

*И.В. Барышева, Е.В. Малкина, О.А. Козлов*

## **ПРОЕКТНЫЙ МЕТОД ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ СТУДЕНТОВ ПРОФИЛЬНЫХ СПЕЦИАЛЬНОСТЕЙ В УСЛОВИЯХ ДИСТАНЦИОННОЙ РАБОТЫ**



**БАРЫШЕВА Ирина Викторовна** – преподаватель кафедры математического обеспечения и суперкомпьютерных технологий Института информационных технологий, математики и механики;  
Нижегородский государственный университет им. Н.И. Лобачевского  
пр. Гагарина, 23, г. Нижний Новгород, 603950, Россия; e-mail: [ibar1950@yandex.ru](mailto:ibar1950@yandex.ru)

**BARYSHEVA Irina V.** – Lobachevsky State University of Nizhny Novgorod.  
23 Prospekt Gagarina, Nizhny Novgorod, 603022, Russia;  
e-mail: [ibar1950@yandex.ru](mailto:ibar1950@yandex.ru)



**МАЛКИНА Елена Владиславовна** – кандидат педагогических наук, доцент кафедры программной инженерии Института информационных технологий, математики и механики;

Нижегородский государственный университет им. Н.И. Лобачевского  
пр. Гагарина, 23, г. Нижний Новгород, 603950, Россия; e-mail: [malkina@unn.ru](mailto:malkina@unn.ru)

**MALKINA Elena V.** – Lobachevsky State University of Nizhny Novgorod.  
23 Prospekt Gagarina, Nizhny Novgorod, 603022, Russia;  
e-mail: [malkina@unn.ru](mailto:malkina@unn.ru)



**КОЗЛОВ Олег Александрович** – доктор педагогических наук, профессор, ведущий научный сотрудник;

Институт стратегии развития образования Российской академии образования.  
ул. Жуковского, 16, Москва, 101000, Россия; e-mail: [ole-kozlov@yandex.ru](mailto:ole-kozlov@yandex.ru)

**KOZLOV Oleg A.** – Institute of Education Development Strategy of the Russian Academy of Education  
ul. Zhukovskogo, 16, Moscow, 101000, Russia;  
e-mail: [ole-kozlov@yandex.ru](mailto:ole-kozlov@yandex.ru)

---

В статье рассматриваются проблемы проведения практических и лабораторных работ в курсе «Алгоритмы и структуры данных» при подготовке будущих ИТ-специалистов дневного отделения в условиях перехода на онлайн обучение во время пандемии. Особое внимание уделяется методике изучения различных структур данных как базы подготовки разработчиков информационных систем. Предлагается методика проведения онлайн занятий с использованием презентаций. Выделяются основные компоненты при подготовке презентаций. В рамках дисциплины «Алгоритмы и структуры данных» студент должен выполнить семь лабораторных работ. Каждая лабораторная работа – это программный проект, для выполнения которого требуется несколько занятий. Во время этих занятий вырабатывается структура проекта и последовательно увеличивается количество создаваемых классов-подпрограмм для решения поставленной задачи. Показа-

но, как вовлечение студентов в совместное программирование может поднять лабораторные программные проекты на уровень проектов Scrum. Анализируется проектный метод, и приводятся некоторые результаты применения предложенной методики при ведении практических и лабораторных занятий.

АЛГОРИТМЫ; ДИСТАНЦИОННОЕ ОБУЧЕНИЕ; ПРОГРАММНЫЙ ПРОЕКТ; ПРОЕКТНЫЙ МЕТОД; СТРУКТУРЫ ДАННЫХ

---

**Ссылка при цитировании:** Барышева И.В., Малкина Е.В., Козлов О.А. Проектный метод обучения программированию студентов профильных специальностей в условиях дистанционной работы // Вопросы методики преподавания в вузе. 2021. Т. 10. № 38. С. 40–55. DOI: 10.18720/HUM/ISSN 2227-8591.38.04

---

**Введение.** Дисциплину «Алгоритмы и структуры данных» много лет преподают в Нижегородском университете им. Н.И. Лобачевского в третьем и четвертом семестрах студентам, обучающимся на профильных специальностях [1]. Лекционный курс сопровождается практическими и лабораторными занятиями [2]. За время существования курса сменялись поколения компьютеров, языки программирования, используемые на практике, и интегрированные среды для разработки программного обеспечения. За это время развивались не только формальные методологии разработки программ – структурное программирование, объектно-ориентированное программирование и другие, – но менялись и требования к конечному продукту при его промышленной разработке, и этого нельзя не учитывать, обучая программированию. Требования к конечному продукту породили изменения требований к проектировке программных продуктов. Появилась новая дисциплина, и даже новая специальность – «Программная инженерия». В профессиональном сообществе в последние годы при разработке программных проектов исходят из гибких, итеративных методологий, основанных на идеологии Agile, которая описана в электронном ресурсе Agile Manifesto [3]. Agile предлагает следовать некоторым ценностям, которые помогают разработчикам отбросить второстепенное,

и сосредоточиться на главном, применять наиболее простые решения, эффективно взаимодействовать между собой и с заказчиком, адекватно реагировать на изменения. Команда разработчиков должна систематически анализировать возможные способы улучшения эффективности и соответственно корректировать стиль своей работы.

Существует множество методов, базирующихся на идеях Agile, и один из самых популярных – Scrum [4]. В любом случае речь идет о некотором проекте и методе его реализации. Проект в широком смысле, как известно, имеет такие существенные характеристики, как цель, уникальность, ограниченность по времени, ограниченность по ресурсам (людским и материальным), сложность, неопределенность (гибкость) и другие. В Scrum весь проект разбивается на ряд небольших задач-итераций, которые выполняются за равные промежутки времени. В Scrum такие промежутки их называют спринтами. Вся работа выполняется небольшой командой, в которую входят разработчики, владелец продукта (заказчик) и скрам-мастер (отвечающий за эффективность и правильное применение Scrum). Команда регулярно встречается, чтобы планировать и находить способы решения поставленных задач [5].

В этом смысле проведение лабораторных работ как педагогических проектов,

выполняемых в режиме онлайн, хорошо вписывается в методологию Scrum. Каждая лабораторная – это отдельный проект, длящийся от полутора до семи-восьми недель. Команда разработчиков – группа студентов из 12-15 человек. Спринты – обычные учебные пары. На каждую учебную пару ставится своя задача, по которой должен быть получен свой результат. При очном обучении в терминал-классе все студенты за своим компьютером выполняли одну и ту же задачу, а преподаватель консультировал каждого студента по отдельности. В условиях дистанционной работы с использованием видеоконференций и возможностью каждому участнику делиться функциями своей системы, получать мгновенную обратную связь, все становятся равноправными участниками совместной онлайн деятельности. Таким образом, учебная группа становится командой, работающей над одной задачей. Единственное отличие методологии Scrum от дистанционной работы над проектом студенческой группы: владелец продукта, заказчик и скрам-мастер одно и то же лицо – преподаватель, и для него нет неопределенности в смысле результатов проекта.

