения, и правильный порядок их освоения.

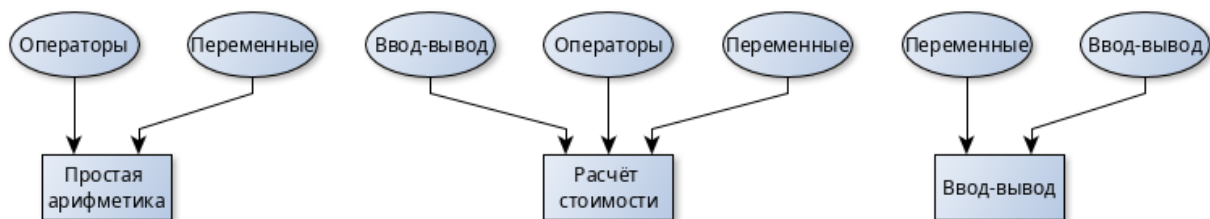


Рисунок 1.3 - Пример нескольких аннотированных единиц контента ЭОС. Прямоугольники - единицы контента, овалы - метки.

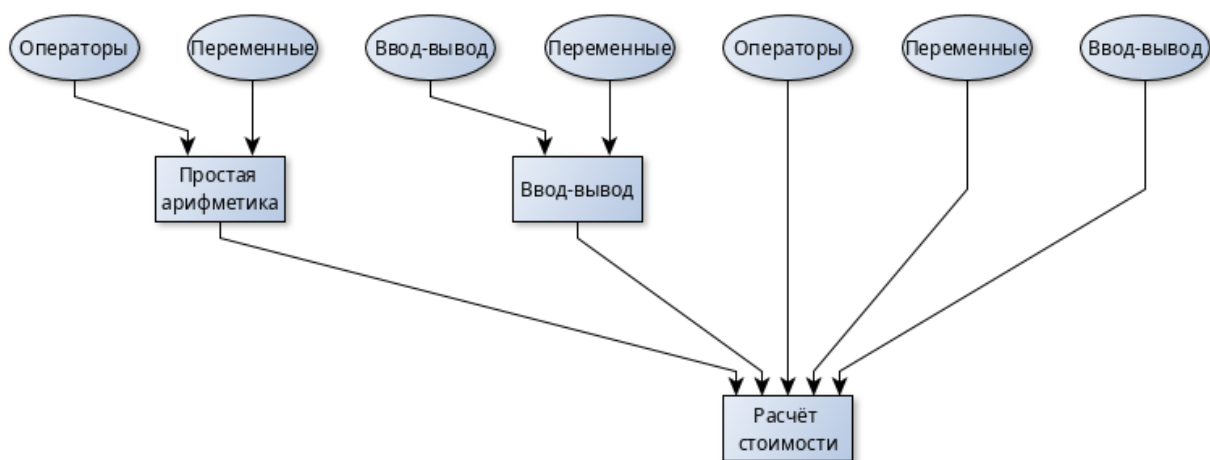


Рисунок 1.4 - Карта курса. Более простые единицы контента вверху графа, при переходе по графу зависимости количество меток растёт.

Подобные идеи разрабатываются в литературе. Ментальные схемы Н.И. Пака и Е. В. Асауленко [27], микрокурсы [25], перевёрнутые учебники и учебники-трансформеры Д.А. Бархатовой. В отличие от перечисленных работ, составление схемы - крайне когнитивно сложный и спорный этап - делается автоматически, и педагог использует результат как опору.

Визуализированный граф позволяет ясно видеть возможные трудности, и служит опорой, поверх которой строится содержание всего курса, включая методику, дидактические материалы, и т.д. Граф позволяет оценить разрывы,

если они есть, убрать дублирование материала. Добавление и автоматизация работы с единицами контента (интервальное повторение), построение индивидуального маршрута - в любом случае это требует построения такой базы контента как фундамента. Ненужные единицы контента убираются автоматически. Возможно пополнение базы новыми единицами контента в реальном времени, без остановки. При необходимости, можно пересчитать граф.

Для создания среды был выбран фреймворк Flask. Flask – это маленький фреймворк, который часто называют «микрофреймворк». После непродолжительного знакомства с ним вы сможете читать и понимать код, написанный на нём. Но при этом он обладает всеми функциями, которые необходимы для создания веб-приложения. Flask имеет монолитное ядро, реализующее основные службы, а всё остальное поддерживается посредством расширений. Поскольку вы можете выбирать только необходимые пакеты, в результате получается достаточно ограниченный комплект программных средств, которых достаточно для реализации ваших идей. Фреймворк Flask довольно популярный, понятный в использовании. На нём возможно написать хороший код, причём своими силами, не обращаясь к специалистам. Если говорить о недостатках, то Flask не имеет встроенной поддержки доступа к базам данных, проверки веб-форм, аутентификации пользователей или других высокоуровневых задач. Существует и множество иных ключевых служб, необходимых большинству веб-приложений и доступных в виде расширений, интегрируемых с основными пакетами. Для корректного отображения содержимого страниц были выбраны HTML 5 и Bootstrap 4 - библиотека шаблонов CSS.

Программный продукт не требует хостинга и помощи администратора

образовательного учреждения. В минимальной версии, для того, чтобы использовать электронную образовательную среду, достаточно распаковать архив и запустить скрипт. После запуска открывается окно консоли, в котором виден процесс установки всех необходимых библиотек, баз данных и файлов, как показано на рисунке 2.1.

