

Введение

Программированию сегодня посвящено множество учебников и пособий. Однако в основном внимание в них уделяется непосредственно написанию программ на каком-либо языке программирования.

Цель данного учебного пособия — осветить данный вопрос с другой стороны — с основ.

В первой главе будет рассмотрен вопрос представления информации в цифровой форме. Поскольку все вычислительные устройства обрабатывают информацию именно в цифровом виде, необходимо понимать, как она оцифровывается, какие преимущества и недостатки это дает. Данный процесс будет рассмотрен на примере текстов и изображений.

Во второй главе описываются ключевые элементы любого вычислительного устройства — центральный процессор, память и подсистема ввода-вывода. Описание дано в упрощенном виде, который облегчает понимание принципов работы ЭВМ в целом.

Третья глава посвящена непосредственно вопросам написания программ. Будут описаны самые первые способы программирования — на машинных кодах, на ассемблере и затем будут рассмотрены языки более высоких уровней.

Понимание того, что реально происходит в компьютере при компиляции и выполнении программы, позволит обучающимся подходить к программированию более профессионально.

1 Цифровое представление информации

1.1. О множествах, аналоговом и дискретном представлении.

Множество — это набор каких-либо элементов. Например, в мире есть множество городов (и в данном примере *город* — элемент множества «города»). В каждом городе есть множество домов. В каждом доме проживает некоторое множество жильцов.

Множества бывают *конечные* и *бесконечные*. Конечное множество содержит конечное число элементов — другими словами, все элементы этого множества можно пересчитать и сказать: в этом множестве N элементов. Все приведенные ранее примеры — это конечные множества, так как все их элементы можно пересчитать (это может быть трудно в некоторых случаях, но главное что такая операция выполнима *в принципе*).

Бесконечное множество, как следует из названия, содержит бесконечное число элементов. Здесь трудно привести пример из реальной жизни, поэтому воспользуемся понятием из математики — допустим, множеством целых чисел. Известно, что нет «верхней границы» чисел, т. е. для любого числа есть большее число. Поэтому в множестве целых чисел — бесконечное число элементов. То же самое можно сказать про множество иррациональных чисел: в десятичной дроби число знаков после запятой может быть бесконечно большим.

Применительно к компьютерной технике чаще говорят об аналоговом и дискретном представлении чего-либо, но они имеют много общего с бесконечными и конечными множествами.

Возьмем, для примера, автомобиль и его органы управления. Рассмотрим руль: как известно, это некоторое колесо, которое водитель поворачивает на некоторый угол. Как описать положение руля математически, т. е. с помощью чисел? Очевидно, что угол поворота задается градусами, например в виде «руль повернут вправо на 10 градусов». Теперь зададимся вопросом: сколько всего положений руля возможно в принципе? Иными словами, каково множество

значений возможного угла поворота руля? Конечно оно или бесконечно? Допустим, ответ такой: руль может быть повернут на пол оборота вправо и на столько же — влево, т. е. 180 градусов в одну сторону, 180 в другую — таким образом, возможно 360 значений угла поворота руля ($-180; -179; \dots; 0; \dots; 179; 180$). Однако на такой ответ можно возразить: а если мы задаем угол поворота с точностью не 1 градус, а 0,1 градуса? Тогда количество значений, описывающих возможный угол поворота руля, возрастает в 10 раз ($-180; -179,9; -179,8; \dots; 179,9; 180$) и будет равно 3600. Можно продолжить: задавать угол поворота с точностью в 0,01 градус — тогда число возможных значений возрастает еще в 10 раз и будет равно уже 36000. Повторять эту операцию можно бесконечно: представить положение с точностью 0,001; потом 0,0001 и т. п. Получается, что множество возможных значений угла поворота руля — бесконечно, так как можно бесконечно увеличивать точность представления. В таких случаях говорят, что величина, которая описывает ту или иную сущность (в нашем случае — это угол поворота руля), непрерывна, т. е. может иметь бесконечное множество значений. Хотя это и не очень корректно, но для простоты будем считать, что угол поворота руля — это *аналоговая* величина. Другими словами, аналоговая величина — это непрерывная величина.

В то же время в автомобиле есть и органы управления, множество положений которых конечно. Например, это коробка передач: допустим, существует всего 6 передач (включая передачу заднего хода и нейтральную передачу). Число передач конечно: нельзя включить, скажем, полуторную передачу. Другим примером являются различные выключатели, например выключатель обогрева заднего стекла имеет всего два значения: включен или выключен. Это — примеры *дискретных* величин, т. е. величин, множество значений которых конечно. Например, величина «передача» имеет 6 возможных значений: (1-я; 2-я; 3-я; 4-я; нейтральная; задняя).