В данной статье мы сосредоточимся на анализе главных, на наш взгляд, методических проблемах онлайн обучения программированию студентов дневного отделения в условиях чрезвычайных обстоятельств (пандемии). Это касается представления изучаемого материала, обеспечения взаимосвязи «учитель-ученик», не затрагивая материальные проблемы, обеспеченности техническими средствами, качества интернет-соединения, наличия лицензионного программного обеспечения учителя и учеников и тому подобные.

**Актуальность.** Чтение лекций «онлайн» появилось раньше пандемии, причем в разных вариантах: это и популярные лекции в различных образовательных телеви-

зионных программах, и стримы в интернете, и онлайн семинары (вебинары). Большое количество учебных материалов доступно в записях, которые хранятся на разных платформах и, в частности, на YouTube: например, замечательные уроки физики в Ришельевском лицее [6], Гарвардский курс «Основы компьютерных наук и искусства программирования CS50» [7], множество видеоблогов, рассказывающих, как решать конкретные практические задачи, в том числе в области программирования. Университеты во всем мире предлагают как изучение отдельных онлайн курсов, так и получение некоторых специальностей. Несмотря на то, что дистанционное образование уже имеет долгую историю, тем не менее, изучение программирования в режиме онлайн все еще недостаточно разработано в литературе [8]. В России образование формализовано образовательными стандартами, рабочими программами дисциплин (РПД), оценочными средствами (ФОС). Нарушение формальных норм высшим учебным заведением может привести даже к недовольству обучающихся, закрытию соответствующей программы обучения. Поступив на дневное обучение, студенты хотят получить тот объем знаний и в том виде, на который они рассчитывали при поступлении. Это одна из причин, почему руководство университета, переводя студентов на онлайн обучение, не дает переходить на более свободное асинхронное дистанционное обучение, а продолжает требовать работы строго по расписанию, включая лабораторные работы, которые обычно проводятся в терминал-классах. Было бы неэффективно дать студентам выполнять самостоятельно лабораторные работы на домашних компьютерах, а преподавателю сидеть целую пару перед экраном Zoom, на котором изображены маленькие черные прямоугольники с фамилиями студентов, и ждать, не задаст

ли кто-нибудь вопрос. Поэтому каждая лабораторная, которую студенты должны выполнить, превращается в проект в смысле методологии Scrum и выполняется совместно.

Дневное обучение в классическом университете предполагает ежедневное общение студентов с преподавателями, совместное обсуждение учебных и научных вопросов, причем не только во время занятий. Неожиданный массовый переход на дистанционное обучение в начале пандемии Covid-19 поставил перед преподавателями, традиционно ведущими очные занятия, сложную задачу: обеспечить то же качество, что и при очном обучении. При таком резком переходе преподавателям и студентам пришлось столкнуться с множеством проблем, которые необходимо было решать очень быстро: освоить новые возможности взаимодействия со студентами, обеспечить их учебными материалами. Исключение очной формы ведения занятий разорвало установившиеся связи между традиционными формами представления предмета, изменило акценты и соотношения между лекциями, практической работой и самостоятельной работой студентов. Самые болезненные и невосполнимые потери – это нарушение непосредственных связей «учитель–ученик», как бы они ни назывались: учитель–школьник, преподаватель (или профессор) – студент, и параллельных отношений «ученик–ученик». Ушли на второй план выработанные годами методики преподавания, а новые – находятся еще только в процессе зарождения и разработки. При этом осталась ответственность преподавателя по выполнению учебного плана.

В Нижегородском университете довольно быстро для чтения лекций и проведения практических занятий в курсе «Алгоритмы и структуры данных» выбрали платформу Zoom. Методические материалы можно

было размещать на сайте под управлением системы электронного обучения MOODLE и на портале университета. Для изучения курса «Алгоритмы и структуры данных» студентам, лишившимся терминал-класса, было рекомендовано установить у себя на компьютерах бесплатную версию Visual Studio Community 2019 с сайта <https://visualstudio.microsoft.com/ru/vs/community/>. Вопрос заключался только в том, что преподаватели практики, ведущие очные занятия, не имели готовых материалов для видеоконференций, их нужно было готовить заново. А еще нужно было как-то решать вопрос с проведением лабораторных занятий.

Преподавать программирование очно достаточно сложно, еще сложнее преподавать онлайн. До сих пор нет единого мнения о том, что такое процесс программирования и как ему следует обучать. Проблема обучения программированию остается актуальной на протяжении многих лет, и появление новых программных продуктов, новых направлений подготовки не снижает интереса к этой научно-педагогической задаче [9; 10; 11; 12; 13].

При очном обучении на практических занятиях преподаватель в обычной аудитории обсуждает со студентами выполнение предстоящей лабораторной работы. При этом по реакции слушателей всегда видно, насколько они понимают материал. Всегда можно остановиться, объяснить по-другому, вернуться к начальному варианту.

Ведение практических занятий без традиционных доски и мела требует огромной предварительной работы по подготовке методических материалов. Хорошо известным и часто применяемым инструментом представления учебного материала являются презентации, но даже в рамках одного предмета методически различаются презентации для лекций и презентации для проведения практических занятий. В Ниже-

городском университете при чтении лекций по дисциплине «Алгоритмы и структуры данных» в аудитории презентации используются давно, они опубликованы как дидактический материал [14], более того, практика и лабораторные работы по содержанию синхронизированы с лекциями.

В чем различия в представлении материала для лекций и для проведения практических занятий? Первое – целевая аудитория: лекции читаются на потоке в 150-200 студентов, эмоции и реакция коллективная, а практические занятия ведутся с подгруппой из 12-15 слушателей, контакт в данном случае должен быть фактически персональным. Для обеспечения таких различий применяются различные методики.

Второе – различаются цели: студент, слушая лекцию, должен понять материал, узнать, какие бывают методы, какую-то часть записать в виде конспекта (хотя при наличии электронной презентации запись зачастую может рассматриваться как рудимент, но не надо забывать, что при ручной записи включается еще и механическая память и начинают работать другие разделы мозга, что способствует лучшему восприятию материала). С помощью лабораторных работ студенты должны научиться применять изученные ими методы, получить совершенно определенный результат, соизмеряя при этом экономию памяти компьютера и быстродействие программы. Цель любой практической работы заключается в том, чтобы обучаемый научился самостоятельно получать реальный результат.

**Проектный метод.** В рамках дисциплины «Алгоритмы и структуры данных» студент должен выполнить семь лабораторных работ. Каждая лабораторная работа – это программный проект [15; 16; 17], на выполнение которого требуется несколько занятий. Во время этих занятий вырабатывается структура проекта и последователь-

но увеличивается количество создаваемых классов-подпрограмм для решения поставленной задачи. Темы программных проектов [14; 18]:

- Алгебра множеств
- Работа с матрицами специального вида на примере треугольных матриц
- Транслятор арифметических выражений. Динамическая структура «стек». Динамическая структура «таблица»
- Динамическая структура «очередь». Поразрядная сортировка
- Алгебра полиномов от  $N$  переменных. Циклический список
- Орфографический словарь. Иерархический список как структура
- Графический редактор. Плексы.

Этот список может варьироваться: например, иерархический список, используемый для организации орфографического словаря, может быть рассмотрен на примере хранения текстов; динамическая структура «стек» может быть проиллюстрирована на более простой задаче контроля парности скобок в арифметическом выражении; различные формы организации динамической структуры «таблица» могут быть выполнены в виде отдельных лабораторных работ. Выбор конкретного набора лабораторных работ зависит как от уровня конкретной группы студентов, так и от методики ведения занятий преподавателем.