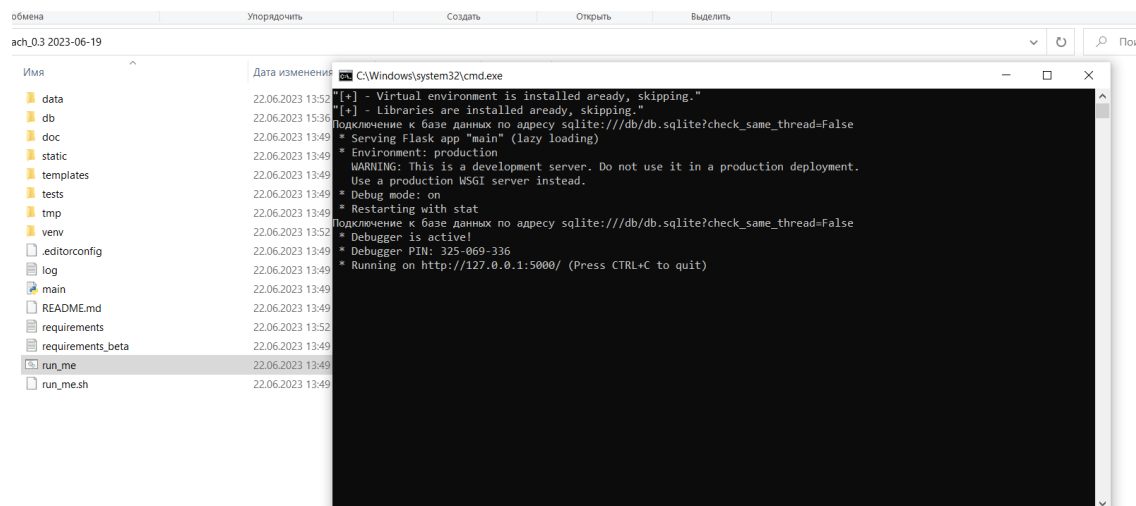


Рисунок 2.1 - Запуск среды

Далее мы видим, что среда готова к запуску по адресу, указанному на рисунке 2.2.

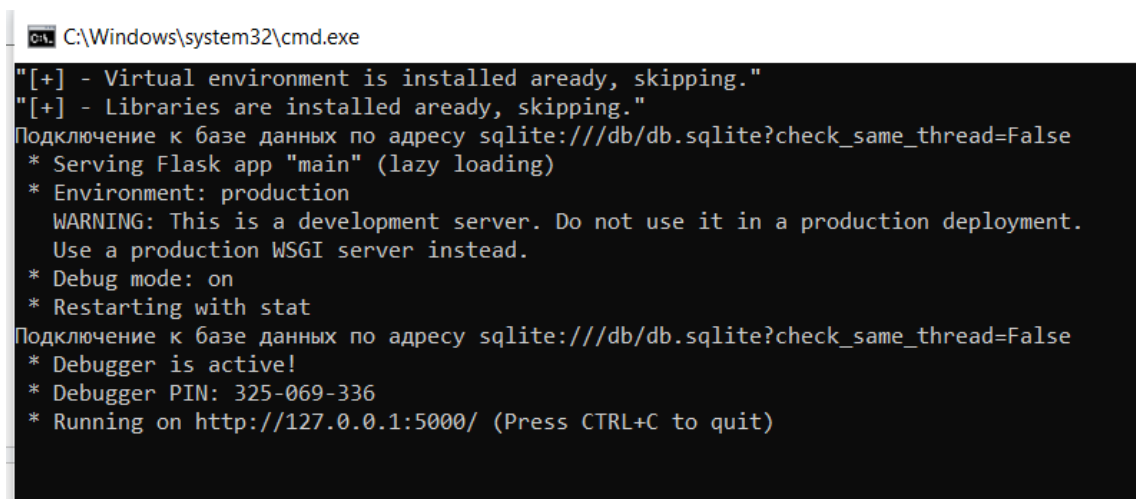


Рисунок 2.2 - Адрес для запуска среды

После необходимо ввести указанный адрес в адресную строку браузера и откроется окно главной страницы электронной образовательной среды, как

на рисунке 2.3.

ID	Заголовок	Метки	
1	1. Анализ информационных моделей	Цикл Массив Граф	Edit
2	2. Построение таблиц истинности логических выражений	Логические операторы	Edit
3	3. Поиск информации в реляционных базах данных	Цикл Ветвление Массив Функция Ввод/ Вывод	Edit
4	4. Кодирование и декодирование информации	Кодирование	Edit
5	ппп	Цикл	Edit
6	5. Анализ и построение алгоритмов для исполнителей (двоичное)	Двоичная запись числа	Edit
7	5. Анализ и построение алгоритмов для исполнителей (десятичное)	Ветвление	Edit
8	6. Определение результатов работы простейших алгоритмов	Цикл Ветвление	Edit
9	7. Кодирование и декодирование информации. Передача информации (передача звуковых файлов)		Edit
10	7. Кодирование и декодирование информации. Передача информации (Хранение звуковых файлов)		Edit
11	7. Кодирование и декодирование информации. Передача информации (Хранение изображений)		Edit
12	8. Перебор слов и системы счисления	Цикл Ветвление	

[Добавить новую задачу](#)

Активация Windows
Чтобы активировать Windows, перейдите в "Параметры".

Рисунок 2.3 - Главная страница

Здесь мы видим кнопку “Добавить новую задачу”, подсвеченную синим цветом гиперссылку “Граф”. Если мы нажмем на кнопку, то перед нами всплывет окно, как на рисунке 2.4. Если же пройдем по гиперссылке - окажемся на странице, которая показывает наш граф курса, который изображен на рисунке 2.5.

