А. П. Побегайло

C/C++ для студента

Санкт-Петербург
«БХВ-Петербург»
2006

УДК 681.3.068+800.92С/С++
ББК 32.973.26-018.1
П41

Побегайло А. П.

П41 С/С++ для студента. — СПб.: БХВ-Петербург, 2006. — 528 с.: ил.

ISBN 5-94157-647-1

Подробно рассматриваются языки программирования С и С++. Описаны типы данных, функции, классы, шаблоны, а также библиотеки стандартных функций. Язык программирования С++ рассматривается как объектно-ориентированное расширение языка С, что позволяет последовательно изучить процедурное программирование, объектно-ориентированное программирование и обобщенное программирование. Изложение материала отличается краткостью и снабжено большим количеством простых примеров и листингов, которые поясняют технику программирования на языках С и С++.

Для студентов и программистов

УДК 681.3.068+800.92С/С++
ББК 32.973.26-018.1

Группа подготовки издания:

Главный редактор	*Екатерина Кондукова*
Зам. главного редактора	*Игорь Шишигин*
Зав. редакцией	*Григорий Добин*
Редактор	*Андрей Смышляев*
Компьютерная верстка	*Ольги Сергиенко*
Корректор	*Зинаида Дмитриева*
Дизайн серии	*Игоря Цырульникова*
Оформление обложки	*Елены Беляевой*
Зав. производством	*Николай Тверских*

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 25.05.06.
Формат 60×90$^{1}/_{16}$. Печать офсетная. Усл. печ. л. 33.
Тираж 3000 экз. Заказ №
"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Санитарно-эпидемиологическое заключение на продукцию
№ 77.99.02.953.Д.006421.11.04 от 11.11.2004 г. выдано Федеральной службой по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"
199034, Санкт-Петербург, 9 линия, 12

ISBN 5-94157-647-1

Оглавление

Рецензия на книгу "Языки программирования C и C++" доцента кафедры технологии программирования, факультета прикладной математики и информатики Белорусского государственного университета, к. т. н. Побегайло Александра Павловича

В рецензируемой книге дано довольно полное изложение языков программирования C и C++. Книга представляет собой учебное пособие по программированию на языках C и C++ для студентов высших учебных заведений, специальностей "Информатика" и "Прикладная математика". В книге полностью представлены конструкции языков программирования C и C++: типы данных, операторы и выражения, функции, классы, шаблоны. Поэтому можно считать, что рецензируемая книга представляет собой полное и замкнутое пособие по языкам программирования C и C++.

Книга "Языки программирования C и C++" может использоваться как начинающими, так и подготовленными программистами. А также может быть полезной профессиональным программистам.

Книга имеет несколько особенностей. Во-первых, изложение отличается краткостью и простотой. Во-вторых, все вопросы рассматриваются подробно и полностью. Каждое положение снабжено простым примером, что позволяет студентам быстро перейти к практическому программированию. В-третьих, процедурно-ориентированная часть языка программирования C++ излагается на базе и в сравнении с языком программирования C. Это позволяет построить курс программирования следующим образом: процедурное программирование, объектно-ориентированное программирование, обобщенное программирование.

Считаю, что книга "Языки программирования C и C++" может быть рекомендована к изданию и использоваться в учебном процессе.

Зам. генерального директора
Объединенного института проблем информатики НАН
Республики Беларусь
доктор физико-математических наук, профессор *Тузиков А. В.*

Рецензия на книгу "Языки программирования C и C++" доцента кафедры технологии программирования, факультета прикладной математики и информатики Белорусского государственного университета, к. т. н. Побегайло Александра Павловича

В рецензируемой книге рассмотрены актуальные в настоящее время языки программирования C и C++. Изложение материала отличается простотой и логической стройностью, что позволяет изучать процедурное и объектно-ориентированное программирование последовательно, обращая внимание на расширение языка программирование C конструкциями для объектно-ориентированного программирования.

Книга "Языки программирования C и C++" состоит из четырех частей:

Часть I. Язык программирования C.

Часть II. Язык программирования C++.

Часть III. Стандартная библиотека языка программирования C.

Часть IV. Стандартная библиотека языка программирования C++.

