

УДК 004.4

ПРИМЕНЕНИЕ СРЕДЫ LAZARUS В ОБРАЗОВАНИИ

Е.К. Суняйкина, студентка 5 гр. МДИ-110, V курс
Научный руководитель: В.И. Сафонов, к.ф.-м.н., доцент
Мордовский государственный педагогический институт
имени М. Е. Евсевьева
г. Саранск

Алгоритмизация и программирование являются важными содержательными линиями школьного курса информатики, в связи с чем актуален вопрос подготовки учителя информатики к обучению школьников этим линиям. Изучение алгоритмизации в школьной информатике может иметь два целевых аспекта: первый – развивающий аспект, под которым понимается развитие алгоритмического (операционного) мышления учащихся; второй – программистский аспект, под которым понимают развитие навыков составления учебных программ. Первый аспект связан с усилением фундаментальной компоненты курса информатики. Ученикам даются представления о следующих понятиях: что такое языки программирования, что представляет собой программа, как создается программа на разных языках программирования и в различных средах. Второй аспект носит профориентационный характер. Профессия программиста является достаточно распространенной и престижной. Изучение программирования в рамках школьного курса позволяет ученикам испытать свои способности в данной деятельности.

Перед началом обучения учителю необходимо выбрать язык программирования с учетом интересов учащихся, их направленности и структуры образовательного процесса в школе. Безусловно, в начале необходимо изучать алгоритмический язык, являющийся основой для формирования алгоритмического мышления, для понимания и правильного построения алгоритмических конструкций. Но в последнее время уже наблюдается тенденция перехода обучения от алгоритмических языков к объектно-ориентированным языкам программирования, что показывает об изменении общего подхода к преподаванию программирования в школе.

Обучение теме «Алгоритмизация» предполагает следующую структуру: в начале изучения учащиеся должны ознакомиться с основными алгоритмическими конструкциями, командами языка, правилами описания объектов языка программирования, структурой программы и правилами написания. Учащимся необходимо сначала сформировать навыки написания простейших программ с использованием алгоритмических конструкций и основных объектов языка. Данные навыки можно развить при изучении школьного алгоритмического языка программирования, а затем перейти к изучению простых и далее более сложных методов программирования. Необходимо на началь-

ной ступени изучения алгоритмизации рассмотреть такую форму представления алгоритма как блок-схема, так как она помогает обеспечить структурное представление алгоритма, что играет немаловажную роль при формировании алгоритмического мышления.

Роль алгоритмов в науке и образовании очень велика. Особое значение имеют алгоритмы, накопленные в математике [1], потому что математика пронизывает другие науки и ее богатство является богатством всех наук. Уже довольно давно ученые и инженеры заметили, что если удалось получить алгоритм решения какой-нибудь задачи, то можно создать машину, которая решала бы эту задачу, т. е. можно автоматизировать ее решение. Алгоритмы являются: 1) формой изложения научных результатов; 2) руководством к действию при решении уже изученных проблем и, как следствие: 3) средством, позволяющим экономить умственный труд; 4) необходимым этапом при автоматизации решения задач; 5) средством (инструментом), используемым при исследовании и решении новых проблем (особенно это касается математических алгоритмов); 6) одним из средств обоснования математики одним из средств описания сложных процессов.

Алгоритмы составляют важную часть каждой науки, но не исчерпывают ее содержания. Не менее важны, конечно, понятия и определения, входящие в данную науку, установленные ею факты (в математике — это доказанные теоремы), выработанный наукой подход к изучаемым объектам и явлениям.

Большая ценность алгоритмов обуславливает интерес к ним. Естественно, что специалисты каждой отрасли науки и техники все время ищут алгоритмы решения различных задач. Каждый новый алгоритм немедленно включается в «золотой фонд» науки. При этом интересны как новые алгоритмы, так и алгоритмы для решения вновь поставленных проблем. Алгоритмы реализуются языками программирования. Одним из распространенных семейств алгоритмических языков программирования является язык Pascal. Существуют различные среды Pascal, обратимся к свободным.

Free Pascal Compiler (FPC) – это свободно распространяемый компилятор языка Pascal с открытыми исходными кодами, распространяется на условиях GNU General Public License (GNU GPL). Он совместим с Borland Pascal 7.0 и Object Pascal Delphi, но при этом обладает рядом дополнительных возможностей, например, поддерживает перегрузку операторов. Free Pascal Compiler имеет свою собственную интегрированную среду разработки. Применяется также аббревиатура IDE (Integrated Development Environment). Среда имеет текстовый интерфейс очень похожий на интерфейс Turbo Pascal 7.0. Однако со временем текстовые интерфейсы были практически полностью вытеснены так называемыми графическими интерфейсами, работать в которых значительно удобнее. В 1999 г. Клифф Байзмент, Шейн Миллер и Майкл А. Гесса написали графическую среду для бесплатного компилятора FPC. Проект получал название Lazarus. На сегодняшний день следует признать, что идея оказалась весьма плодотворной потому, что среда существует и развива-

ется и поныне. Она нашла свое место в учебном процессе и ее освоение позволит ученику осваивать язык Pascal, решать с его помощью различные задачи, что и определяет актуальность работы.

