

Предисловие

Эта книга представляет собой учебник по программированию на языке C++. Ее цель, как и всех аналогичных книг, — научить программированию на C++. Как это сделать? Существует два подхода к решению этой задачи: “от частного к общему” и “от общего к частному”. Приверженцы первого подхода предпочитают излагать темы в историческом порядке: от императивного программирования на языке C к объектно-ориентированному и обобщенному программированию на языке C++. Как правило, такие книги содержат подробное описание свойств языка и сопровождаются иллюстративными фрагментами программ. Сторонники второго подхода считают традиционное изложение слишком медленным. Чтобы ускорить освоение языка C++, они предлагают пренебречь подробностями и поскорее приступить к крупноблочному проектированию программ, основанному на стандартных средствах, откладывая изучение деталей на будущее.

Каждый из этих подходов вполне разумен. Нельзя категорично утверждать, какой из них лучше. Думается, что их различие заключается лишь в выборе очередности тем. Совершенно очевидно, что свободное владение языком — необходимое условие для профессиональной работы программиста. Чем богаче ваш лексикон, тем более глубокую мысль вы можете выразить. Это утверждение в полной мере касается не только иностранных языков, но и языков программирования. Следовательно, либо вы сначала подробно изучаете словарь и грамматические правила, а затем осваиваете “разговорные темы”, либо заучиваете фразы из разговорника и лишь потом открываете грамматику. Опасности, подстерегающие читателя, известны каждому, изучавшему иностранный язык. Многие учат слова и правила, но не могут сформулировать простейшую мысль. Другие бойко разговаривают, но не знают грамматики. И то, и другое для программиста недопустимо. Дело в том, что он вынужден общаться с самым придирчивым собеседником на свете — с компилятором. Человек, с которым вы разговариваете на иностранном языке, выполняет интерполяцию искаженных слов и прощает грамматические ошибки, но компилятор не пропускает ни одной неточности. Следовательно, программист, освоивший лишь стандартные идиомы языка, будет крайне ограничен в средствах выражения алгоритмов.

Итак, без подробного изложения свойств языка не обойтись. Некоторые сторонники подхода “от общего к частному” поступают очень просто — за подробностями они отсылают к учебникам, написанным по принципу “от частного к общему”. Продолжая аналогию с иностранным языком, можно сказать, что многие программисты, изучающие язык C++ по таким книгам, осваивают лишь основное подмножество языка, откладывая на будущее те области, которые выходят за пределы учебников.

Поскольку в названии книги употреблено понятие “полный курс”, подход “от общего к частному” неприемлем. Для того чтобы читатель полностью овладел языком C++, нельзя заставлять его играть с “кубиками”, содержание которых ему не известно.

Исходя из этого, я решил описать все три основных слоя языка C++ (императивный, объектно-ориентированный и обобщенный), дополнив их описанием основных технологий: статического метапрограммирования и шаблонного проектирования. Иллюстрируя текст, я отдавал предпочтение полным текстам программ, ко-

которые предоставляют читателям возможность собственноручно проверить сказанное в книге. Кроме того, такие программы позволяют экспериментировать, что, несомненно, способствует более полному освоению языка.

Учитывая разнообразие компиляторов языка C++ и тот факт, что ни один из них не соответствует стандарту языка C++ на 100%, я не стал выбирать один из них в качестве главного. Поэтому работая с программами, приведенными в книге, сначала следует уточнить, какой компилятор для этого подходит лучше других. Например, для частичной специализации шаблонов можно использовать компилятор Code Warrior 8.0, а для управления кучей — Turbo C++ 3.0 или Borland C++ 5.02. Стандартная библиотека шаблонов поддерживается компиляторами Visual C++ 6.0, Borland C++ 5.02 и Code Warrior 8.0. Необходимо также обратить внимание на объявление счетчика в заголовке цикла — компиляторы по-разному трактуют область видимости таких переменных.

Работая над книгой, я представлял себе любознательного и вдумчивого читателя, который с уважением воспринимает точку зрения автора, но не спешит с ней соглашаться, не проверив на практике. Надеюсь, моя книга окажется для вас полезной.

Выражаю глубокую благодарность заведующему редакцией Александру Вадимовичу Слепцову за поощрение и поддержку, а также литературному редактору Екатерине Давидян, чьи советы несомненно улучшили качество изложения. Я признателен также студентам факультета кибернетики Киевского национального университета имени Тараса Шевченко, прослушавшим курс лекций на основе этой книги. Их вопросы и замечания были чрезвычайно полезными и дали много информации для размышлений.

Д.А.Клюшин
Киев, 1.12.2003

От издательства “Диалектика”

Вы, читатель этой книги, и есть главный ее критик. Мы ценим ваше мнение и хотим знать, что было сделано нами правильно, что можно было сделать лучше и что еще вы хотели бы увидеть изданным нами. Нам интересно услышать и любые другие замечания, которые вам хотелось бы высказать в наш адрес.

Мы ждем ваших комментариев и надеемся на них. Вы можете прислать нам бумажное или электронное письмо либо просто посетить наш Web-сервер и оставить свои замечания там. Одним словом, любым удобным для вас способом дайте нам знать, нравится ли вам эта книга, а также выскажите свое мнение о том, как сделать наши книги более интересными для вас.

Посылая письмо или сообщение, не забудьте указать название книги и ее авторов, а также ваш обратный адрес. Мы внимательно ознакомимся с вашим мнением и обязательно учтем его при отборе и подготовке к изданию последующих книг. Наши координаты:

E-mail: info@diagnostika.com
WWW: http://www.diagnostika.com

Информация для писем из:

России: 115419, Москва, а/я 783
Украины: 03150, Киев, а/я 152