При работе над проектами изучаются теоретические вопросы программирования, приобретается опыт разработки проектов, их реализации, описания проделанной работы и полученных результатов. Проектный метод при выполнении лабораторных работ предусматривает реализацию модели реальной жизни в форме «заказчик–исполнитель». При этом должна присутствовать постановка задачи «заказчиком», построена модель решения и предложены варианты решения. В традиционном формате все это требует живого

обсуждения как по линии «преподаватель-студент», так и между студентами. Преподаватель в режиме реального времени должен реагировать на реакцию аудитории. В дистанционном формате для удержания фокуса внимания студентов материал должен быть представлен в виде, допускающем варианты выбора, должен предугадывать возможные вопросы и содержать ответы на еще не поставленные вопросы.

Например, в лабораторной работе «Алгебра множеств» постановка задачи имеет вид:

- 1) разработать систему хранения конечных множеств;
- 2) обеспечить выполнение операций над множествами;
- 3) сформировать дружественный интерфейс.

Так выглядит постановка содержательной части задачи, но необходимо помнить и об учебных целях, ради которой разрабатывается проект, а именно:

- научиться использовать WindowsForm для организации интерфейса;

- разработать структуру программы с использованием WindowsForm;

- познакомиться с побитовыми логическими операциями;

- изучить операции сдвига.

Перечень учебных целей, достигнутых в процессе выполнения проекта, для каждой лабораторной работы студенты должны представлять в отчете в разделе итогов и заключений.

Презентация при этом не является главным источником информации: материал слайдов есть иллюстрация к голосу преподавателя, он служит дидактическим материалом, слайды не должны заменять учебники. В этом варианте дистанционное занятие может приблизиться к обычному очному формату, так как слайд становится продвинутым аналогом доски. Достичь приближения к аудиторным занятиям позволяет представление двух или даже более вариантов разработки проекта с оценкой сложности каждого варианта, экономии времени вычисления и используемой памяти с целью вызвать обсуждение и перевести занятие в интерактивный режим.

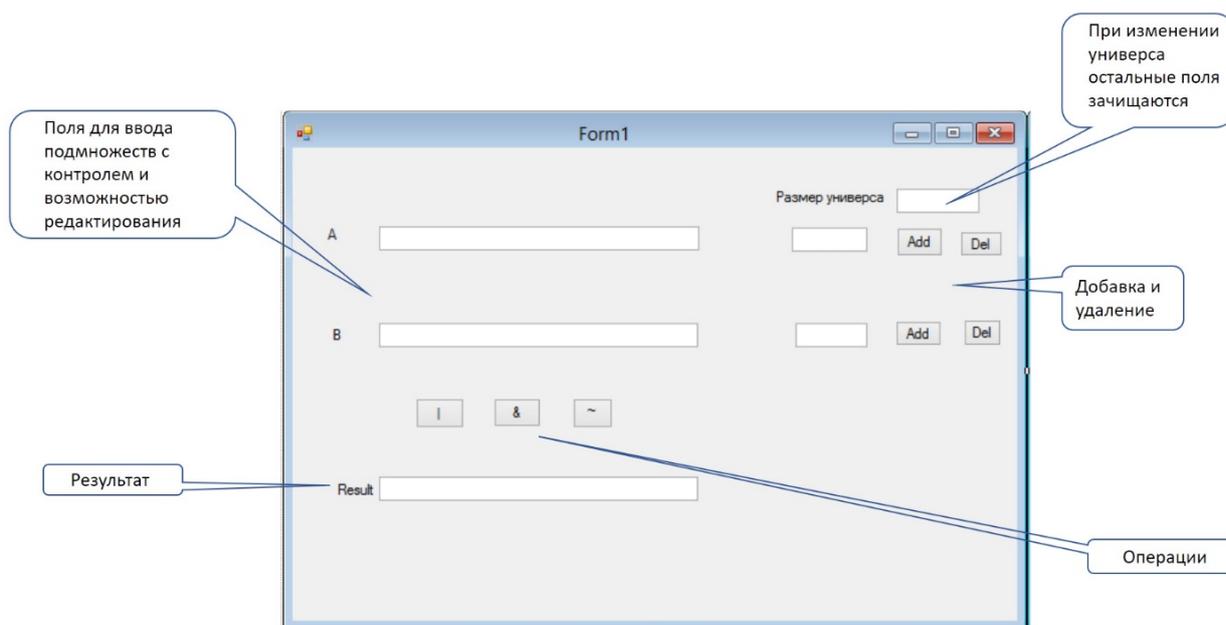


Рис. 1. Работа с множествами

В обсуждаемом примере вариантами могут стать различные структуры хранения множеств: массив номеров элементов, входящих в подмножество, или битовая строка, указывающая на принадлежность элементов подмножеству. В презентации приводится механизм, устанавливающий взаимно-однозначное соответствие между номером элемента и битом, соответствующим этому элементу. Оценки сложности вариантов по времени и по памяти приводятся в презентации в таблицах.

Более сложные проекты выдвигают и более жесткие требования к презентации. Например, в проекте «Транслятор арифметических выражений» технически сложным моментом является воспроизведение в презентации алгоритма преобразования арифметического выражения, в состав которого входят операнды (константы и имена переменных), знаки арифметических операций и скобки (левая и правая) в польскую инверсную запись (полиз). Можно описать алгоритм словами:

1) *Операнд заносится в полиз и таблицу переменных;*

2) *левая скобка помещается в стек;*

3) *правая скобка «выталкивает» из стека в полиз все операции до соответствующей левой скобки;*

4) *«операция» «выталкивает» в полиз из стека все более «важные» или равнозначные по приоритету операции;*

5) *пункт 4 для равнозначных операций обеспечивает выполнение слева направо последовательности операций одинаковой «важности».*

Обучающий момент состоит в том, что описание алгоритма содержит те понятия, ради изучения которых и выполняется проект (это и полиз, и стек, и таблица и многое другое). При очном обучении обычно на каком-либо эффектном примере с помощью доски и мела показывается процесс получения бесскобочной записи, и

по мере разбора примера подводятся итоги работы над примером, фиксируется последовательность выполненных манипуляций и получается схема, определяющая постановку проблемы. Таким образом, под руководством преподавателя студенты сами приходят к пониманию и построению представленного алгоритма. Процесс соучастия в построении алгоритма вдохновляет студентов значительно больше, чем, если бы им предложили написать программу по готовому алгоритму.

На рис. 2 представлен порядок действий при обработке арифметического выражения для данной лабораторной работы. В онлайн этот сложный процесс необходимо продемонстрировать в презентации. Единственное преимущество перед доской – возможность повторения показа. Однако остается проблема удержания внимания студентов, которые должны разобраться в вопросе, чтобы написать адекватную программу. Повторим хорошо известную истину: программировать можно только то, что хорошо понимаешь, в чем полностью разбираешься. Презентация позволяет использовать анимацию, но для грамотной анимации нужно привлекать специалиста-дизайнера, а преподавателю программирования перед проведением конкретного занятия и перед конкретной аудиторией сделать это трудно, и поэтому он создает свою авторскую анимацию, которая показывает, как он понимает проблему.