Целью нашей работы являлось построение элективного курса по изучению технологии реализации основных алгоритмических структур в среде Lazarus. Для достижения указанной цели потребуется решить ряд задач: изучить и описать интерфейс Lazarus; рассмотреть типы данных, операторы ввода/вывода, операции и выражения Free Pascal; изучить применение операторов Free Pascal управления вычислительным процессом; разработать элективный курс «Программирование в Lazarus».

Lazarus – это стабильная, богатая возможностями интегрированная среда разработки для создания самостоятельных графических и консольных приложений, полностью поддерживающая синтаксис Pascal. Lazarus создавался по образу и подобию Delphi, поэтому они так сильно похожи. Тем не менее, у них есть принципиальные различия.

1. Это два продукта совершенно с разными идеологиями.

Borland Delphi – это полностью коммерческий продукт. Чтобы им пользоваться, необходимо его приобрести. Минимальная стоимость Delphi, на сегодняшний день, составляет 28000 рублей. Но этого мало, так как часто требуется система управления базами данных (СУДБ). Производитель рекомендует использовать СУБД InterBase. Это еще 6000 рублей. В итоге получается, что для того чтобы укомплектовать место современного разработчика программного обеспечения, требуется минимум 34000 рублей, не говоря об оснащении школ. Lazarus – это полная идеологическая противоположность Delphi. Данный продукт полностью свободен, что позволяет использовать его в любых целях. Lazarus – это плод труда большого общества программистов со всего мира. Российские программисты занимают одно из ключевых мест. Любой может участвовать в этом проекте и безвозмездно пользоваться им, без опаски стать нелегалом в компьютерном мире.

2. Серьезное техническое различие. Для того чтобы установить компонент Lazarus нужно перекомпилировать всю визуальную систему программирования. Компонент плотно интегрируется в среду, становится непосредственной ее частью. В Delphi установка нового компонента происходит на много удобнее. Не надо перекомпилировать все, нужно только перекомпилировать модули компонента.

3. Delphi – это Windows-инструмент. С его помощью можно разработать приложения только для этой операционной системы. При использовании получившейся программы на других системах необходимо использование различных ухищрений – эмуляторов, сред. Еще недавно существовал проект Borland Kylix для Linux – Delphi для Linux. Однако проект был закрыт. С Lazarus дело обстоит иначе. Полная интеграция среды и компилятора Free Pascal позволяет создавать кроссплатформенные приложения, т.е. приложения, которые могут выполняться на различных платформах. К их числу относятся Mac, Linux, Windows. Разработав программу для одной операционной

системы, ее можно с легкостью перекомпилировать для другой, тем самым рынок реализации программы увеличивается.

Процесс создания программы в Lazarus состоит из двух этапов: формирования внешнего вида программы, её интерфейса; написания программного кода на языке программирования Free Pascal, заставляющего работать элементы интерфейса.

Нами была разработана программа элективного курса «Программирование в Lazarus». Рассмотрим ее содержание.

1. Среда программирования Lazarus
 - 1.1. Интерфейс среды программирования Lazarus
 - 1.2. Создание приложения в среде Lazarus
2. Основы языка Паскаль.
 - 2.1. Данные. Типы данных
 - 2.2. Организация ввода-вывода. Оператор присваивания.
 - 2.3. Правила записи математических выражений. Операции и выражения. Стандартные функции.
3. Реализация основных алгоритмических структур.
 - 3.1. Алгоритмы линейной структуры
 - 3.2. Алгоритмы разветвляющейся структуры. Основные понятия математической логики.
 - 3.3. Решение задач «Организация ветвлений в программах»
 - 3.4. Циклический алгоритм. Виды циклов.
 - 3.5. Решение задач «Программирование циклических алгоритмов».
4. Массивы
 - 4.1. Одномерные массивы: описание и задание элементов. Действия над одномерными массивами.
 - 4.2. Поиск, замена в одномерном массиве. Сортировка, способы сортировки.
 - 4.3. Решение задач «Массивы» в среде программирования Lazarus.

Разработанный элективный курс «Программирование в Lazarus» апробирован в ходе работы в Муниципальном общеобразовательного учреждения «Пензятская средняя общеобразовательная школа» Лямбирского муниципального района Республики Мордовия. Результаты работы используются при проведении занятий со студентами по курсам «Информационные технологии в образовании», «Программирование» и «Свободное программное обеспечение в образовании». Также результаты работы докладывались на конференциях различных уровней.

Список литературы:

1. Сафонов В.И. Методы математики в изучении школьной информатики // Ученые записки ИИО РАО. – 2014. – № 52. – С. 23–33.
2. Электронные УМК по информатике [Электронный ресурс]. Адрес доступа: <http://e-umk.lbz.ru/BookList.html?Subject=Информатика>.

3. Алексеев, Е. Р. Самоучитель по программированию на Free Pascal и Lazarus // Донецк: ДонНТУ, Технопарк ДонНТУ УНИТЕХ, 2009. – 320 с.