

МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ И ИТ-ИНФРАСТРУКТУРА ДЛЯ ПРЕПОДАВАНИЯ НИЗКОУРОВНЕГО ПРОГРАММИРОВАНИЯ

Маевский А.Э., Пеленицын А.М.

Южный федеральный университет, apel@sfedu.ru

В 2009/10 учебном году на факультете математики, механики и компьютерных наук Южного федерального университета был введён новый учебный курс «Микропрограммирование» для магистрантов направления «Информационные технологии» и студентов пятого курса специальности «Прикладная математика и информатика» (специализация «Математическая кибернетика»). Было решено использовать лабораторный практикум по этому курсу для обучения низкоуровневому программированию на примере языка ассемблера для архитектуры x86.

Задачи курса. Приводятся доводы в пользу выбора языка ассемблера для архитектуры x86. Среди прочего это: наличие учебно-методических материалов (актуальных при обучении старшекурсников, когда большая часть учебного процесса смещается в сферу самостоятельной работы), востребованность «устаревшего» языка ассемблера x86 для прикладных программистов, занимающихся оптимизацией программ, написанных на высокоуровневых языках, решение задач системных программистов в самых разных областях: от реверс-инжиниринга до разработки компиляторов и программирования встраиваемых устройств, которые хотя и поддерживают собственные наборы инструкций, отличные от x86, следуют общей парадигме низкоуровневого программирования, отработанной на архитектуре x86.

Важным является согласованность курса с другими предметами в учебном плане и ведущейся на факультете научной работой. Во-первых, данный курс, предшествует курсу «Методы разработки оптимизирующих компиляторов» в магистратуре направления «Информационные технологии», где могут быть использованы заложенные в нашем курсе знания и навыки; во-вторых, он читается одновременно с курсом по истории информатики [1]: мы полагаем, что эволюция архитектуры x86 и её программных интерфейсов является ярким примером основных тезисов этого курса по истории и заостряем внимание на соответствующих примерах. Наконец, на кафедре алгебры и дискретной математики разрабатывается крупный проект по автоматическому распараллеливанию программ, частью которого является оптимизирующий компилятор — его разработка затрагивает вопросы низкоуровневого программирования.

Методическое обеспечение. При построении курса решены ряд методических задач. Выбор между различными диалектами языка ассемблера влияет на возможности использования так называемых

макроассемблеров. Эта возможность была признана не соответствующей задачам курса, изложенным выше. Кроме абстракции базового языка ассемблера, достигаемой развитыми макросредствами, можно видеть аналогичную тенденцию в учебниках, изначально снабжающих студентов программными библиотеками, например, для решения рутинных задач ввода-вывода. Хотя подобный подход был обоснован ещё Б. Мейером [2], он не был принят на первом году чтения данного курса, чтобы не создавать жёсткого ограничения на структуру курса. Тем не менее, эта идея выглядит достаточно перспективной. Для обучения базовым синтаксическим элементам языка оказалось удобным использовать, в частности, задачи из электронного задачника М.Э. Абрамяна [3].

ИТ-инфраструктура. Наиболее приемлемой средой выполнения заданий по курсу была признана ОС DOS, которая, в отличие от современных ОС, предоставляет одну из наиболее тонких прослоек между программистом и аппаратным обеспечением. Для работы в этой системе использовалась свободная среда виртуализации VirtualBox, развёрнутая в дисплейных классах факультета, и свободный дистрибутив FreeDOS. В составе последнего можно найти достаточный выбор свободных линковщиков и ассемблеров, в том числе, поддерживающих «синтаксис MASM», наиболее адекватным с точки зрения имеющейся литературы на русском языке. Самым приемлемым способом обеспечения связи ОС с «внешним миром» оказалось сетевое соединение по протоколу SFTP — для этих целей на сервере факультета был запущен SFTP-сервер. Мы использовали систему Moodle, развёрнутую по адресу edu.mmcs.sfedu.ru в сети факультета, а также информационно-образовательный портал по направлению «Информационные технологии» [4].

Литература.

1. Брагилевский В.Н. Курс истории информатики в магистратуре по направлению «Информационные технологии» // Труды IV международной научно-практической конференции «Современные информационные технологии и ИТ-образование». Москва. 2009.
2. Meyer B. The outside-in method of teaching introductory programming // In Manfred Broy and Alexandre V. Zamulin eds., Ershov Memorial Conference, LNCS vol. 2890, pp. 66–78. Springer, 2003.
3. Абрамян М.Э. Programming Taskbook 4 — электронный задачник по программированию на языках Pascal, C, Visual Basic // Научно-методическая конференция «Современные информационные технологии в образовании: Южный

- Федеральный округ», Ростов н/Д., 13–14 мая 2004 г. Тезисы докладов. — Ростов н/Д.: Изд-во «ЦВВР», 2004. — С. 23–26.
4. Брагилевский В.Н., Михалкович С.С., Пеленицын А.М. Разработка информационно-образовательного портала по направлению «Информационные технологии» // Научно-методическая конференция «Современные информационные технологии в образовании: Южный Федеральный округ», 23–25 октября 2008 г. Тезисы докладов. — Ростов н/Д.: Изд-во «ЦВВР», 2008. — С.48–49.