Ввод задачи:

Название

Условие задачи

Н В I S | — 66 | ☰ ☷ ☑ ☒ ☓ ☔ | 🗑 📄 🔗 | </> CB | Scroll 🔵

Пожалуйста, введите условие задачи

Markdown WYSIWYG

Цикл Ветвление Массив Функция Ввод/Вывод Граф Логические операторы Кодирование

Двоичная запись числа Таблицы Строки Сортировка Рекурсия Аргумент

Добавить

Отмена

Ввод меток:

Рисунок 2.4 - Ввод задач и меток

Все задачи
Select layout: cose-bilkent Randomize [Start] [Stop]

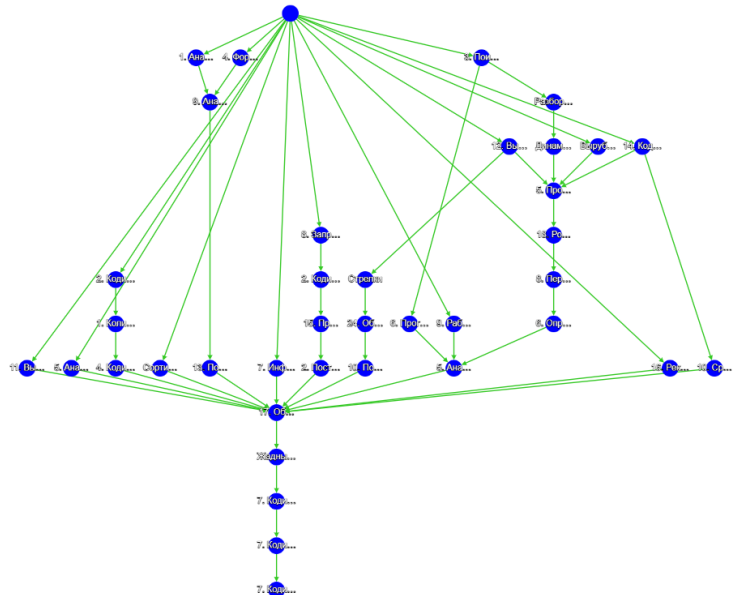


Рисунок 2.5 - Граф

Можно не только добавлять новые задачи, но и новые метки, как

показано на рисунке 2.6 и рисунке 2.7.

Добавляя задачу, мы можем теперь добавить не только текст, но также формулы, изображения, гиперссылки и т.п. См. рисунок 2.9.

Цикл Ветвление Массив Функция Ввод/Вывод Граф Логические операторы Кодирование
 Двоичная запись числа Таблицы Строки Сортировка Рекурсия Аргумент

Добавление новой метки

Метка — математическое/физическое понятие, термин, класс значений, вроде "путь", "скорость", "угол", "радиус", "треугольник", "линейное уравнение"

Название: "путь", "угол", "радиус" (без кавычек)

Сокращение: s, θ, r, лин.уравнение

Введите здесь короткое описание сути понятия или переменной

Рисунок 2.6 - Панель ввода новых меток

Цикл Ветвление Массив Функция Ввод/Вывод Граф Логические операторы Кодирование
 Двоичная запись числа Таблицы Строки Сортировка Рекурсия Аргумент

Добавление новой метки

Метка — математическое/физическое понятие, термин, класс значений, вроде "путь", "скорость", "угол", "радиус", "треугольник", "линейное уравнение"

Двумерный массив

Двумерный массив

Двумерный массив — это массив одномерных массивов (строк), последовательно расположенных в памяти.

Рисунок 2.7 - Добавление новой метки

После добавления метки можно отследить, что она появилась в перечне меток выше, и теперь ее также можно использовать. См. рисунок 2.8.

Все поля данных форм должны быть обязательно заполнены, если этого не сделать, программа сделает замечание.

- Цикл
 - Ветвление
 - Массив
 - Функция
 - Ввод/Вывод
 - Граф
 - Логические операторы
 - Кодирование
 - Двоичная запись числа
 - Таблицы
 - Строки
 - Сортировка
 - Рекурсия
 - Аргумент
 - Двумерный массив
-

Рисунок 2.8 - Панель меток

Мы видим эти метки на главной странице справа от названия задачи. Также, нажав 1 щелчком мыши на название задачи, можно увидеть ее текст. См. рисунок 2.10 и рисунок 2.11.


Если добавить метку, которая уже встречалась, программа также выдаст соответствующую ошибку, как на рисунке 2.12.

Название

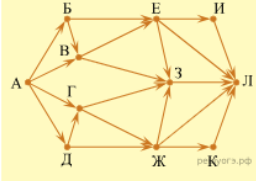
9. Анализирование информации, представленной в виде схем

Условие задачи

Н В I S | — 66 | ☰ ☷ ☑ ☒ ☓ | 🗃 🖼 🔗 | </> CB | Scroll 🔴

На рисунке — схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, К и Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город Л?  (https://inf-oge.sdamgia.ru/get_file?id=47383)

На рисунке — схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, К и Л. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город Л?



Markdown WYSIWYG

Цикл
 Ветвление
 Массив
 Функция
 Ввод/Вывод
 Граф
 Логические операторы
 Кодирование
 Двоичная запись числа
 Таблицы
 Строки
 Сортировка
 Рекурсия
 Аргумент
 Двумерный массив
 Кодирование символов
 Количество информации
 IP-адрес
 Системы счисления

Рисунок 2.9 - Добавление задачи с изображением

Задачи Граф		search	🔍
ID	Заголовок	Метки	
1	1. Анализ информационных моделей	Цикл	Массив Граф Edit
2	2. Построение таблиц истинности логических выражений	Логические операторы	Edit
3	3. Поиск информации в реляционных базах данных	Цикл Ветвление Массив Функция Ввод/Вывод	Edit

Рисунок 2.10 - Метки для задач

ID	Заголовок	Метки	
1	1. Анализ информационных моделей	Цикл Массив Граф	Edit
2	2. Построение таблиц истинности логических выражений	Логические операторы	Edit
3	3. Поиск информации в реляционных базах данных	Цикл Ветвление Массив Функция Ввод/ Вывод	Edit

На вход алгоритма подаётся натуральное число N. Алгоритм строит по нему новое число R следующим образом.

