

МЕТОДИЧЕСКИЕ ОСОБЕННОСТИ ПРЕПОДАВАНИЯ КУРСА «ПРОГРАММИРОВАНИЕ»

Н.А. Карпович, Т.А. Карпович

Белорусский государственный университет, ФПМИ, кафедра технологии программирования
г. Минск, пр. Независимости, 4, г. Минск, Беларусь
телефон(ы): 2095273; e-mail: karpovichna@bsu.by

Объектно-ориентированное программирование стало основной технологией разработки программных систем. В обучении этой технологии отводится существенное место. Остаются споры лишь о том, сразу или не сразу начинать обучение с введения классов и объектов. Важную роль в обучении играет язык программирования и среда разработки. На роль первого языка претендентом является язык C++, можно использовать среду Delphi, появились более молодые конкуренты - Java и C#.

В докладе рассматриваются вопросы преподавания курса «Программирование» на языке C++ в среде Visual Studio .NET с использованием встроенного языка Ассемблера.

Ключевые слова: C++, Ассемблер, программирование, лабораторные.

Дисциплина «Программирование» ориентирована на обучение студентов базовым знаниям, умениям и навыкам в области программирования. Изучаемые темы базируются на использовании современных информационных технологий, новейшего программного и технического обеспечения компьютеров.

Основная цель при изучении программирования: подготовить специалиста, умеющего проектировать эффективные алгоритмы решения поставленной задачи, выбирать наиболее подходящие структуры данных, программные и технические средства его реализации и с учетом операционного окружения разрабатывать программные приложения, отвечающие современным требованиям и новейшим компьютерным технологиям. Кроме этого, нужно дать студентам базу, необходимую для усвоения материала последующих учебных дисциплин в области информатики, и сформировать составную часть банка знаний, необходимого студентам для успешной дальнейшей работы.

При изложении курса важно показать возможности использования инструментария программирования при решении прикладных задач, возникающих в различных областях науки, техники, экономики и производства.

В соответствии с типовым учебным планом специальности программа предусматривает для изучения дисциплины 374 аудиторных часа, в том числе лекции – 154 часа, лабораторные занятия – 220 часов.

Курс читается в течение трех семестров и начинается со знакомства с языком C++ в среде Visual Studio .NET.

При изучении первого модуля рассматриваются основы программирования на языке C++: синтаксис и семантика, основные управляющие структуры и операторы, подпрограммы, организация ввода-вывода, структурированные данные, способы упорядочения и поиска информации. При решении практических задач анализируются основные этапы разработки и сопровождения программ, методы тестирования и отладки программ, типы ошибок и их обработка на этапах проектирования, трансляции, выполнения.

Студенты решают задачи по следующим темам: скалярные типы данных и управляющие операторы, обработка последовательностей, одномерные и двумерные массивы, символы и строки, текстовые файлы, функции, рекурсия, указатели и динамические структуры данных.

Во втором модуле изучаются принципы функционирования микропроцессоров, архитектура и структурная схема компьютера, представление информации в оперативной памяти компьютера, основные сведения о машинном языке. Рассматривается структура программы на языке Ассемблера, директивы сегментации, общая структура машинных команд, режимы адресации, форматы команд и система команд, организация взаимосвязей между модулями, написанными на языках высокого уровня и на языке Ассемблера.

Третий модуль предназначен для изучения объектно-ориентированного программирования на языке C++. При изучении модуля рассматриваются абстрактные типы и классы, наследование, множественное наследование, полиморфизм и виртуальные функции, абстрактные классы, объектная модель ввода-вывода, потоки ввода-вывода, обработка исключительных ситуаций, шаблоны функций и шаблоны классов, библиотеки стандартных классов (контейнеры, итераторы, функциональные объекты, алгоритмы).

В четвертом модуле изучается среда программирования и платформа .NET. Рассматриваются основные архитектурные концепции среды и ее возможности, архитектурные решения и возможные подходы к решению таких задач, как распределение и управление памятью, обработка ошибок, повторного использования кода, позднего связывания, динамического определения типов и т.п. Возможны различные подходы при изучении языка программирования. Предпочтение отдано платформо-независимым языкам программирования, именно языку программирования Java. В качестве среды разработки

используется Visual Studio .NET и Eclipse, изучаются основы синтаксиса и конструкций, реализация ООП в языке Java, обзор существующих пространств имен, возможности, предоставляемые библиотеками и средой .NET. Четвертый модуль сразу решает две задачи: изучаются классы элементов управления и разрабатываются законченные проекты. На практических занятиях проектируются и разрабатываются приложения с оконным интерфейсом, используются доступные GUI-библиотеки.

