

Введение

В прошлом тысячелетии основами образования были чтение, письмо и арифметика; теперь это чтение, письмо и *компьютеры*. Умение программировать является основой образования любого студента технических специальностей. Кроме самих приложений, это первый этап в понимании природы бесспорного воздействия информатики на современный мир. Эта книга предназначена для тех, кто хочет изучить программирование или нуждается в нем для научных целей.

Задача книги — предоставить студентам практические навыки и знания основных инструментальных средств, необходимых для эффективного использования вычислительной техники. Наш подход — убедить студентов, что создание программ — это вполне естественная, занимательная и творческая практика. Для иллюстрации основных концепций и демонстрации возможностей написания программ в практических целях здесь представлены классические приложения в области прикладной математики и научных задач.

Для всех программ в этой книге используется язык программирования Python. Хотя язык Python упоминается в названии книги, по существу эта книга о *фундаментальных концепциях программирования*, а не о самом языке Python. Книга позволяет приобрести практические навыки решения задач средствами, применимыми во многих современных вычислительных системах, а исчерпывающее изложение материала предназначено для тех, кто не обладает опытом программирования.

В книге использован *междисциплинарный* подход к традиционной программе обучения CS1, при этом подчеркивается роль вычислений в других дисциплинах: от материаловедения и генетики до астрофизики и сетевых систем. Этот подход демонстрирует студентам основную идею: математика, наука, техника и компьютерные вычисления тесно переплетаются в современном мире. Хотя этот учебник по программе CS1 предназначен для всех студентов-первокурсников, интересующихся математикой, наукой и техникой, книга применима также и для самостоятельного изучения или как дополнительный курс при интеграции программирования с другой областью науки.

Изложение. Книга организована по четырем уровням обучения программированию: базовые элементы, функции, объектно-ориентированное

программирование и алгоритмы. На каждом уровне, перед переходом к следующему, необходимая для создания программ информация предоставляется читателю в доверительной форме. Основой применяемого подхода является использование примеров программ, решающих интригующие задачи, и предоставление упражнений, варьирующихся по сложности от достаточно простых до довольно сложных, требующих творческих решений.

К базовым элементам относятся переменные, оператор присвоения, встроенные типы данных, управление потоком, массивы, ввод-вывод, графика и звук.

Функции и модули — это первый шаг студента к модульному программированию. Предполагается, что читатель уже знаком с математическими функциями перед переходом к функциям Python и последующим рассмотрением значения программных функций, включая библиотеки функций и рекурсии. Здесь подчеркивается фундаментальная идея разделения программы на компоненты, допускающие независимую отладку, поддержку и многократное использование.

Объектно-ориентированное программирование — это введение в абстракцию данных. Здесь излагаются концепции типов данных и их реализации с использованием механизма классов Python. Студенты обучаются тому, как *использовать*, *создавать* и *проектировать* типы данных. Модульность, инкапсуляция и другие современные парадигмы программирования — это центральные концепции на данном этапе.

Алгоритмы и структуры данных объединяют современные парадигмы программирования с классическими методами организации и обработки данных, которые остаются эффективными и для современных приложений. Этап содержит введение в классические алгоритмы сортировки и поиска, а также фундаментальные структуры данных. Здесь подчеркивается использование научного метода для изучения характеристик эффективности реализаций.

Приложения в науке и технике — главная тема книги. Каждая рассматриваемая концепция программирования мотивируется исследованием ее воздействия на определенные приложения. Приводятся примеры из прикладной математики, физики, биологии и самой информатики. Они включают моделирование физических систем, числовые методы, визуализацию данных, синтез звука, обработку изображений, финансовые модели и информационные технологии. К конкретным примерам относится применение в первой главе цепей Маркова для рангов веб-страницы, а также анализа и решения задачи просачивания, моделирования N тела и феномена “тесного мира”. Эти приложения — неотъемлемая часть книги. Они привлекают внимание студентов к материалу, иллюстрируют важность концепций программирования и предоставляют убедительное доказательство важности роли компьютерных вычислений в современной науке и технике.

Главная задача книги — научить определенным техникам и привить навыки, необходимые для поиска эффективных решений любых задач программирования.

Поэтому мы сосредоточимся на программировании частностей, а не на общем программировании.

Использование в учебном процессе. Книга предназначена для студентов первого курса, для обучения новичков программированию в контексте научных приложений. Материал книги подходит для студентов, обучающихся программированию в знакомом контексте. Студенты, закончившие курс с использованием этой книги, будут хорошо подготовлены к применению своих навыков в последующих курсах, если понадобится дальнейшее изучение информатики.

Студенты информационных специальностей также могут извлечь пользу из примеров программирования в контексте научных приложений. Программист нуждается в той же базовой подготовке и научных методах, а также в том же знании роли компьютерных вычислений в науке, как и биолог, инженер или физик.