- 1) Строится двоичная запись числа N.
- 2) К этой записи дописываются справа ещё два разряда по следующему правилу:
 - а) складываются все цифры двоичной записи, и остаток от деления суммы на 2 дописывается в конец числа (справа). Например, запись 11100 преобразуется в запись 111001;

Рисунок 2.11 - Текст задачи

sqlalchemy.exc.IntegrityError

```
sqlalchemy.exc.IntegrityError: (sqlite3.IntegrityError) UNIQUE constraint failed: tags.nick  
[SQL: INSERT INTO tags (title, nick, annotation) VALUES (?, ?, ?)]  
[parameters: ('Цикл', 'Цикл', 'Цикл')]  
(Background on this error at: http://sqlalche.me/e/gkpkj)
```

Traceback (most recent call last)

```
File "C:\Users\Юлия\Desktop\Karach_0.3 2023-06-19\venv\Lib\site-packages\sqlalchemy\engine\base.py", line 1247, in _execute_context  
    self.dialect.do_execute(  
    ~~~~~
```

```
File "C:\Users\Юлия\Desktop\Karach_0.3 2023-06-19\venv\Lib\site-packages\sqlalchemy\engine\default.py", line 588, in do_execute  
    cursor.execute(statement, parameters)
```

The above exception was the direct cause of the following exception:

```
File "C:\Users\Юлия\Desktop\Karach_0.3 2023-06-19\venv\Lib\site-packages\flask\app.py", line 2463, in __call__  
    return self.wsgi_app(environ, start_response)
```

```
File "C:\Users\Юлия\Desktop\Karach_0.3 2023-06-19\venv\Lib\site-packages\flask\app.py", line 2449, in wsgi_app  
    response = self.handle_exception(e)
```

```
File "C:\Users\Юлия\Desktop\Karach_0.3 2023-06-19\venv\Lib\site-packages\flask\app.py", line 1866, in handle_exception  
    reraise(exc_type, exc_value, tb)
```

```
File "C:\Users\Юлия\Desktop\Karach_0.3 2023-06-19\venv\Lib\site-packages\flask\compat.py", line 39, in reraise  
    raise value
```

```
File "C:\Users\Юлия\Desktop\Karach_0.3 2023-06-19\venv\Lib\site-packages\flask\app.py", line 2446, in wsgi_app  
    response = self.full_dispatch_request()
```

Рисунок 2.12 - Ошибка повторного ввода метки

У графа добавился новый функционал: теперь его можно представлять в различных формах, а также двойным щелчком мыши полностью видеть название задачи. См. рисунок 2.13 и рисунок 2.14.

Все задачи
Select layout: **klay** Randomize **Start** **Stop**

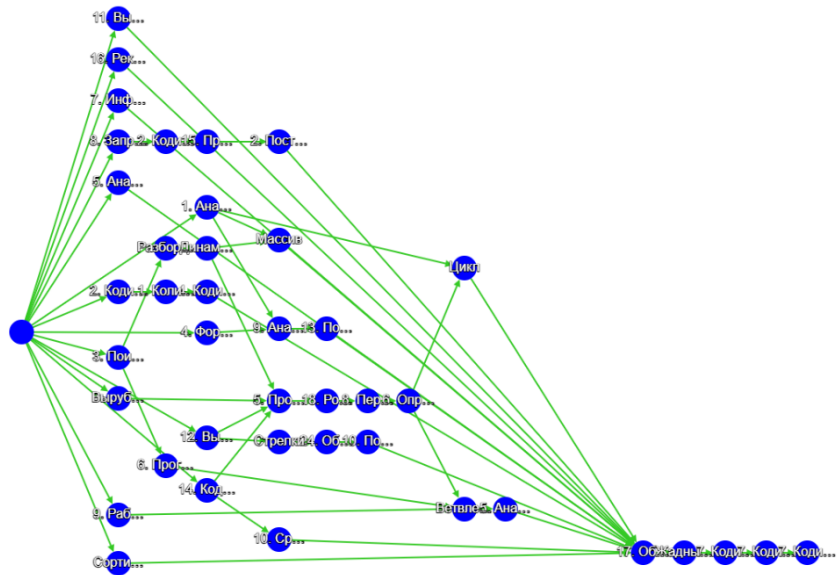


Рисунок 2.13 - Граф вида “klay”

Все задачи
Select layout: **euler** Randomize **Start** **Stop**

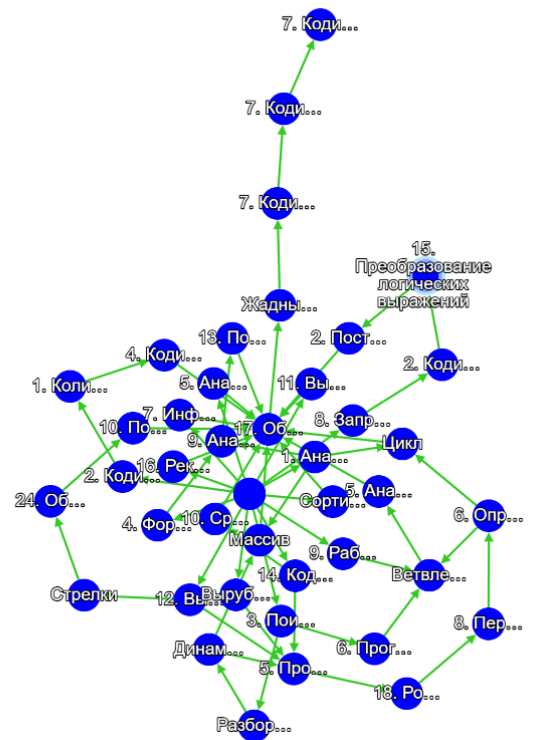


Рисунок 2.14 - Граф вида “euler”

Для того, чтобы выстроилась индивидуальная траектория необходимо выбрать необходимую задачу, к уровню которой стремишься. Тогда получится как на рисунке 2.15.