Обсуждение новых понятий должно начинаться с четкого определения, например:

*Динамическая структура  $S = (M_i, p_1, p_2)$ , в которой отношение следования  $p_1$  (следующий) реализовано операцией «вставки», а отношение следования  $p_2$  (предыдущий) – операцией «исключения», организованными по правилу LIFO (last in, first out, «последним зашел – первым вышел») называется стек.*

С другой стороны, мы можем продемонстрировать это понятие на рисунке (рис. 3):

## Порядок действий при обработке арифметического выражения

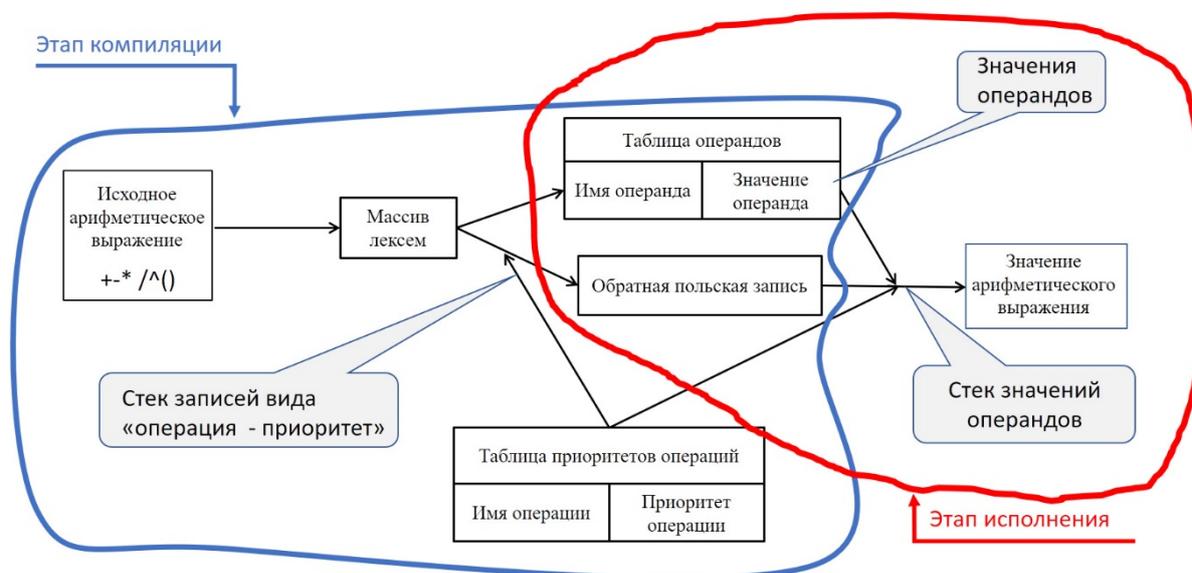


Рис. 2. Схема проекта

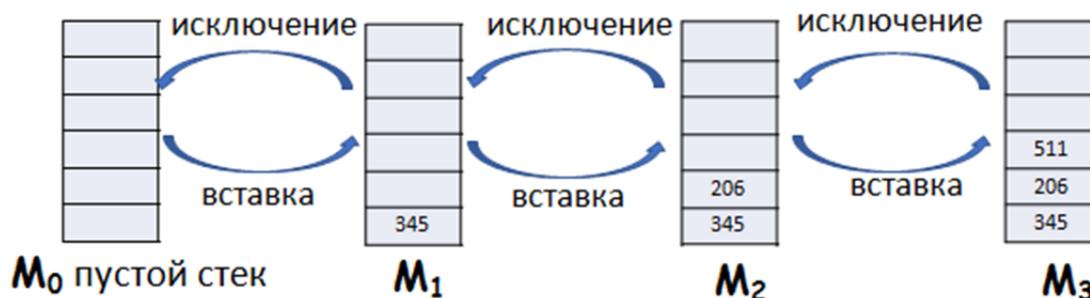


Рис. 3. Динамическая структура стек

При этом динамические структуры, изучение которых является практически главной темой курса «Алгоритмы и структуры данных», требуют и динамическую картинку, поскольку свойства динамических структур проявляются во времени, а не статично. Сам процесс нельзя показать на примере работающей программы, все основные характеристики стека остаются недоступны для показа во время работы программы. Обычно это показывается на доске.

Так, например, в рамках выполнения лабораторной работы «Трансляция арифметических выражений» необходимо организовать две таблицы. Таблица как дина-

мическая структура появляется в учебном курсе здесь впервые. Существуют четыре способа организации таблицы, и необходимо предложить студентам выбор, который осуществляется на основе характеристик каждого способа затрат как по времени, так и по памяти. Эти данные представлены в презентации, и при правильной организации внимание слушателей получит дополнительный импульс, тем более, что выбор происходит с учетом проектных требований к таблицам.

Постановка задачи принимает вид:

1) для организации интерфейса должна быть использована форма в приложении WindowsForm (рис. 4);