Материалы, представленные в частях I и II книги, соответствуют программе курса "Программирование" для студентов специальностей "Информатика" и "Прикладная математика" университетов и поэтому могут использоваться в качестве основы курса лекций по данному предмету. Части III и IV позволяют использовать книгу в качестве справочного пособия по программированию, так как содержат большое количество справочного материала по стандартным библиотекам языков программирования C и C++.

Рекомендую книгу "Языки программирования C и C++" к печати и использованию в учебном процессе.

Зав. кафедрой технологии программирования
факультета прикладной математики и информатики
Белорусского государственного университета
доктор технических наук, профессор *Курбацкий А. Н.*

Введение

Язык программирования C был разработан в период с 1969 по 1973 годы. Прежде чем переходить к изучению языка программирования C, скажем немного о целях, которые преследовали создатели этого языка, американские программисты Кен Томпсон и Деннис Ритчи. Язык C разрабатывался как язык системного программирования и предназначался для кодирования операционной системы Unix, которая была бы переносима на различные аппаратные платформы. Поэтому в языке программирования C так много возможностей для программирования на низком уровне. Вследствие этого язык программирования C часто также рассматривают как язык ассемблера высокого уровня. Следует отметить, что язык C получился очень мощным и компактным. С годами стало понятно, что разработка этого языка явилась выдающимся достижением в области информационных технологий.
В 1999 году разработчики этого языка были удостоены Национальной медали США за достижения в области технологии.

Язык программирования C++ был разработан с 1980 по 1983 годы Бьерном Страуструпом, сотрудником AT&T Bell Laboratories. Цель разработки этого языка программирования состояла в добавлении в язык программирования C конструкций для объектно-ориентированного программирования, а именно — классов. В 1986 году Бьерн Страуструп опубликовал книгу "Язык программирования C++", которая стала классическим руководством по программированию на языке C++ и переиздавалась не-

сколько раз. Первая спецификация языка программирования C++ была опубликована Бьерном Страуструпом и Маргарет Еллис в 1990 году и называлась "Annotated C++ Reference Manual" или сокращенно ARM. В дальнейшем эта спецификация стала основой для стандартизации языка программирования C++. Международный стандарт языка C++ был утвержден в 1998 году. По этому стандарту язык программирования C++ базируется на стандарте языка программирования C, утвержденном в 1990 году. Эти версии языков программирования C и C++ и рассматриваются в данном пособии.

Настоящая книга представляет собой курс по изучению языков программирования C и C++ для студентов университетов. Содержание и форма изложения курса обусловлены временем, отводимым программами университетов на изучение данного курса.

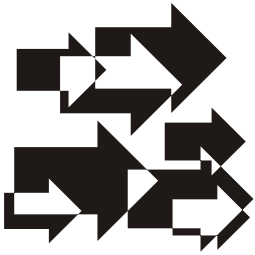
Сначала кратко коснемся содержания курса. Первые две части книги содержат описание языков программирования C и C++. Предполагается, что вопросы, изложенные в этих частях, рассматриваются и обсуждаются на лекциях. Третья и четвертая части разбираются студентами самостоятельно при выполнении лабораторных работ, т. к. на изучение стандартных библиотек языков программирования C и C++ времени в программе не хватает.

Теперь перейдем к форме изложения курса. Я считаю, что на лекциях нужно излагать концепции и технику программирования, а не решать алгоритмические задачи. Для этого предназначены другие курсы. Поэтому форма изложения книги базировалась на стандартах языков программирования C и C++, при этом каждое положение сопровождается максимально простым и понятным работающим примером. Вопросы, касающиеся практического программирования, как, например, работа со структурами, обработка строк, организация файлов, рассматриваются на семинарах и при выполнении лабораторных работ.

Особенно отмечу, что курс разбивается на две части: процедурное программирование и объектно-ориентированное программирование, которые условно совпадают с первыми двумя частями книги. В последнее время можно часто слышать, что изучать про-

граммирование надо сразу с объектно-ориентированных концепций. Мой опыт показывает, что это примерно то же самое, что начинать изучение математики с теории множеств и математической логики. Пока студент четко не усвоил технику и принципы процедурного программирования, он не сможет даже понять, почему класс имеет такой интерфейс, а не другой, не говоря уже о грамотной реализации этого интерфейса. Поэтому сначала нужно обратить внимание на основы и технику программирования, особенно при работе с такими языками программирования, как C и C++, и только затем переходить к объектно-ориентированному программированию.