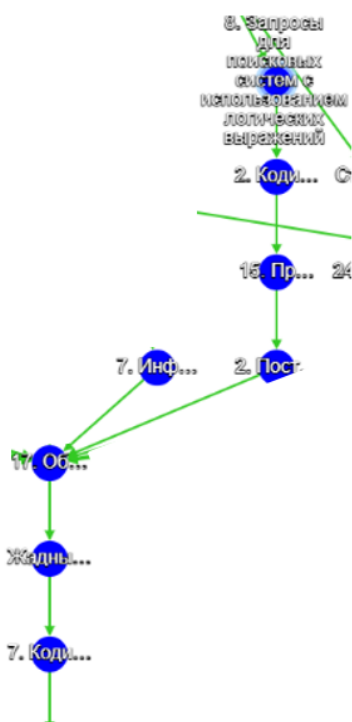


Рисунок 2.15 - Индивидуальная траектория

2.2. Методические рекомендации по наполнению и использованию среды

Данный продукт может применяться для поддержки индивидуализированной подготовки обучающегося, исходя из его запроса и индивидуальных особенностей.

Среду стоит наполнить задачами и проектами от начальной до максимальной сложности. Аннотацию задача следует делать, исходя из необходимого минимума. Отдельные авторы [6] переносили в среду содержание задачник курса и наборы задач для подготовки к ОГЭ.

После наполнения среды туда можно поместить ноду, описывающую желаемый образовательный результат или их набор, и выполнить построение графа сложности курса.

При использовании задач из интернета желательно сохранять обратные ссылки - это позволит обратиться к авторскому порталу и работать там, используя уже доступные инструменты тестирования. Как результат, наполнение ЭОС задачами с курса acmp.ru, Stepic, pythontutor.ru позволит наполнить ЭОС и задачами с готовыми тренажёрами, и с форумами, содержащими много ценной для обучающегося информации.

В случае цикла в графе учитель с учеником могут выбрать ветвь образовательной траектории, проходящей через известные, апробированные курсы.

При подготовке к мультидисциплинарным дисциплинам (STEM: робототехника, машинное обучение, искусственный интеллект) целесообразно добавление задач отдельных дисциплин (см. п. 2.3.2).

После всех этих операций и с помощью данной среды может быть выстроена индивидуальная цепочка конкретных задач, после освоения которых обучающийся сможет достигнуть своей цели.

Для каждого обучающегося цели могут быть различными, поэтому варианты траекторий будут самыми разными. Кто-то захочет решить определенную задачу, а кто-то полностью освоить курс по программированию и/или решению уравнений.

Прелесть данной разработки в том, что ее могут применять учителя не только информатики, но и математики, физики и т.д. Среда может совмещать в себе несколько курсов одной предметной области, несколько задачников. И теперь не нужно будет вручную тратить время и подбирать задачи, достаточно будет лишь задать необходимые критерии, которые запрашивает отдельный обучающийся.

Применяться данная технология может на дополнительных занятиях, элективных курсах, при индивидуальной работе для отстающих или, наоборот, вперед идущих учеников.

На данном этапе в электронную образовательную среду добавлены задачи ОГЭ и ЕГЭ по информатике, а также некоторые олимпиадные задачи. На основе этих задач построен граф, показывающий как они взаимосвязаны между собой.

2.3. Кейсы

В качестве примеров проектирования отдельных образовательных маршрутов, приведём наиболее типичные кейсы:

1. Олимпиадное программирование.
2. Робототехника.
3. Машинное обучение.

2.3.1 Олимпиадное программирование

Олимпиадное программирование характеризуется высоким уровнем сложности задач. Как правило, оно требует знания жадных алгоритмов, рекурсии, и ряда специфических алгоритмов, которые относятся к конкретным узким областям. Также оно требует умения писать сложные функции с использованием нескольких парадигм.

Примеры задач такого рода приведены ниже.

1. **“Волосатый бизнес”**. Одного неформала выгнали с работы, и теперь ему надо как-то зарабатывать себе на жизнь. Поразмыслив, он решил, что сможет иметь очень неплохие деньги на продаже собственных волос. Известно, что пункты приема покупают волосы произвольной длины стоимостью C у.е. за каждый сантиметр. Так как волосной рынок является очень динамичным, то цена одного сантиметра волос меняется каждый день как и курс валют. Неформал является очень

хорошим бизнес-аналитиком. Он смог вычислить, какой будет цена одного сантиметра волос в каждый из ближайших N дней (для удобства пронумеруем дни в хронологическом порядке от 0 до $N-1$). Теперь он хочет определить, в какие из этих дней ему следует продавать волосы, чтобы по истечению всех N дней заработать максимальное количество денег. Заметим, что волосы у неформала растут только ночью и вырастают на 1 сантиметр за ночь. Следует также учесть, что до 0-го дня неформал с горя подстригся наголо и к 0-му дню длина его волос составляла 1 сантиметр. *Жадный алгоритм* [49].