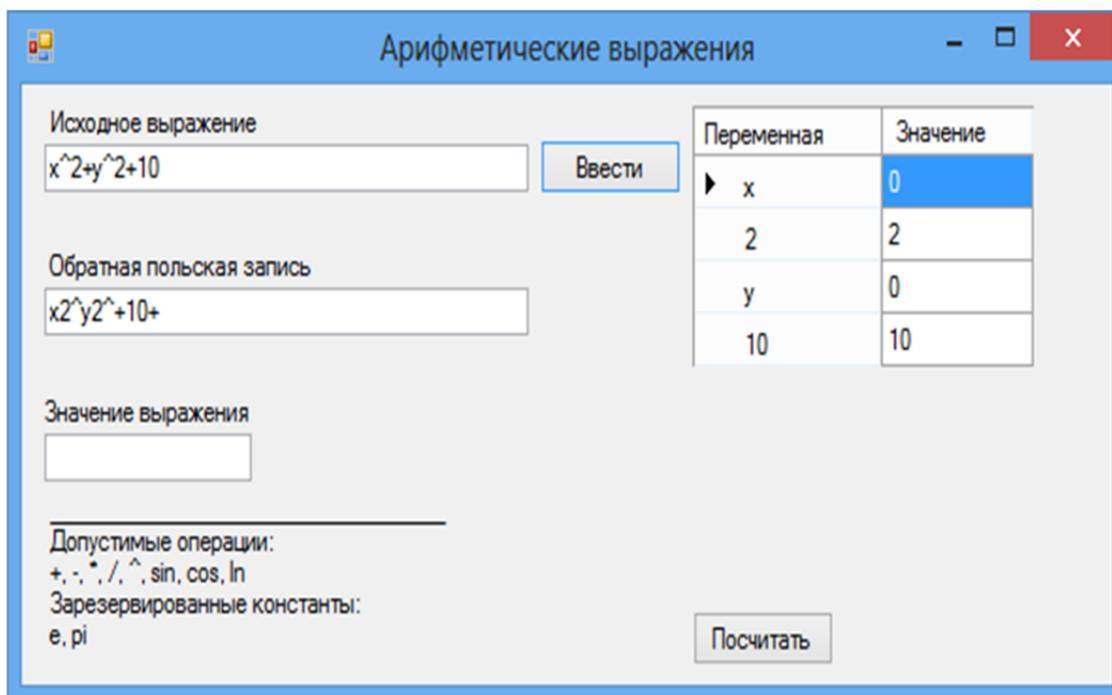


Рис. 4. Задание на лабораторную работу

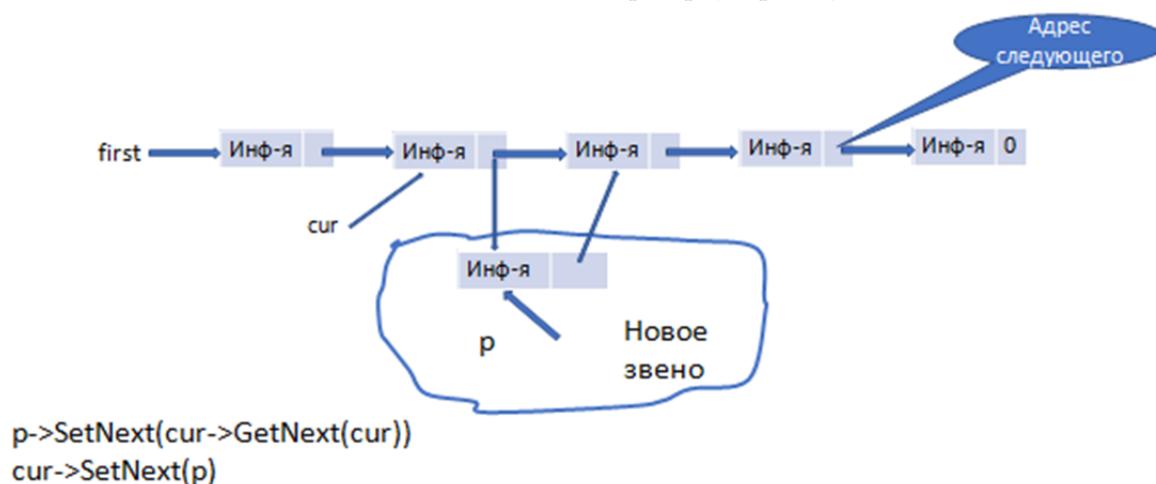


Рис. 5. Вставка нового звена в линейный список

2) в поле «Исходное выражение» на форме задается арифметическое выражение, использующее допустимый заданный набор операций; имена переменных и константы могут состоять из одного или более символов;

3) необходимо получить бесконечную запись выражения и таблицу использованных в выражении переменных;

4) после определения значений переменных в таблице, представленной на форме, должно появиться значение выражения.

Следующая проблема в использовании презентаций – это иллюстрация процесса формирования сложных структур хранения: иерархических списков, деревьев. Отметим, что само понятие «списки» требует внимательного рассмотрения. На рис. 5 представлен алгоритм вставки нового звена в линейный список.

Без динамики процесс вставки нового звена в список объяснить сложно: не имеет смысла записать просто строки текста без

иллюстрации последовательности действий, тем более, что понимание структуры хранения, распределенной в памяти компьютера, формируется у студентов не сразу.

Постановка задачи для выполнения последней лабораторной работы «Графический редактор. Плекс» [14;18;19] предполагает, прежде всего, иллюстрацию процесса формирования структуры хранения геометрических объектов в виде ветвящегося списка со звеньями разного типа (плекса) по мере появления рисунка. Выбранная структура хранения позволяет произвести однозначное воспроизведение рисунка по полученному плексу, хотя ситуация осложняется неоднозначностью отношения «рисунок – плекс». Одному плексу соответствует один рисунок, но плексов для одного рисунка может быть несколько, и каждый вариант отличается от другого последовательностью добавления элементов рисунка, то есть зависит от динамики процесса формирования рисунка, которую необходимо показать при постановке задачи.

Например, для рисунка простого треугольника (рис. 6) при рисовании сторон в последовательности АВ, АС, ВС получаем вариант 1, а в последовательности АС, АВ, ВС – вариант 2.

На доске все просто: один студент мелом чертит рисунок, добавляя в него элементы, а другой синхронно добавляет но-

вые звенья в плекс, и в ходе процесса приходит понимание существования различий при выполнении процедуры вставки звеньев, а именно, для элементов, имеющих две общие с рисунком точки, либо одну общую точку – левую или правую, либо при отсутствии общих точек. Конечно, динамику процесса при подготовке презентаций можно реализовать средствами, поддерживающими мультипликацию, но задача преподавателя в данном случае несколько иная – это изучение структур данных, алгоритмов их построения и использования, а не мультимедийные приложения.

В режиме онлайн одна презентация используется в течение нескольких занятий во время работы над программным проектом. Для эффективного использования презентации в ходе практических занятий по программированию полезно присутствие следующих компонент:

- точная постановка задачи со стороны «заказчика»;
- определение и пояснение теоретических положений, необходимых для разработки проекта;
- обсуждение альтернативных вариантов решения;
- сравнение сложности по времени и количественные характеристики предлагаемых к использованию разных вариантов структур хранения;

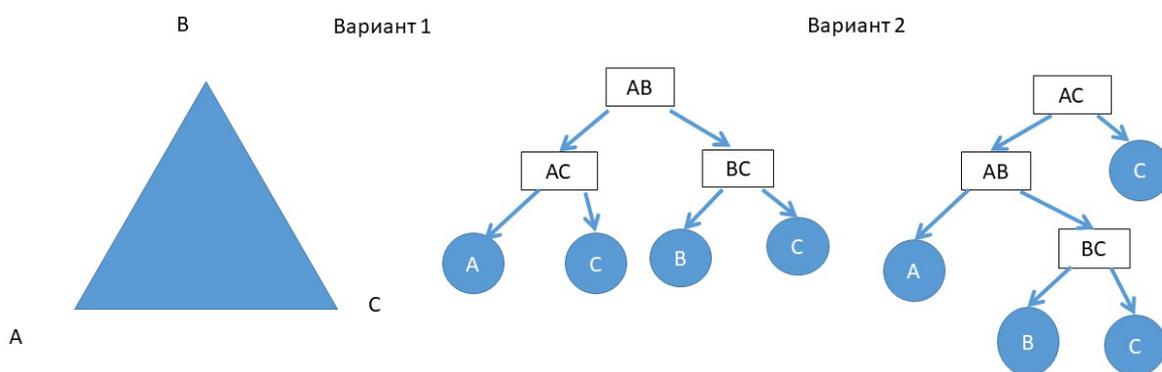


Рис. 6. Варианты плексов для треугольника

– различные варианты организации показа динамических процессов;

– обсуждение структуры программного продукта, который должен быть получен как результат разработки проекта;

– формирование четких требований к сдаче проекта;

– порядок сдачи работ и отчетов.

На каждом из этапов необходимо инициировать внимание студентов, чередуя слайды с текстовой информацией, имеющей визуальную структуру. При этом слайды не должны быть перегружены текстом, а слайды с «картинками» не должны превращать презентацию в забавный мультфильм.