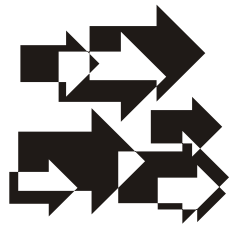
Представленный материал предназначен для первого ознакомления с предметом, после усвоения изложенного материала можно переходить к более продвинутым пособиям, которые приведены в списке литературы. Изложение материала краткое, но довольно полное. Все приведенные в книге программы были проверены на работоспособность компилятором Microsoft Visual C++ 7.0 (.NET).



ЧАСТЬ I

Язык программирования C

ГЛАВА 1



Структура языка С

1.1. Элементы языка С

Как и любой другой, язык программирования С включает следующие элементы: символы, слова и предложения. *Символы* — это базовые графические элементы языка, из которых строятся слова. *Слово* — это последовательность символов, которая имеет смысл в данном языке. Слова, используемые в языке программирования С, разбиваются на три группы: ключевые слова, идентификаторы и константы. Подробнее об этих группах слов будет рассказано в следующих разделах этой главы. Из слов и символов формируются предложения, которые в языке программирования С называются инструкциями. То есть, *инструкция* — это последовательность слов и символов языка программирования С, которая имеет смысл в этом языке. Последовательность инструкций языка С является *программой* на языке С. Программа на языке С преобразуется компилятором в последовательность машинных команд, которые исполняются микропроцессором. Схематически алгоритм работы компилятора может быть представлен блок-схемой, которая показана на рис. 1.1.

Вообще, *компилятор* — это такая программа, входными данными для которой является программа, написанная на определенном языке программирования, а выходными данными — программа в кодах команд микропроцессора, которая функционально соответствует входной программе. Другими словами можно сказать, что

компилятор переводит программу с некоторого языка программирования высокого уровня на машинный код. В соответствии с приведенной схемой процесс компиляции исходной программы можно разбить на следующие этапы:

- ❒ лексический анализ;
- ❒ синтаксический анализ;
- ❒ генерацию кода.

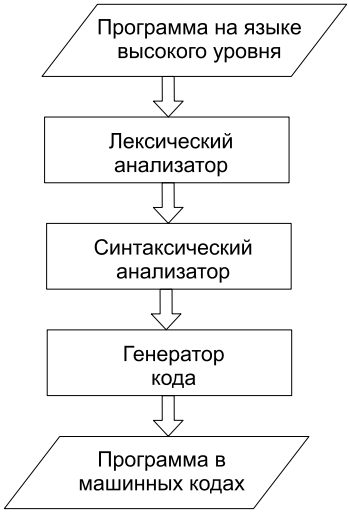


Рис. 1.1. Структурная схема работы компилятора

На этапе лексического анализа компилятор выделяет те элементы языка, из которых строятся предложения (к таким элементам относятся слова и символы). При этом символы должны иметь самостоятельное назначение и использоваться для специальных целей. К таким символам, например, относятся символы для обозначения знаков препинания и операторов. Элементы языка C, которые распознаются на этапе лексического анализа, называются *лексемами* (или *токенами*) (lexical elements или tokens). При этом компилятор проверяет правильность написания слов языка C. То есть определяется, состоят ли слова из допустимых символов языка и правильно ли эти символы используются.

На этапе синтаксического анализа компилятор проверяет, правильно ли составлены инструкции из лексем. А на этапе генерации кода в соответствие каждой инструкции ставится последовательность кодов команд микропроцессора. Если компилятор может выполнять оптимизацию кода, то генератор кода имеет более сложную структуру. В этом случае отображение инструкций языка программирования в коды команд микропроцессора не такое прямолинейное.

1.2. Символы

Каждый язык допускает только определенный набор символов. Не является исключением и язык программирования С, в котором могут использоваться только следующие символы:

- строчные латинские буквы: `a ... z`;
- прописные латинские буквы: `A ... Z`;
- цифры: `0 1 2 3 4 5 6 7 8 9`;
- специальные символы: `. , : ; ' " # { } [ ] ( ) < > & | ^ ! _ + - * / \ % = ? ~ ;`
- символ "пробел";
- нулевой символ или "пусто" (NULL).

Некоторые из символов могут обозначаться специальным образом:

- `\?` — знак вопроса;
- `\'` — апостроф;
- `\"` – кавычки;
- `\\` — обратная косая черта;
- `\0` — пусто.