2. **“Неподвижные точки”**. Перестановкой $P[1..n]$ размера n называется набор чисел от 1 до n , расположенных в определенном порядке. При этом в нем должно присутствовать ровно один раз каждое из этих чисел. Примером перестановок являются 1,3,4,5,2 (для $n=5$) и 3,2,1 (для $n=3$), а, например, 1,2,3,4,5,1 перестановкой не является, так как число 1 встречается два раза. Число i называется неподвижной точкой для перестановки P , если $P[i] = i$. Например, в перестановке 1,3,4,2,5 ровно две неподвижных точки: 1 и 5, а перестановка 4,3,2,1 не имеет неподвижных точек. Даны два числа: n и k . Найдите количество перестановок размера n с ровно k неподвижными точками. *Рекурсия, перебор* [49].

3. **“Зайчик”**. В нашем зоопарке появился заяц. Его поместили в клетку, и чтобы ему не было скучно, директор зоопарка распорядился поставить в его клетке лесенку. Теперь наш зайчик может прыгать по лесенке вверх, перепрыгивая через ступеньки. Лестница имеет определенное количество ступенек N . Заяц может одним прыжком преодолеть не более K ступенек. Для разнообразия зайчик пытается каждый раз найти новый путь к вершине лестницы. Директору любопытно, сколько различных способов есть у зайца добраться до вершины лестницы при

заданных значениях K и N . Помогите директору написать программу, которая поможет вычислить это количество. Например, если $K=3$ и $N=4$, то существуют следующие маршруты: $1+1+1+1$, $1+1+2$, $1+2+1$, $2+1+1$, $2+2$, $1+3$, $3+1$. Т.е. при данных значениях у зайца всего 7 различных маршрутов добраться до вершины лестницы. *Динамическое программирование* [49].

Список дополнительных дисциплин широк:

- геометрия;
- бинарный поиск;
- динамическое программирование;
- жадные алгоритмы.

Требуется знать:

- комбинаторику;
- математическое моделирование;
- работу со строками;
- системы счисления;
- различные структуры данных.

Также требуется знать элементы теории графов, различные сортировки последовательностей, - и этот список далеко не полон.

По каждой из техник и тем желательно внести несколько задач от крайне простых, буквально иллюстрирующих технику, до задач уровня края и всероссийской олимпиады. Несколько задач приведены ниже.

1. **Геометрия.** Отрезок задан координатами своих концевых точек. Требуется вычислить длину этого отрезка. Сложность 12%.
2. **Геометрия.** Группа экспертов обнаружила на территории нежилого

массива множество опасных участков, соприкосновение с которыми небезопасно для жизни человека. В целях безопасности требуется создать защитный периметр в форме выпуклого многоугольника, который бы смог обезопасить проникновение человека в эту зону. По заданным координатам опасных участков требуется вычислить минимально возможную площадь опасной зоны, которая попадет в защитный периметр. Сложность 57%.

3. **Теория графов.** В галактике «Milky Way» на планете «Snowflake» есть N городов, некоторые из которых соединены дорогами. Император галактики «Milky Way» решил провести инвентаризацию дорог на планете «Snowflake». Но, как оказалось, он не силен в математике, поэтому он просит вас сосчитать количество дорог. Требуется написать программу, помогающую императору сосчитать количество дорог на планете «Snowflake». Сложность 18%
4. **Теория графов.** Дан ориентированный взвешенный граф. По его матрице смежности нужно для каждой пары вершин определить: существует кратчайший путь между ними или нет. Кратчайший путь может не существовать по двум причинам: либо нет ни одного пути, либо есть путь сколь угодно маленького веса. Сложность 65%.

2.3.2. Робототехника

Робототехника как дисциплина требуют знания физики, математики и программирование.

Из физики требуется знать механику:

- золотое правило механики;
- понятия силы, пути, скорости, мощности;
- нужно знать электрические цепи: законы Ома, сопротивление, ток, напряжения, физические свойства элементной базы.

Нужно уметь спроектировать устройство в режиме инженера, что тоже требует соответствующих компетенций.

Для того, чтобы решать возникающие задачи и описывать все эти физические процессы, требуется знать математику: алгебру, геометрию и тригонометрию для описания шестерёночных механизмов и колёс, и т.д.

Реализация алгоритмов интенсивно использует компетенции программиста.

Все эти ветви следует покрывать задачами снизу-вверх, от простого к сложному, и все эти темы по отдельности в литературе уже достаточно хорошо проработаны. Литература и материалы для этого доступны.

2.3.3. Машинное обучение

Машинное обучение возникло на стыке искусственного интеллекта, статистики, теории вероятностей, и математического анализа. Соответствующие задачи требуется добавить в систему. После первоначального знакомства с аппаратом каждой дисциплины в минимально необходимом объёме, можно приступать к решению практических задач по возрастанию сложности.

Выводы по второй главе

В данной главе был описан разработанный программный продукт, его интерфейс и функционал на данный момент.

Сейчас среда наполнена задачами ОГЭ и ЕГЭ по информатике, а также олимпиадными задачами. Граф действительно отображает взаимосвязь между

задачами, опираясь на те понятия, которые используются в каждой из них.

Были приведены примеры индивидуальных кейсов, которые можно сделать с помощью данной электронной среды.

Программный продукт универсален и может быть наполнен различным содержанием, даже смежных областей. Его можно далее усовершенствовать и использовать в работе.

Заключение

В заключение отметим основные результаты и выводы, полученные в ходе проделанной работы.

Во-первых, был разработан программный продукт в виде электронной среды для помощи в проектировании детального индивидуального учебного плана минимальной трудоемкости с учётом персональных целей ученика и опыта педагога.