Второй, не менее важной частью работы преподавателя программирования является совместное написание программ, реализующих разрабатываемый проект. Существует мнение, что при обучении программированию можно сформулировать постановку задачи, а затем само программирование перенести на самостоятельную работу. Такое мнение сформировалось исторически, и пока объем знаний и навыков, требуемый для программистов, был небольшим, такое мнение можно было считать оправданным [20]. Однако за последние годы в программировании появилось столько новых знаний и методов, что без квалифицированного «супервизора» – преподавателя студенту будет трудно не утонуть в потоке информации. В рамках дистанционного формата появляется сложная задача: необходимо организовать совместное программирование и удержать внимание группы. Нужно достигнуть понимания необходимости совместной работы. Конечно, студенты второго курса, изучающие «Алгоритмы и структуры данных», уже обладают некоторыми знаниями по программированию, но тем сложнее и интереснее должна быть организована работа преподавателя для удержания их внимания. Технически организовать совместную работу можно пере-

дачей права демонстрации экрана одному из студентов, который запускает среду программирования. Обычно остальные студенты также работают за своими компьютерами. Выбор «ведущего» как правило не вызывает проблем, и после проведения двух-трех занятий студенты понимают полезность этой функции, на эту роль появляется несколько желающих. В данной ситуации лидирующая роль преподавателя не должна сводиться к демонстрации кода, а должна состоять в своевременной помощи в преодолении сложностей в действиях «ведущего» и в вовлечении всей аудитории в процесс с учётом возможностей каждого. При этом преподаватель может поручить кому-то из студентов поиск в интернете описания возникающих ошибок или поиск описания синтаксиса используемых функций, так как при смене версий Visual Studio иногда возникают некоторые новшества. Это происходит в режиме «здесь и сейчас», и «ведущий» использует результаты поиска. Часть студентов должна отслеживать код, который набирает ведущий, с целью поиска логических ошибок, кому-то можно поручить отслеживать по разосланной всем презентации следующие шаги процесса. В этот момент благодаря режиму онлайн программный проект приобретает черты проекта в смысле Scrum.

Необходимо точно определить момент, начиная с которого студенты могут справиться самостоятельно без потери в уровне получаемой квалификации. Студенты даже из одной и той же группы имеют разную степень самостоятельности и разную самооценку. Всегда находятся те, кто, едва прослушав постановку задачи, пробуют самостоятельно завершить работу. Конечно, надо поощрять такую самостоятельность, но при этом контролировать результат, приветствовать успешные моменты и обсуждать недостатки, привлекая к этому остальную аудиторию.

При совместной работе происходит об-суждение и последовательности действий, и порядка тестирования новых элементов программирования. В каждом проекте должны быть новые знания из области программирования, которые в предыдущих задачах не встречались. В этом состоит очень большая невидимая часть внеауди-торной работы преподавателя, независимо от того, предъявляются ли результаты этой работы в аудитории или в дистанционном формате обучения. Например, в лаборатор-ной работе «Алгебра множеств» новое – разработка проектов с классами, техника тестирования класса в условиях консоли, переход интерфейса на WindowsForm и по-том решение задачи, постановка которой приведена выше.

В работе «Треугольные матрицы» но-вое – классы на шаблоне, использование новых инструментов на WindowsForm, рекурсия в объявлении полей класса и практически во всех методах, включая конструктор, например, запись вида `Tvector<Tvector<TM>> mtr`, как не частое яв-ление в программировании.

«Транслятор арифметических выраже-ний» – большой проект, в котором заложены многие важные, как теоретические, так и практические вопросы: понятие транслятора, процесс получения exe-кода, динамические структуры, таблицы переменных и области действия переменных, алгоритм Дейкстры, приоритеты операций, технология разработ-ки и тестирования проекта, состоящего из нескольких, более чем двух, классов, работа с формой, установка соответствия между динамической структурой таблицы и ее ви-зуальным представлением на форме.

«Поразрядная сортировка» при исполь-зовании динамической структуры «оче-редь» – небольшая работа, как сувенирная шкатулка, содержит красивую вещь: выбор структуры хранения для очереди, которая может дать кратный выигрыш по памяти.

Впервые появляются списки как распреде-ленная структура хранения, к которым студенты привыкают не быстро.

«Алгебра полиномов» – это аналитиче-ские преобразования на компьютере, це-лый раздел теоретического программиро-вания, выбор структуры хранения, которая может решить проблему 0-полинома, со-кратит сложность по времени, упростит алгоритмы; в режиме реального времени пошаговое тестирование циклического списка с контролем адресов памяти и их содержимым. Этот проект, наверно, един-ственный, который сначала до конца (кро-ме вывода на форму, она к этому моменту уже становится привычной) выполняется в условиях совместной работе группы.

Последние два проекта сложны как для представления материала, так и для реали-зации. Включают сложные алгоритмы со-здания структуры, обхода, копирования, сохранения в файлах, восстановления структуры, новые теоретические понятия: наследование, абстрактные классы, преоб-разование типов и так далее. Но эти проек-ты при большой и сложной базе, которая реализуется совместными усилиями, поз-воляют выполнить красивые приложения с использованием WindowsForm и в разных вариантах, у студентов появляется и воз-можность проявить свою индивидуаль-ность, и к этому времени они набирают достаточный уровень квалификации.

**Выводы.** Таким образом, проектный ме-тод изучения курса «Алгоритмы и структу-ры данных», разработанный и многократно опробованный в классическом режиме рабо-ты – практика, с доской и мелом, и лабора-торными работами в терминал-классе, – впервые был перенесен в дистанционную среду, где приобрел черты Scrum проекта.

Альтернатива доске и мелу в режиме он-лайн – подготовленные презентации. Пре-зентации как дидактический материал для

выполнения лабораторных работ [19; 21] должны включать следующие компоненты:

- точная постановка задачи со стороны «заказчика»;
- определение и пояснение теоретических положений, необходимых для разработки проекта;
- обсуждение альтернативных вариантов решения;
- сравнение сложности по времени и количественные характеристики предлагаемых к использованию разных вариантов структур хранения;
- различные варианты организации показа динамических процессов;
- обсуждение структуры программного продукта, который должен быть получен как результат разработки проекта;
- формирование четких требований к сдаче проекта;
- порядок сдачи работ и отчетов.

При проведении практических занятий и лабораторных работ в режиме онлайн особое внимание следует уделять удержанию внимания. Для этого необходимо чередовать слайды с текстовой информацией и слайды с «картинками», ставить вопросы, требующие ответа или выбора. Проектирование презентаций к занятиям вызывает сложности как технического, так и содержательного характера.

В рамках дистанционного формата появляется сложная задача: необходимо организовать совместное программирование

и удержать внимание группы. Нужно достигнуть понимания необходимости совместной работы. Активность и соучастие студентов в период выполнения проектов в режиме «онлайн» поддерживать непросто, психологическая нагрузка на преподавателя возрастает многократно.

Опыт работы показывает, что в группах с сильными студентами активная работа позволяет полностью выдерживать, график работ, иногда даже с опережением. Слабые группы целиком «висят» на преподавателе, самостоятельной работы у студентов практически не получается, хотя в коллективной совместной работе участвуют с удовольствием, проблем с выбором «ведущего» не возникает, они готовы обсуждать и помогать в решении общих задач. Но при этом график и сроки не выдерживаются, проекты приходится упрощать.