Действительно, междисциплинарный подход позволяет институтам и университетам обучать будущих специалистов в области информатики и других научно-технических специальностей на *той же* курсе. Здесь изложен материал согласно учебному плану CS1, но перенос основного внимания на приложения вносит жизнь в абстрактные концепции и поощряет студентов изучать их. Междисциплинарный подход предлагает студентам задачи по многим дисциплинам, помогая им выбрать профилирующий предмет более осмысленно.

Безотносительно конкретного механизма, эта книга наилучшим образом подходит для упомянутого ранее учебного плана. Во-первых, такая позиция позволяет манипулировать знакомым по средней школе математическим материалом. Во-вторых, студенты, изучавшие программирование на предыдущих курсах, научатся эффективнее использовать компьютеры при переходе к курсам по своим специальностям. Подобно письму и чтению, программирование бесспорно является основополагающим навыком для любого ученого или инженера. Студенты, овладевшие концепциями из этой книги, будут непрерывно совершенствовать полученные навыки в практической деятельности, пользуясь преимуществами использования компьютеров для решения или лучшего понимания задач, возникающих в выбранной ими области.

Предпосылки. Книга предназначена для обычных студентов первого курса научно-технических вузов. Таким образом, здесь не предусматривается предварительная подготовка свыше обычной обязательной для других базовых научных и математических курсов.

Знание математики важно. Хотя на математический материал здесь акцент и не делается, в книге учтен школьный курс математики, включая алгебру, геометрию и тригонометрию. Большинство студентов из нашей целевой аудитории изначально отвечают этим требованиям. На самом деле авторы полагаются на знакомство читателя с базовым курсом математики, чтобы ознакомить их с основными концепциями программирования.

Интерес к науке — также необходимый компонент. Студенты научных и технических специальностей априори полагают, что научный поиск способен помочь объяснить происходящее в природе. Это мнение подкрепляется примерами простых программ, говорящих красноречивее всяких слов о природном мире. Здесь не подразумевается наличие специальных знаний, кроме предоставляемых обычными уроками математики, физики, биологии или химии в средней школе.

Опыт программирования необязателен, но и не повредит. Обучение программированию является главной задачей книги, поэтому наличие предварительного опыта программирования не подразумевается. Однако создание программы для решения новой проблемы является сложной интеллектуальной задачей, поэтому студенты, писавшие программы в средней школе, могут извлечь пользу из имеющихся знаний. Книга подходит для обучения студентов с различной предварительной подготовкой, поскольку приложения ориентированы и на новичков, и на экспертов.

Таким образом, практически все студенты научных и технических специальностей готовы пройти курс по этой книге в составе учебного плана первого семестра.

Задачи. Что преподаватели старших курсов научных и технических вузов могут ожидать от студентов, закончивших курс на основании этой книги?

Книга учитывает учебный план CS1, но любой, кто вел вводный курс программирования, знает, что ожидания преподавателей старших курсов обычно весьма высоки: каждый из них полагает, что все студенты знакомы с компьютерной средой и подходом, который они хотели бы использовать. Профессор физики мог бы ожидать, что студенты за выходные разработают моделирующую программу; технический профессор мог бы ожидать, что студенты будут использовать некий пакет для решения дифференциальных уравнений в цифровой форме; а профессор информатики мог бы ожидать знания подробностей определенной среды программирования. Реально ли удовлетворить столь разнообразные ожидания? Должен ли быть отдельный вводный курс для каждого набора студентов?

Институты и университеты решают эти вопросы с тех пор, как компьютеры получили широкое распространение во второй половине XX столетия. Наш ответ на них вы найдете в этом общем вводном курсе по программированию, похожем на обычные вводные курсы по математике, физике, биологии или химии. Эта книга призвана обеспечить базовую подготовку, необходимую всем студентам научных и технических специальностей, а также помочь уяснить, что в информатике есть много больше, чем только программирование. Преподаватели, обучающие студентов, учившихся по этой книге, могут ожидать, что они будут иметь знания и опыт, необходимые для адаптации к новым вычислительным системам, и эффективно использовать компьютеры для разных целей.

Что могут ожидать *студенты* старших курсов, закончившие обучение на основании этой книги?

Мы заявляем, что изучить программирование вовсе не трудно и использовать компьютерные средства весьма полезно. Овладевшие материалом этой книги студенты готовы решать вычислительные задачи везде, где они могут возникнуть в последующей практической деятельности. Они узнают, что современные среды программирования, такие как язык Python, помогают решить любую вычислительную задачу, с которой они могли бы встретиться позже, а также приобрести опыт по изучению, освоению и использованию других вычислительных средств. Студенты, интересующиеся информатикой, смогут удовлетворить свой интерес, а студенты научных и технических специальностей будут готовы интегрировать компьютерные вычисления в свои исследования.

Сайт книги. Дополнительная информация по этой книге приведена на веб-сайте по адресу <http://introcs.cs.princeton.edu/python>.

Для экономии места мы называем этот сайт в книге просто *сайтом книги*. Он содержит материал для преподавателей, студентов и всех читателей книги. Хотя материал сайта кратко описывается здесь, все пользователи Интернета знают, что лучше посмотреть самому, чем прочесть описание. За несколькими исключениями, весь материал общедоступен.