Во, вторых, были исследованы литературные источники и авторские разработки, которые подтвердили актуальность и новаторство данного продукта.

В-третьих, приведены примеры индивидуальных кейсов, которые можно сделать с помощью данной электронной среды.

В-четвертых, разработанный программный продукт является универсальным и может быть в дальнейшем усовершенствован и использован в других областях знания.

Современный учитель должен быть не только транслятором знаний, но и наставником, навигатором и помощником, должен успеть учесть особенности и запросы каждого обучающегося, однако времени и ресурсов для этого практически нет. Здесь на помощь приходят электронные автоматизированные образовательные ресурсы, которые могут выстроить индивидуальную образовательную траекторию за несколько секунд, при этом будут учтены пожелания и цели конкретного обучающегося.

Программирование - сложный, но не менее интересный раздел информатики, который вызывает множество трудностей как у обучающихся, так и у учителей. Обучение программированию происходит от простого к сложному, при этом за годы обучения у обучающихся возникают пробелы, недопонимания, забывания и т.п. А задачи, которые все усложняются, обрастая комом новых понятий, все более вызывают трудности, которые

можно преодолеть благодаря индивидуальным учебным траекториям.

В заключение можно сказать, что индивидуализация дополнительной подготовки старшеклассников по программированию является важным и необходимым этапом в образовании. Разработка индивидуальных учебных планов с учетом уровня знаний и интересов каждого ученика позволяет достичь наилучших результатов в обучении. Более того, такой подход мотивирует учеников к изучению программирования и развитию своих навыков. Рекомендуется внедрение этой методики в образовательный процесс для достижения наилучших результатов в обучении программированию.

Подводя итог вышесказанному, можно утверждать, что все задачи исследования выполнены, а цель работы достигнута.

Библиографический список

1. Адаптивное обучение в вузе: что это такое, методы, этапы, платформы [Электронный ресурс] URL: <https://zaochnik.ru/blog/adaptivnoe-obuchenie-v-vuze-chto-eto-takoe-metod-y-etapy-platformy/> (дата обращения: 15.05.2023).
2. Адаптивное обучение: что это и зачем нужно [Электронный ресурс] URL: <https://skillbox.ru/media/education/adaptivnoe-obuchenie-chto-eto-i-zachem-nuzhno/> (дата обращения: 11.05.2023).
3. Алексеев Е. П., Чеснокова О. В., Кучер Т. В. Free Pascal и Lazarus: Учебник по программированию. М: ALT Linux, 2012. 438 с.
4. Алексеев Е. Р., Чеснокова О. В., Кучер Т. В. Самоучитель по программированию на Free Pascal и Lazarus. Донецк.: ДонНТУ, Технопарк ДонНТУ УНИТЕХ. 2011. 503 с.
5. Асмолов А. Г., Семенов А. Л., Уваров А. Ю. Российская школа и новые информационные технологии: взгляд в следующее десятилетие / [Асмолов А.Г., Семенов А.Л., Уваров А.Ю.] — М.: Изд-во «НексПринт», 2010. 95с.
6. Балева А. А. Технология построения индивидуального элективного курса по математике на основе дерева знаний: ВКР, 2020.
7. Батурина О. С. Обучение решению задач по программированию массивов в среде Free Pascal / Батурина О. С., Бакаева О. А. // Наука, образование, инновации: гуманитарные, естественно-научные и технические решения современности. 2020. №5. С. 108-110.
8. Богданова Д. А. Об адаптивной платформе для индивидуального обучения [Электронный ресурс] / Д. А. Богданова // Новые образовательные технологии в вузе : материалы XI международной научно-методической конференции. — Екатеринбург, 2014. — Режим

- доступа: <http://hdl.handle.net/10995/24744>.
9. Босова Л. Л. ИКТ как инструмент индивидуализации обучения в современной школе // Преподаватель XXI. 2016. №5. С. 60-67.
 10. Брыкин Ю. В. Электронная образовательная среда: нормативные и содержательные составляющие // Вестник РМАТ. 2018. №7. С. 78-81.
 11. Вайнштейн Ю. В. Адаптивное электронное обучение в современном образовании / Вайнштейн Ю. В., Шершнева В. А. // Педагогика. 2020. №5. С. 48-57.
 12. Войт Н. Н., Беляева И. В. Автоматизация процесса обучения языкам высокого уровня // Информатика, моделирование, автоматизация проектирования. 2016. №8. С. 99-104.
 13. Войтович И. К. Модель электронной образовательной среды вуза // Высшее образование в России. 2016. №9. С. 10-15.
 14. Головачева О. В., Перевалова С. Н. Информационно-образовательная среда школы — новые возможности педагога // Образование: прошлое, настоящее и будущее. 2017. №3. С. 55-57.
 15. Гусякова А. В. Психологические аспекты взаимосвязи механизмов сознания и индивидуализации обучения // Педагогика и психология образования. 2019. №10. С. 35-39.
 16. Дополнительное образование для школьников [Электронные ресурсы] URL: <https://edunews.ru/deti/info/do-v-shkole.html> , (дата обращения: 10.06.2023).
 17. Идиатуллов Т. Т. Применение гибридных систем оценивания в целях повышения качества подготовки и при адаптивном обучении // Социальная политика и социология. 2011. №7. С. 405-419.
 18. Касьянов В. Н., Касьянова Е. В. Дистанционное обучение: методы и средства адаптивной гипермедиа // Программные средства и математические основы информатики. Новосибирск: ИСИ СО РАН. 2004.

