

Министерство образования и науки Российской Федерации
Федеральное агентство по образованию
Ярославский государственный университет им. П. Г. Демидова
Кафедра математического моделирования

Д. В. Глазков

Пакеты прикладных математических программ

*Методические указания
к проведению лабораторных работ*

*Рекомендовано
Научно-методическим советом университета
для студентов, обучающихся по специальности
Прикладная математика и информатика*

Ярославль 2009

УДК 519.2
ББК В185я 73
Г52

*Рекомендовано
Редакционно-издательским советом университета
в качестве учебного издания. План 2009 года*

Рецензент
кафедра математического моделирования
Ярославского государственного университета им. П. Г. Демидова

Г52 Глазков, Д. В. Пакеты прикладных математических программ:
метод. указания к проведению лабораторных работ / Д. В. Глазков;
Яросл. гос. ун-т им. П. Г. Демидова. — Ярославль: ЯрГУ, 2009. — 40 с.

В методических указаниях приведено описание лабораторных работ. Предназначены для студентов третьего курса, обучающихся по специальности 010501 Прикладная математика и информатика (дисциплина «Пакеты прикладных математических программ», блок ДС), очной формы обучения.

УДК 519.2
ББК В185я 73

©Ярославский государственный
университет им. П. Г. Демидова, 2009

Содержание

Введение	4
1. Лабораторная работа №1. Устойчивость состояний равновесия динамических систем	7
2. Лабораторная работа №2. Вычисление ляпуновских показателей	14
3. Лабораторная работа №3. Решение линейных интегральных уравнений	23
Заключение	32
Список литературы	34
Приложение	37

Введение

Современное математическое исследование редко обходится без использования вычислительной техники. В некоторых областях науки обойтись без численных алгоритмов уже невозможно. Многие рутинные процедуры давно стали частью пользовательских библиотек, которые входят в состав сред программирования. Однако стандартные средства разработки приложений зачастую не могут в полной мере удовлетворить потребности специалистов-математиков. Нельзя приветствовать ситуацию, когда ради приближенного вычисления значения обычного интеграла приходится писать собственные функции, а задача элементарного символьного дифференцирования и вовсе способна поставить в тупик начинающего программиста.

Необходимость решения такого рода задач привела к развитию одного из важных направлений приложения вычислительной техники — систем символьной математики. Их главной особенностью по сравнению с традиционным использованием ПК — действиями с числами — является возможность оперировать с выражениями по определенным правилам преобразований.

Основы для развития таких систем были заложены в 1960-х годах Джоном Маккарти — разработчиком языка списков Lisp. Главным объектом в Lisp является имя элемента, а главной операцией — подстановка. Это привело в дальнейшем к появлению систем, позволяющих работать с формулами, в частности, системы R-Lisp и языка Reduce.

Развитие принципов, заложенных в Lisp-е, послужило созданию более современных и функциональных систем символьной математики, таких как Derive, Maple и Mathematica. На основе ядра системы Maple возможности производить символьные преобразования были также заложены в популярные пакеты MATLAB и MathCad, которые широко известны не в последнюю очередь благодаря удобному пользовательскому интерфейсу. Кроме этого, в перечисленных системах реализованы все стандартные средства привычного «вычислительного» программирования, но уже на языках «сверхвысокого» уровня, дополненные большим числом встроенных библиотек. Все это позволяет рассматривать их как наиболее универсальный инструмент для математического исследования.

Отметим ключевое отличие между системами Mathematica и Maple, с одной стороны, и MATLAB — с другой. Оно заключается в том, что их создатели изначально преследовали различные цели. Если Mathematica

и Maple, как в свое время Lisp, предназначены для символьных преобразований, то MATLAB «заточен» на достижение максимальной эффективности программирования операций с векторами и матрицами. «Векторизация кода» не только облегчает труд программиста, ускоряя процесс разработки, но и позволяет сделать код обозримее по сравнению с обычными «скалярными» языками, что снижает вероятность «человеческих» ошибок при решении сложных задач. Тем не менее Symbolic Math Toolbox пакета MATLAB позволяет выполнять символьные преобразования, а в системах Mathematica и Maple реализованы мощные и удобные механизмы работы со списками. Поэтому перечисленные пакеты являются универсальными инструментами.

Универсальные математические системы [1–8] позволяют компактно и быстро реализовать специализированные математические преобразования, как числовые, так и символьные, входящие в состав сложных алгоритмов, из любой области математики. Такими системами являются пакеты MATLAB, Mathematica, Maple, в меньшей степени MathCad из-за ограниченности возможностей программирования. Отметим, что у MATLAB существуют свободно распространяемые аналоги, такие как Octave, SciLab. Их главное и несомненное преимущество может быть передано словами Free License Software. Обратная сторона медали — отсутствие удобной пользовательской оболочки. Весь интерфейс Octave ограничивается текстовым редактором SciTE, графической оболочкой gnuplot и командной строкой. Правда, Scintilla Text Editor, где первое слово можно перевести как «собранный по крупицам», понимает семантику многих языков программирования. Пакет Octave полностью совместим с MatLab и позволяет выполнять несложные программы, написанные для MatLab, практически без редактирования.

Перечисленные универсальные математические системы используют языки интерпретируемого типа, сравнимые с Basic или Perl в том смысле, что созданные в них программы не требуют специальной стадии компиляции в последовательность машинных команд, сохраняемую в отдельном файле. Ядро пакета Mathematica (или другого пакета) анализирует и выполняет команды в пределах одной входной ячейки (строки ввода) независимо от команд в других ячейках (строках). В MATLAB реализована возможность создания сценариев в m-файлах, сравнимых с CGI-сценариями в Perl.

Предлагаемые лабораторные работы имеют целью приобретение навыков программирования в двух-трех различных пакетах. Представ-