Переход на онлайн обучение явился сложным испытанием для всех участников процесса обучения, потребовал разработки новых методик обучения, изменения привычных понятий. Предложенную работу можно рассматривать как описание методики практической работы в режиме чрезвычайной ситуации, целью которой была попытка минимизации потерь, вызванных изменениями формы обучения. Работа проводилась на достаточно сложном фактическом материале, изучение которого и в классическом варианте представляет определенные проблемы.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Стронгин Р.Г., Гергель В.П. и др. Методы программирования 1. Программа общего курса и описания лабораторных работ. Часть 1. – Н.Новгород: ННГУ, 1999, 151 с.
2. Барышева И.В., Мееров И.Б., Сысов А.В., Шестакова Н.В. Лабораторный практикум (по программе «Алгоритмы и структуры данных»): учебно-методическое пособие. 2017. – 105с. Фонд образовательных электронных ре-

сурсов ННГУ №1438.17.06 [http://www.unn.ru/books/met\\_files/Pract\\_ADS.pdf](http://www.unn.ru/books/met_files/Pract_ADS.pdf)

3. Agile Manifesto. [Электронный ресурс] – URL: <https://www.agilealliance.org/agile101/> (дата обращения: 04.07.2021)

4. Методология управления проектами SCRUM. [Электронный ресурс] – URL: <https://ru.wikipedia.org/wiki/SCRUM> (дата обращения: 04.07.2021)

5. Обзор Agile. Что это: методология, метод или философия? [Электронный ресурс] – URL: <https://scrumtrek.ru/blog/agile-scrum/4029/metodologiya-agile/> (дата обр.: 04.07.2021)
6. Записи прямых трансляций около 700 уроков физики в специализированном физическом классе Ришельевского лицея города Одессы, [Электронный ресурс]– URL: <https://www.youtube.com/channel/UCSdDqsIYf9v5UEWTNda1YBw> (дата обр.: 04.07.2021)
7. Основы компьютерных наук и искусства программирования CS50 на русском (Гарвардский курс по основам программирования) [Электронный ресурс] //– URL: <https://www.youtube.com/playlist?list=PLawfWYMUziZqyUL5QDLVbe3j5BKWj42E5> (дата обращения: 04.07.2021)
8. **Amber Settle, Arto Vihavainen, and Craig S. Miller.** Research Directions for Teaching Programming Online [Электронный ресурс] – URL: [https://www.researchgate.net/publication/269095318\\_Research\\_Directions\\_for\\_Teaching\\_Programming\\_Online](https://www.researchgate.net/publication/269095318_Research_Directions_for_Teaching_Programming_Online) (дата обращения: 04.07.2021)
9. **Баженова И.В.** Методика проективно-рекурсивного обучения программированию студентов математических направлений подготовки: дисс. канд. пед. наук: 13.00.02. – Красноярск, 2015. – 159 с.
10. **Бобков В.В.** Реализация дифференцированного подхода в обучении студентов программированию с помощью дидактического технологического комплекса: дисс. канд. пед. наук: 13.00.02. – Красноярск, 2005. – 244 с.
11. **Бордюгова Т.Н.** Методические подходы к формированию компетенций в области программирования на основе реализации индивидуальной траектории обучения: на примере подготовки бакалавров по направлению «Педагогическое образование, профиль «Информатика»: дисс. канд. пед. наук: 13.00.02. – Москва, 2011.
12. **Калитина В.В.** Формирование программно-алгоритмической компетентности бакалавров информационных направлений при обучении программированию: автореф. дисс. канд. пед. наук: 13.00.02. – Красноярск, 2015.
13. **Кирьякова И.В.** Задачный подход в обучении основам программного обеспечения для развития продуктивного мышления будущего учителя информатики: дисс. канд. пед.: 13.00.02. – Омск, 2007. – 179 с.
14. **Гергель В.П.** Презентации по курсу «Алгоритмы и структуры данных», Нижегородский университет, 2001-2019гг. [Электронный ресурс] URL: <http://www.itmm.unn.ru/most/prezentatsii-po-kursu-algoritmy-i-strukturny-dannyh/> (дата обращения: 04.07.2021)
15. **Барышева И.В., Козлов О.А.** Формирование структурного мышления школьников в процессе обучения программированию в рамках школьного курса информатики // «Вопросы современной науки»: колл. науч. моногр. [под ред. Н.Р. Красовской]. – М.: Интернаука, 2016. Т.14. ISBN: 978-5-990855144 – С. 112-129.
16. **Барышева И.В., Козлов О.А.** Проектный метод в изучении основ программирования студентами профильных специальностей // Педагогическая информатика. 2016. №4. С. 78–83. ISSN: 2070-9013
17. **Барышева И.В., Козлов О.А.** Изменение роли и объема самостоятельной работы студентов в условиях дистанционного изучения программирования // Педагогическая информатика. 2020. №4. С. 53-58. ISSN: 2070-9013
18. Compiling Techniques by Bob Hopgood, 1969 [Электронный ресурс] URL: <http://www.chilton-computing.org.uk/acl/literature/books/compilingtechniques/p002.htm> (дата обращения: 04.07.2021)
19. **Барышева, И.В.** Практика по курсу «Алгоритмы и структуры данных», 2021 [Электронный ресурс] URL: <https://e-learning.unn.ru/course/view.php?id=8627> (дата обращения: 04.07.2021)
20. **Bennedsen J., Caspersen M.E. (2008)** Exposing the Programming Process. In: Bennedsen J., Caspersen M.E., Kölling M. (eds) Reflections on the Teaching of Programming. Lecture Notes in Computer Science, vol 4821. Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-540-77934-6\\_2](https://doi.org/10.1007/978-3-540-77934-6_2)
21. **Барышева И.В., Козлов О.А.** Использование электронных ресурсов в методике изучения программирования // Электронные ресурсы в непрерывном образовании: Труды VI Междунар. науч.-метод. симпозиума ЭРНО-2017 (Адлер 24-27 сентября 2017 г.) Редкол.: В.И. Мареев, О.А. Козлов, М.И. Коваленко. Ростов-на-Дону; Таганрог: Изд-во Южного федерального университета. – 2017. ISBN 978-5-9275-2489-1–С. 64-68.