Кроме того, каждый символ может быть обозначен своим кодом в восьмеричной или шестнадцатеричной системе счисления:

- `\ddd` — где буквы `ddd` обозначают код символа в восьмеричной системе счисления;

- `\xdd` — где буквы `dd` обозначают код символа в шестнадцатеричной системе счисления.

Каждый из символов языка C имеет свое назначение. Буквы и цифры используются главным образом для написания идентификаторов и литералов. Как следует из самого названия специальных символов, они имеют специальное назначение и используются главным образом для обозначения операторов.

Дополнительно в языке программирования C используются *управляющие символы*. Это такие символы, при вставке которых в текст происходит некоторое действие. К ним относятся следующие символы, которые обозначаются специальным образом:

- `\a` — сигнал тревоги;
- `\b` — возврат на шаг;
- `\f` — переход на следующую страницу;
- `\n` — переход на следующую строку;
- `\r` — переход на первую позицию текущей строки;
- `\t` — горизонтальная табуляция;
- `\v` — вертикальная табуляция.

В заключение этого раздела отметим символы, которые используются для разделения слов языка C, такие символы называются *символами разделителями* или *пробельными символами*. К этим символам относятся: пробел, `\b`, `\f`, `\n`, `\r`, `\t`, `\v`.

1.3. Ключевые слова

Ключевые слова языка программирования C это такие слова, которые имеют предопределенное назначение в этом языке и не могут использоваться для других целей. Ниже перечислены все ключевые слова языка C:

```
auto      double  int       struct
break     else    long      switch
case      enum    register  typedef
char      extern  return    union
```

```
const     float   short     unsigned
continue  for     signed    void
default   goto    sizeof    volatile
do        if      static    while
```

Иногда ключевые слова также называют *зарезервированными словами* языка программирования.

1.4. Идентификаторы

Идентификатор — это такое слово языка С, которое может использоваться для обозначения имени переменной, имени функции, имени типа или метки инструкции. Идентификаторы могут включать только алфавитно-цифровые символы языка программирования С, а также символ подчеркивания "_". Кроме того, при написании идентификаторов должны учитываться следующие правила:

- идентификатор должен отличаться от ключевых слов языка С;
- идентификатор не должен начинаться с цифры;
- допускается любая длина идентификатора, но компилятор различает только первые 31 символ;
- в идентификаторе прописные и строчные буквы считаются различными.

Кроме того, не рекомендуется использовать символ "_" в качестве первого символа идентификатора, т. к. этот символ часто используется для именования системных переменных и функций.

В заключение этого раздела скажем, что стиль программирования языка С предлагает для имен переменных, функций и типов использовать строчные буквы, а для имен макросов — прописные буквы.

1.5. Константы

Константами или *литералами* называются некоторые фиксированные значения данных, т. е. такие значения, которые не могут изменяться.

В языке программирования C различаются четыре типа констант:

- целые константы;
- плавающие константы;
- символьные константы;
- строковые константы.

Целая константа может быть записана в десятичной, восьмеричной или шестнадцатеричной системе счисления. В десятичной системе целая константа записывается как обычное десятичное число, при условии, что первая цифра не является нулем. Например, следующие числа являются целыми десятичными целыми константами:

```
12, 234, 1009
```

В восьмеричной системе счисления целая константа записывается восьмеричными цифрами и должна начинаться с нуля. Примерами восьмеричных констант являются следующие числа:

```
012, 0234, 01007
```

В шестнадцатеричной системе счисления целая константа записывается шестнадцатеричными цифрами и должна начинаться с символов `0x` или `0X`. При этом для обозначения шестнадцатеричных цифр от `10` до `15` могут использоваться как строчные буквы `a`, `b`, `c`, `d`, `f`, так и прописные буквы `A`, `B`, `C`, `D`, `F`. Например, следующие целые числа являются целыми шестнадцатеричными константами:

```
0x12, 0X120xABC, 0Xавс
```

Кроме того, в языке программирования C разрешается объявление длинных целых констант, под которые компилятор отводит в два раза больше памяти, чем под целые константы. Для этой цели в конце целой константы ставится буква `l` или `L`. При этом заметим, что если заданное значение целой константы превышает диапазон целого типа данных, то она автоматически представляется длинной целой константой.

