

Андрей Викторович Столяров
Елена Игоревна Большакова
Наталья Валерьевна Баева

InteLib Lisp в обучении программированию на Лиспе

© А. В. Столяров, Е. И. Большакова, Н. В. Баева, 2006

Настоящий документ представляет собой тезисы доклада, сделанного на конференции «Свободное программное обеспечение в высшей школе» 28 января 2006 года. Просьба при цитировании ссылаться на версию, опубликованную в сборнике тезисов докладов конференции с использованием следующей библиографической информации:

А. В. Столяров, Е. И. Большакова, Н. В. Баева. InteLib Lisp в обучении программированию на Лиспе // Тезисы докладов конференции «Свободное программное обеспечение в высшей школе», Переславль, январь 2006 г.

Для пользователей L^AT_EX приводится библиографическая информация в формате bibtex:

```
@INCOLLECTION{stolyarov:teaching2006,  
  AUTHOR={A. В. Столяров and Е. И. Большакова and Н. В. Баева}},  
  TITLE="Мультипарадигмальный подход  
        к преподаванию программирования и  
        роль свободного ПО",  
  BOOKTITLE = "Тезисы докладов конференции  
              {{{<Свободное программное обеспечение в высшей школе>}}}",  
  MONTH = "январь",  
  YEAR = "2006",  
  PAGES = "15--17",  
  ADDRESS = "Переславль"  
}
```

InteLib Lisp в обучении программированию на Лиспе

А. В. Столяров, Е. И. Большакова, Н. В. Баева
ВМиК МГУ им. Ломоносова

1 Обучение функциональному программированию: выбор языка

Изучение основ функционального программирования [1] чрезвычайно важно при обучении студентов программистских специальностей и является необходимой компонентой образования в области computer science. Функциональное программирование может изучаться на основе многих современных языков программирования (Scheme, OCaml, Haskell и т.п.) С нашей точки зрения, наиболее удобным для изучения основ функционального программирования является классический язык Лисп, позволяющий освоить основные приемы функционального программирования, такие как рекурсия и функции высших порядков, а также гетерогенные структуры данных.

Перед преподавателями встает проблема выбора диалекта языка и системы программирования. Такие диалекты Лиспа, как Common Lisp [2], громоздки; в них включены, в числе прочих, и императивные конструкции, что провоцирует императивный стиль программирования и мешает освоению функциональных приемов. Другие диалекты (например, MuLisp) не столь громоздки, однако сам факт использования при обучении заведомо устаревшего программного обеспечения создает у студентов ложное впечатление устарелости языка Лисп как такового.

С другой стороны, многие современные системы программирования на Лиспе для MS Windows слабо подходят для обучения Лиспу, поскольку отвлекают внимание студента на вопросы, связанные с синтезом пользовательского интерфейса (это отмечается, например, в статье [3]).

Таким образом, для обучения необходим диалект Лиспа, полностью включающий его функциональное ядро, но при этом, по возможности, не отягощенный посторонними языковыми возможностями.

2 Практикум по языку программирования Лисп

Любой теоретический курс по программированию должен дополняться практикумом. На кафедре алгоритмических языков ф-та ВМиК МГУ был разработан практикум по функциональному программированию на базе языков Рефал и Лисп.

Практикум на языке Лисп начинается с простых этюдов на программирование вспомогательных функций обработки списков. При их выполнении разрешается использование только базового алгоритмически полного набора функций (CONS, CAR, CDR, COND, EQ, ATOM, QUOTE, EVAL), четырех арифметических функций и средства определения новых функций. Несмотря на минимальность набора допустимых встроенных функций, он позволяет освоить программирование разных видов рекурсии и функций высших порядков.

На следующих этапах обучения студенту предлагается на выбор решение более сложных задач, призванных продемонстрировать студентам практическую ценность языка Лисп. Среди предлагаемых вариантов заданий – различные виды символьной обработки, такие как преобразования формул математической логики или восстановление текста регулярной грамматики по заданному конечному автомату. При выполнении этих заданий допускается использование более широкого множества встроенных функций (всего около пятидесяти).

Таким образом, для проведения практикума желателен инструмент, позволяющий гибко настраивать используемый диалект Лиспа.

3 Пакет Intelib

Указанная гибкость может быть достигнута использованием пакета Intelib [4, 5], изначально разработанного для интеграции различных парадигм программирования в рамках одного языка (C++). Развившийся в рамках проекта Intelib диалект Лиспа, названный Intelib Lisp, предназначался для использования в качестве дополнительного языка. Поэтому получившийся диалект не перегружен посторонними для традиционного Лиспа возможностями. Текущая версия пакета Intelib предоставляет около 150 встроенных Лисп-функций и включает в себя интерактивный интерпретатор языка Intelib Lisp.

Важно, что средствами пакета легко можно построить дополнительный интерпретатор, поддерживающий ограниченное множество встроенных функций Лиспа, например, минимально необходимое при выполнении простых этюдов на программирование.

Недостатком интерпретатора, реализованного в рамках пакета Intelib, является его сравнительно низкая эффективность по времени исполнения, но для учебных задач это не столь существенно. Вместе с тем, для студентов, желающих приобрести дополнительные навыки программирования, Intelib предоставляет как возможность дописывания дополнительных функций на C++ для их использования в Лисп-программе, так и возможность включения фрагментов на Лиспе в программы на C++, что дополнительно иллюстрирует практическую применимость Лиспа.

Указанные особенности пакета делают реализованный диалект Лиспа перспективным кандидатом на роль учебного диалекта.

Список литературы

- [1] A. J. Field and P. G. Harrison. *Functional Programming*. Addison-Wesley, Reading, Massachusetts, 1988.
- [2] G. L. Steele Jr. *Common Lisp the Language*. Digital Press, Burlington MA, second edition, 1990.
- [3] И. Г. Головин, А. В. Столяров. *Мультипарадигмальный подход к преподаванию программирования и роль свободного ПО*. // Тезисы докладов II конференции разработчиков свободных программ «на Протве», г. Обнинск, июль 2005 г. с.114-120
- [4] И. Г. Головин, А. В. Столяров. Объектно-ориентированный подход к мультипарадигмальному программированию. *Вестник МГУ*, сер. 15 (ВМиК), №1, 2002 г., стр. 46–50.
- [5] Официальный сайт проекта Intelib. <http://www.intelib.org>