## REFERENCES

1. **Strongin R.G., Gergel' V.P. i dr.** Metody programmirovaniya 1. Programma obshhego kursa i opisaniya laboratornyh rabot. Chast' 1. – N. Novgorod: NNGU, 1999, 151 s.
2. **Barysheva I.V., Meerov I.B., Sysoev A.V., Shestakova N.V.** Laboratornyj praktikum (po programme «Algoritmy i struktury dannyh»): uchebno-metodicheskoe posobie. 2017. – 105s. Fond obrazovatel'nyh jelektronnyh resursov NNGU №1438.17.06 [http://www.unn.ru/books/met\\_files/Pract\\_ADS.pdf](http://www.unn.ru/books/met_files/Pract_ADS.pdf)
3. Agile Manifesto. [Jelektronnyj resurs] – URL: <https://www.agilealliance.org/agile101/> (data obrashhenija: 04.07.2021)
4. Metodologija upravljenija proektami SCRUM. [Jelektronnyj resurs]– URL: <https://ru.wikipedia.org/wiki/SCRUM> (data obrashhenija: 04.07.2021)
5. Obzor Agile. Chto jeto: metodologija, metod ili filosofija? [Jelektronnyj resurs] – URL: <https://scrumtrek.ru/blog/agile-scrum/4029/metodologiya-agile/>(data obrashhenija: 04.07.2021)
6. Zapisi prjamyh transljacij okolo 700 urokov fiziki v specializirovannom fizicheskom klasse Rishel'evskogo liceja goroda Odessa, [Jelektronnyj resurs] – URL: <https://www.youtube.com/channel/UCSdDqsIYf9v5UEWTNda1YBw> (data obr.: 04.07.2021)
7. Osnovy komp'juternyh nauk i iskusstva programmirovaniya CS50 na ruskom (Garvardskij kurs po osnovam programmirovaniya) [Jelektronnyj resurs]– URL: <https://www.youtube.com/playlist?list=PLawfWYMUziZqyUL5QDLVbe3j5BKWj42E5> (data obrashhenija: 04.07.2021)
8. **Amber Settle, Arto Vihavainen, and Craig S. Miller.** Research Directions for Teaching Programming Online [Jelektronnyj resurs] – URL: [https://www.researchgate.net/publication/269095318\\_Research\\_Directions\\_for\\_Teaching\\_Programming\\_Online](https://www.researchgate.net/publication/269095318_Research_Directions_for_Teaching_Programming_Online) (data obrashhenija: 04.07.2021)
9. **Bazhenova I.V.** Metodika proektivno-rekursivnogo obuchenija programmirovaniyu studentov matematicheskikh napravlenij podgotovki: diss. kand. ped. nauk: 13.00.02. – Krasnojarsk, 2015. – 159 s.
10. **Bobkov V.V.** Realizacija differencirovannogo podhoda v obuchenii studentov programmirovaniyu s pomoshh'ju didakticheskogo tehnologicheskogo kompleksa: diss. kand. ped. nauk: 13.00.02. – Krasnojarsk, 2005. – 244 s.
11. **Bordjugova T.N.** Metodicheskie podhody k formirovaniyu kompetencij v oblasti programmirovaniya na osnove realizacii individual'noj traektorii obuchenija: na primere podgotovki bakalavrov po napravleniju «Pedagogicheskoe obrazovanie, profil' «Informatika»: diss. kand. ped. nauk: 13.00.02. – Moskva, 2011.
12. **Kalitina V.V.** Formirovanie programmno-algoritmicheskoy kompetentnosti bakalavrov informacionnyh napravlenij pri obuchenii programmirovaniyu: avtoref. diss. kand. ped. nauk: 13.00.02. –Krasnojarsk, 2015.
13. **Kir'jakova I.V.** Zadachnyj podhod v obuchenii osnovam programmno obespechenija dlja razvitija produktivnogo myshlenija budushhego uchitelja informatiki: diss. kand. ped.: 13.00.02. – Omsk, 2007. – 179 s.
14. **Gergel' V.P.** Prezentacii po kursu «Algoritmy i struktury dannyh», Nizhegorodskij universitet, 2001-2019gg. [Jelektronnyj resurs] URL: <http://www.itmm.unn.ru/most/prezentatsii-po-kursu-algoritmy-i-struktury-dannyh/> (data obrashhenija: 04.07.2021)
15. **Barysheva I.V., Kozlov O.A.** Formirovanie strukturnogo myshlenija shkol'nikov v processe obuchenija programmirovaniyu v ramkah shkol'nogo kursa informatiki // «Voprosy sovremennoj nauki»: koll. nauch. monogr. [pod red. N.R. Krasovskoj]. – M.: Internauka, 2016. T.14. ISBN: 978-5-990855144 – S. 112-129.
16. **Barysheva I.V., Kozlov O.A.** The design method in the study of the fundamentals of programming by students of profession-oriented specialties. *Pedagogical informatics*. 2016. No. 4. Pp. 78–83 ISSN: 2070-9013
17. **Barysheva I.V., Kozlov O.A.** Changing the role and scope of independent work of students in the conditions of remote learning of programming. *Pedagogical informatics*. 2020. No. 4. Pp. 53-58 ISSN: 2070-9013
18. Compiling Techniques by Bob Hopgood, 1969 [Jelektronnyj resurs] URL: <http://www.chilton-computing.org.uk/acl/literature/books/compilingtechniques/p002.htm> (data obrashhenija: 04.07.2021)
19. **Barysheva, I.V.** Praktika po kursu «Algoritmy i struktury dannyh», 2021 [Jelektronnyj resurs]URL: <https://e-learning.unn.ru/course/view.php?id=8627>(data obrashhenija: 04.07.2021)

20. **Bennedsen J., Caspersen M.E. (2008)** Exposing the Programming Process. In: Bennedsen J., Caspersen M.E., Kölling M. (eds) Reflections on the Teaching of Programming. Lecture Notes in Computer Science, vol 4821. Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-540-77934-6\\_2](https://doi.org/10.1007/978-3-540-77934-6_2)

21. **Barysheva I.V., Kozlov O.A** Ispol'zovanie jelektronnyh resursov v metodike izuchenija

programmirovanija // Jelektronnye resursy v nepreryvnom obrazovanii: Trudy VI Mezhdunar. nauch.-metod. simpoziuma JeRNO-2017 (Adler 24-27 sentjabrja 2017 g.) Redkol.: V.I. Mareev, O.A. Kozlov, M.I. Kovalenko. Rostov-na-Donu; Taganrog.: Izd-vo Juzhnogo federal'nogo universiteta. – 2017. ISBN 978-5-9275-2489-1–S. 64-68.

**Barysheva Irina V., Malkina Elena V., Kozlov Oleg A. Project method of studying programming by students of profile specialists in remote work conductions.** The article tackles practical and laboratory work within the framework of the Algorithms and Data Structures course aimed at the preparation of IT specialists at the full-time department in the context of the transition to online training during the COVID-19 pandemic. Particular attention is paid to the methodology of studying various data structures as a basis for information systems developers training. A methodology for conducting online classes using presentations is proposed. The main components of the presentation preparation are defined. Within the Algorithms and Data Structures course, students must complete seven laboratory works. Each laboratory work represents a software development project which spans across several study sessions. During each study session, students develop the structure of the project and gradually increase the number of classes representing subtasks of the major task. The article demonstrates how student involvement in collaborative programming gives the laboratory software projects the characteristics of Scrum projects. The project method is analyzed and some results of the proposed methodology application for conducting practical and laboratory classes are given.

ALGORITHMS; DISTANCE LEARNING; PROGRAM PROJECT; DESIGN METHOD; DATA STRUCTURES

Статья поступила в редакцию 30.05.2021; одобрена после рецензирования 16.07.2021; принята к публикации 28.09.2021.  
*The article was submitted 30.05.2021; approved after reviewing 16.07.2021; accepted for publication 28.09.2021*

---

**Citation:** Barysheva I.V., Malkina E.V., Kozlov O.A. Project method of studying programming by students of profile specialists in remote work conductions. *Teaching Methodology in Higher Education*. 2021. Vol. 10. No 38. P. 40–55. DOI: 10.18720/HUM/ISSN 2227-8591.38.04