Лабораторные занятия курса "Программирование" – это универсальное средство обучающего воздействия, поскольку именно в деятельности и происходит становление и развитие студента.

Решение задач целочисленной арифметики формирует навыки разработки простейших алгоритмов, позволяет выделить основные классы таких задач, определить их сложность и преемственность в процессе решения, признаков усваивания материала.

Использование стандартных функций для решения вычислительных задач и выполнение сравнительного анализа результатов выполнения разработанных алгоритмов способствуют проверке правильности решения задач, позволяют самостоятельно найти ошибки.

Решение задач обработки массивов, матриц, структур данных, особенности реализации динамических структур данных развивают навыки и умения разработки эффективных алгоритмов, оценки их сложности и скорости выполнения.

Задания на развитие алгоритмического мышления: рекуррентные соотношения, рекурсии и динамическое программирование, сравнительный анализ итерационных и рекурсивных алгоритмов, разбиение задачи на подзадачи, анализ сложности алгоритмов.

Обработка текстовой информации, использование стандартных методов поиска, преобразования и редактирования, хранения данных на внешних носителях требуют от студентов знаний функций обработки файлов и потоков.

Использование языка Ассемблера позволяет написать более эффективную программу, чем та, что генерируется трансляторами с языков программирования высокого уровня. Для программ на ассемблере характерно использование меньшего количества команд, что позволяет увеличить скорость и уменьшить размер программы, обеспечить максимальное использование возможностей конкретной платформы, создавать более эффективные программы с меньшими затратами ресурсов. позволяет лучше понять и язык Си++, изучить методы связи программ, написанных на разных языках программирования.

Модификация разработанных программ с использованием принципов объектно-ориентированного программирования выполняется во втором семестре.

Разработка классов с использованием перегрузки операций, возможностей наследования, виртуальных функций, абстрактных классов с использованием множественного и виртуального наследования, шаблонов функций и шаблонов классов.

Использование библиотек стандартных классов: контейнеры, итераторы, функциональные объекты, классов

элементов управления позволяет разработать законченные проекты для реализации практических задач.

На учебной практике на 1 курсе отрабатываются методы и приемы программирования на языке C++ с использованием языка Ассемблер и приемы программирования задач с использованием объектно-ориентированного программирования на базовом языке C++, а также с использованием библиотеки шаблонов STL, которая вошла в стандарт языка и отражает современный уровень технологии программирования.

На учебной практике на 2 курсе рассматриваются структура и принципы построения приложений на VC++, с использованием библиотеки STL, MFC, приложений и апплетов на Java, практическая работа с ними. Целью практики является изучение структуры и организации приложений, алгоритмов, управлению памятью и файлами, использование готовых библиотек. Для практической работы используются средства Windows. Усвоение материала курса позволит студентам эффективно разрабатывать приложения, используя интерфейсы прикладных программ операционных систем.

Специально разработанные задания позволяют подготовить специалиста, умеющего проектировать эффективные алгоритмы решения поставленной задачи, выбирать наиболее подходящие структуры данных, программные и технические средства его реализации и с учетом операционного окружения разрабатывать программные приложения, отвечающие современным требованиям и новейшим компьютерным технологиям.

Проверка усвоения теоретического материала осуществляется в системе "E-University". Авторами также подготовлены методические материалы к лекционному курсу в виде презентации PowerPoint и электронные методические разработки лабораторных работ.

ЛИТЕРАТУРА

- [1] Павловская, Т. А. C/C++. Программирование на языке высокого уровня : учебник для студ. вузов, обуч. по напр. "Информатика и вычислительная техника" / Т. А. Павловская. - Санкт-Петербург : Питер, 2006. - 460с.
- [2] Юров В. Assembler : [Учебник] / В.Юров. - СПб. и др. : Питер, 2000. - 623с.
- [3] Зубков С.В. Assembler для DOS, Windows и UNIX для программистов. - Издательство: Питер, 2004. - 608 с.
- [4] Филингенов К.Г. Win32. Основы программирования. -- Издательство ДИАЛОГ-МИФИ, 2002. - 416 с.
- [5] Хорстманн Кей С., Корнелл Гари. Java 2. Библиотека профессионала, том 1. Основы, 7-е изд.: Пер. с англ. - М.: Издательский дом «Вильямс», 2007. - 896 с.: ил. - Парал. тит. англ.
- [6] Хорстманн Кей С., Корнелл Гари. Java 2. Библиотека профессионала, том II. Тонкости программирования, 7-е изд.: Пер. с англ. - М.: Издательский дом «Вильямс», 2007. - 1168 с.: ил. - Парал. тит. англ.