Важнейшее преимущество сайта книги в том, что он позволяет преподавателям и студентам использовать собственные компьютеры для преподавания и изучения материала. Любой читатель, имеющий компьютер и браузер, может начать обучение программированию, следуя инструкциям на сайте книги. Процесс не труднее загрузки медиапроигрывателя и музыки. Подобно любым веб-сайтам, сайт книги непрерывно развивается. Это основной ресурс для всех владельцев данной книги. В частности, размещенные на нем дополнительные материалы критически важны для нашей задачи интеграции информатики в образовательный процесс.

Преподавателям сайт книги предоставляет информацию об обучении. В первую очередь эта информация сосредоточивается на стиле обучения, который за прошедшее десятилетие существенно развился. Мы предоставляем две лекции в неделю для большой аудитории и дополняем их двумя классными занятиями в неделю, когда студенты небольшими группами встречаются с преподавателями или ассистентами. На сайте книги есть также слайды к лекциям.

Ассистентам сайт книги предоставляет подробные наборы задач и программных проектов на основании упражнений из книги. Каждая программа предназначена для демонстрации соответствующей концепции в контексте занимательного приложения, достаточно сложного для каждого студента. Набор заданий воплощает наш подход к обучению программированию. Сайт книги полностью определяет все задания и предоставляет подробную, структурированную

информацию, призванную помочь студентам выполнить их за выделенное время. Здесь также содержатся описания предложенных подходов и уточнение того, что именно следует преподавать на классных занятиях.

Студентам сайт предоставляет быстрый доступ к большей части материала книги, включая исходный код и дополнительные материалы для самоподготовки. Для большинства упражнений “книге” предоставляются решения, включая полный код программ и проверочные данные. Здесь есть много информации, связанной с заданиями на программирование, включая рекомендованные подходы, контрольные списки, часто задаваемые вопросы и проверочные данные.

Другим читателям сайт служит ресурсом дополнительной информации, связанной с материалом книги. Все содержимое сайта представляет собой ссылки и другие указания на подробную информацию по рассматриваемой теме. Детальной информации доступно больше, чем сможет полностью освоить любой человек, но наша задача заключается в том, чтобы предоставить достаточно дополнительной информации о содержимом книги и удовлетворить потребности любого читателя.

Благодарности. Этот проект находится в разработке с 1992 года. На протяжении этих лет слишком много людей поспособствовали его успеху, чтобы упомянуть их всех здесь. Особая благодарность Энн Роджерс (Anne Rogers) за то, что помогла начать дело; Дейву Хэнсону (Dave Hanson), Эндрю Аппель (Andrew Appel) и Крису ван Вик (Chris van Wyk) за их терпение в объяснении абстракций данных; Лайзе Ворthingтон (Lisa Worthington), которая стала первой студенткой, преодолевшей трудности изучения этого материала на первом курсе. Мы также с благодарностью признаем усилия /dev/126. Мы признательны аспирантам, преподавателям и тысячам студентов Принстонского университета, обучавшихся этому материалу прошедшие 25 лет.

Роберт Седжвик
Кевин Уэйн
Роберт Дондеро
Апрель 2015

Соглашения, принятые в книге

Здесь используются соглашения, общепринятые в компьютерной литературе.

- Новые термины в тексте выделяются *курсивом*. Чтобы обратить внимание читателя на отдельные фрагменты текста, также применяется *курсив*.
- Текст программ, функций, переменных, URL веб-страниц и другой код представлен моноширинным шрифтом.
- Все, что придется вводить с клавиатуры, выделено полужирным моноширинным шрифтом.
- Знакоместо в описаниях синтаксиса выделено курсивом. Это указывает на необходимость заменить знакоместо фактическим именем переменной, параметром или другим элементом, который должен находиться на этом месте: `BINDSIZE=(максимальная ширина колонки)*(номер колонки)`.
- Пункты меню и названия диалоговых окон представлены следующим образом: Menu Option (Пункт меню).
- В листингах каждая строка имеет номер. Это сделано исключительно для удобства описания. В реальном коде нумерация отсутствует. Текст некоторых строк кода в листингах иногда бывает слишком длинным и не помещается на странице книги. Таким образом, отсутствие номера в начале строки свидетельствует о том, что строка является продолжением предыдущей и в реальном коде их разрывать не следует.

Текст некоторых абзацев выделен специальным шрифтом. Это примечания, советы и предостережения, которые помогут обратить внимание на наиболее важные моменты в изложении материала и избежать ошибок в работе.

От издательства

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 авторам.

Мы ждем ваших комментариев. Вы можете прислать письмо по электронной почте или просто посетить наш веб-сайт, оставив на нем свои замечания. 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 подходящими для вас.

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

E-mail: info@dialektika.com
WWW: <http://www.dialektika.com>

Наши почтовые адреса:

в России: 195027, г. Санкт-Петербург, ул. Магнитогорская, д. 30, ящик 116
в Украине: 03150, Киев, а/я 152