Константа с плавающей точкой представляет некоторое действительное число и имеет следующий вид:

```
[целая часть].[дробная часть][E|e[+|-]экспонента]
```

где целая часть, дробная часть и экспонента записываются при помощи десятичных цифр. В определении константы с плавающей точкой должна присутствовать, по крайней мере, одна из частей, заключенных во внешние квадратные скобки. Чтобы получить действительное число, которое представляется константой с плавающей точкой, необходимо целую и дробную часть этой константы умножить на десять в степени, которая задается экспонентой этой константы. Ниже приведены примеры констант с плавающей точкой:

```
3., .14        3.14, 0.314e1, 314e-2.
```

Символьная константа состоит из одного символа, который заключается в апострофы. Ниже приведены примеры символьных констант:

```
'c', 'y', '5', '\101'
```

Сам символ апостроф, используемый в качестве символьной константы, нужно обозначать как `\'`. Отметим, что символьные константы могут содержать символы, не входящие в язык программирования С, например, русские буквы.

Строковая константа представляет собой последовательность символов, заключенную в кавычки. По стандарту длина строковой константы не может превышать 509 символов. Ниже приведены примеры строковых констант.

```
"This is a string.", "Это строка." , "a", "1"
```

Сам символ кавычки, используемый в строковой константе, нужно обозначать как `\"`. Отметим, что строковые константы также могут включать символы, не принадлежащие языку программирования С. Кроме того, в конце каждой строковой константы компилятор помещает нулевой символ `\0`, который отмечает конец строки. В языке программирования С строковые константы обычно называются *строковыми литералами*.

1.6. Инструкции

Инструкцией называется любое синтаксически правильно составленное предложение языка программирования С. Инструкция должна заканчиваться символом `;`. Инструкции описывают неко-

торые действия, которые должна выполнять программа. В языке программирования C допускается пустая инструкция, которая состоит только из символа `;` и не выполняет никаких действий.

Любое количество инструкций, заключенное в фигурные скобки `{` и `}`, называется *составной инструкцией* или *блоком*. Особенно отметим, что после блока точку с запятой ставить не нужно.

1.7. Комментарии

Комментарий — это предложение на естественном языке, которое поясняет ход выполнения программы. Компилятор игнорирует комментарии. В языке программирования C комментарии начинаются парой символов `/*` и заканчиваются парой символов `*/`. Комментарии могут содержать символы, не принадлежащие языку программирования C. Комментарии разрешается применять везде, где используются пробелы. Например, комментарий может выглядеть следующим образом:

```
/* подсчет количества вариантов */
```

Замечание

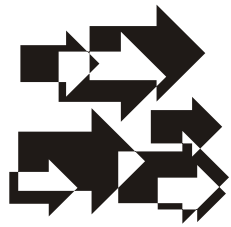
В языке программирования C++ комментарий может начинаться символами `//`. В этом случае весь текст до конца строки считается комментарием.

Например, в языке программирования C++ этот же комментарий можно написать следующим образом:

```
// подсчет количества вариантов
```

В заключение этого раздела сделаем следующее замечание. Так как программы могут использоваться десятилетиями, то поддержка и модификация программного обеспечения может обходиться значительно дороже, чем его разработка. Поэтому считается, что программа, не содержащая комментариев, ничего не стоит и не может быть принята в эксплуатацию.

ГЛАВА 2



Встроенные типы данных и переменные

2.1. Базовые типы данных

Тип данных определяется как множество значений и множество операций, допустимых над этими значениями. В табл. 2.1 приведены *базовые типы данных*, предопределенные в языке программирования C.

Таблица 2.1. *Базовые типы данных*

Тип данных	Длина в байтах	Диапазон значений
`char`	1	0...255 или -128 ... 127
`int`	Не менее 2	Зависит от длины
`float`	4	-3.4E-38 ... 3.4E+38
`double`	8	-1.8E-308 ... 1.8E+308

В стандарте языка C длина в байтах и, соответственно, диапазон типа данных `int` не определены точно и зависят от системы, на которой реализован компилятор. Под системой понимается микропроцессор и операционная система, на которой функционирует компилятор. Однако предполагается, что этот тип данных, по крайней мере, содержит диапазон целых чисел от `-32768` до `32767`.

Диапазоны типов данных `float` и `double` приведены с округлением до первого знака после точки. Точные границы этих диапазонов указаны в заголовочном файле float.h.