- С. 80-141.
- 19.Ковалева Т.М. Антропологический взгляд на современную диактику: принцип индивидуализации и проблема субъективности // Педагогика. 2013. №5. С. 51-56.
 - 20.Кошкин И. С. О педагогическом эксперименте по изучению языков программирования python и pascal в колледже // Актуальные проблемы обучения математике и информатике в школе и вузе. 2020. №4. С. 377-379.
 - 21.Кравченко Д. А. Персонализация в образовании: от программируемого к адаптивному обучению // Современная зарубежная психология. 2020. №9. С. 34-46.
 - 22.Кречетов И. А. Реализация методов адаптивного обучения / Кречетов И. А., Романенко В. В. // Вопросы образования. 2020. №2. С. 252-277.
 - 23.Кузнецов А. А. Электронная образовательная среда нового типа на базе кросс - платформенной модели обучения: методика и апробация // Вестник московского университета. серия 20. педагогическое образование. 2021. №15. С. 78-88.
 - 24.Медведик В. И. Практика программирования на языке Паскаль (задачи и решения). М.: ДМК Пресс, 2013. 590 с.
 - 25.Монахова Г. А., Монахов Н. В. Современные конструкторы микрокурсов // Образование: традиции и инновации. 2020. №24. С. 92-94.
 - 26.Павлов О. Ю., Гизатулин Р. М., Зарипов С. Э. Особенности метода и функциональной структуры адаптивного обучения на основе метода геймификации // Актуальные проблемы науки и образования в современном ВУЗЕ. 2021. №5. С. 497-502.
 - 27.Павловская Т. А. Паскаль. Программирование на языке высокого уровня: Учебник для вузов. 2-е изд. Спб.: Питер, 2021. 464 с.

28. Пак Н. И., Асауленко Е. В. Персонализация самостоятельной работы студентов по формированию умения решать расчетные задачи на основе автоматизированной обучающе-диагностической системы // Информатика и образование. 2018. №8. С. 26-32.
29. Программирование на Pascal [Электронный ресурс] URL: <https://stepik.org/course/3352/promo> (дата обращения: 11.05.2023)
30. Процкая Е. В., создание электронного учебника "Циклы в Pascal" // Информационно-коммуникационные технологии в педагогическом образовании. 2013. №3. С. 11-12.
31. Расулова Н. Ю. Адаптивное обучение в системе высшего образования // Scientific progress. 2021. №3. С. 88-97.
32. Рихтер Т. В. Теория и практика решения математических задач средствами языка программирования Turbo Pascal в условиях дистанционного обучения. ФГБОУ ВПО «Соликамский государственный педагогический институт», 2013. 300 с.
33. Родькина А. Н. Анализ учебников информатики на предмет изучения программирования // Инновационные процессы в современном образовании: Практики, технологии, решения. 2021. С. 190-194.
34. Савченко И. А. Технологии разноуровневой дифференциации как средство индивидуализации обучения школьников // Современные информационные технологии и ИТ-образование. 2015. №4. С. 18-25.
35. Самофалова М. В. Адаптивное обучение как новая образовательная технология // Гуманитарные и социальные науки. 2020. №5. С.90-97.
36. Селиванова Э. Т., Розов К. В. Формирование мотивации у старшеклассников к изучению программирования // Национальная ассоциация ученых. 2016. №13. С.25-28.
37. Сербин В. В. Использование психологических типов в адаптивном процессе обучения / Сербин В. В., Пак В. В. // Студенческий форум.

2019. №17. С. 26-30.
38. Сырицына В. Н., Кадеева О. Е. Модернизация языка программирования Pascal // Инновации в науке. 2017. №3. С. 18-19.
39. Тарануха Н. А. Обучение программированию. Язык Pascal. Учебное пособие / Тарануха Н. А., Гринкруг Л. С., Бурменский А. Д., Ильина С. В. СОЛОН-ПРЕСС, 2009. 384 с.
40. Ушаков Д., Юркова Т. Паскаль для школьников. 2-е изд. Спб.: Питер, 2011. 320 с.
41. Хайрулина Ю. В. Проблемы обучения программированию в школе // Актуальные проблемы преподавания информатики и информатизации образовательного процесса в учреждениях основного и дополнительного образования. 2021. С. 65-71.
42. Харченко Л. Н. Современные тренды развития непрерывного педагогического образования. М.: Директ-Медиа, 2021. 151 с.
43. Царев Р. Ю. Адаптивное обучение с использованием ресурсов информационно-образовательной среды / Царев Р. Ю., Тынченко С. В., Гриценко С. Н. // Современные проблемы науки и образования. 2016. №5. С. 219.
44. Цифровой репетитор Пларио [Электронный ресурс] URL: <https://plario.ru/> (дата обращения 12.06.2023).
45. Цифровые технологии адаптивного обучения [Электронный ресурс] URL: <https://urait.ru/news/2665> (дата обращения: 10.05.2023).
46. Чиксентмихайи М. Поток: психология оптимального переживания. – Альпина Паблишер, 2019.
47. Шихнабиева Т. Ш. Адаптивные семантические модели автоматизированного контроля знаний // Педагогическое образование в России. 2016. С. 1-7.
48. Шихнабиева Т. Ш. О методологии и методике представления и

контроля знаний в области педагогической информатики на основе адаптивных семантических моделей // Вестник Московского государственного областного университета. Серия: Педагогика. 2008. №3. С. 204-213.

49. Школа программиста [Электронный ресурс] URL: <https://acmp.ru/index.asp?main=tasks> (дата обращения 9.06.2023).

50. Шульга О. В. Методические аспекты обучения программирования на языке паскаль в контексте внеурочной работы // Сфера знаний: вопросы взаимодействия наук в современном образовательном процессе. 2018. №3. С. 101-103.