В сегодняшнее время многие широко используют термин «цифровая техника», но далеко не все понимают, что он означает.

Мы только что рассмотрели два важных понятия: аналоговое и дискретное представление величин. В нашем реальном мире мы в основном встречаемся с аналоговым представлением: например, наша текущая координата — это тоже аналоговая величина (аналогично углу поворота руля). Цифровая же техника работает только с дискретными величинами. Рассмотрим еще несколько важных моментов про дискретные величины.

Допустим, у нас есть дискретная величина — конечное множество некоторых значений, например набор из 20 лопаток турбины. Нам необходимо как-то уметь однозначно отличать один элемент множества от другого. Как научиться это делать? Самое очевидное решение — пронумеровать каждый элемент множества. Каждая лопатка получит свою цифру.

Любая цифровая техника оперирует исключительно дискретными величинами. Как только что было показано, самый простой и очевидный способ обозначить каждое значение дискретной величины (каждый элемент конечного множества) — это дать ему некоторое число. Именно поэтому и происходит название — *цифровая* техника, к которой, конечно же, в полной мере относятся и компьютеры. В цифровой технике все представляется в виде чисел.

Во многих областях сначала появилась аналоговая техника. Например, в звукозаписи — грампластинка, кассета — это аналоговое представление звука (технические особенности — почему именно так — мы рассматривать не будем). Однако позже появилась идея представлять звук в виде чисел (каким образом это делается мы еще рассмотрим) — появился класс устройств, которые работают со звуком уже на цифровом принципе (проигрыватели компакт-дисков, компьютеры, флеш-плееры и т.п.). И оказалось, что такой подход имеет ряд преимуществ (и недостатков, но сегодня преимущества явно перевешивают недостатки), поэтому он и получил широкое распространение.

Подобная картина наблюдается и во множестве других областей: видео, связь, промышленность и т.п. Оказалось, что представлять какие-либо значения в виде дискретных величин и работать с ними в числовом виде удобней, чем в аналоговом.

Возникает вопрос: как перейти от аналогового представления, которое используется в реальном мире, в дискретное — которое необходимо для цифровой техники? Например, как картинку представить в числовой форме? Как сделать то же самое с нашими текущими координатами? Мы еще будем рассматривать этот вопрос и увидим, как такие преобразования выполняются в наиболее массовых областях — аудио, графика, видео и т.п.

На самом деле, принцип при переходе от аналогового представления к дискретному везде общий. Поясним его на примере того же руля в автомобиле. Допустим, нам все же понадобилось как-то описывать положение руля в числовом виде (например, мы разрабатываем компьютер-автопилот). Необходимо как-то от бесконечного множества перейти к конечному. Решение довольно просто: нуж-

но оговорить, какая точность задания угла поворота нас устроит, и описывать угол поворота лишь с этой точностью. Тогда мы получим конечное число углов поворота — например, для точности в 1 градус получится 360 элементов и для описания угла поворота руля мы будем использовать числа от -180 до 180 . Однако при этом мы и кое-что теряем: теперь нет возможности задать угол с большей точностью, чем 1 градус. Если мы попросим задать угол в $175,8$ градусов, то цифровое устройство сможет установить лишь наиболее близкий к этой величине угол: 176 градусов.

Поэтому при переходе от аналоговой формы к цифровой всегда необходимо задаваться точностью, с которой нужно выполнять эту операцию. Точность определяется в зависимости от решаемой задачи. Например, в нашей задаче разработки автопилота вряд ли понадобится точность установки руля лучше, чем в 1 градус. Если мы сделаем точнее, то различия в управлении все равно не заметим (человек не отличит поворот руля в 1 градус от поворота руля в $1,4$ градуса), а вот цифр на описание угла поворота будем использовать больше.

Итак, цифровые устройства оперируют цифрами, только цифрами и ничем кроме цифр. Для того чтобы они это могли успешно делать, были придуманы способы представления различной информации в числовом виде, некоторые из которых будут рассмотрены в следующих разделах.

1.2. Системы исчисления

Поскольку цифровые устройства работают с числами, рассмотрим некоторые нюансы этой области, без понимания которых сложно будет понять принципы их функционирования.

Известно, что человечество привыкло пользоваться десятичной системой исчисления. Слово «десятичная» означает, что для описания чисел нам доступно десять различных символов (от нуля до девяти). Также необходимо вспомнить, что такое *разряд*: это одно место для символа. Например, число 538 имеет 3 разряда. Также существует понятие *старшинство разряда*: самым младшим считается самый правый разряд, самым старшим — левый разряд. То есть в числе 538 самый младший разряд — это разряд, где располагается цифра 8 , а самый старший — в котором расположена цифра 5 .

Правило возрастания чисел в любых системах исчисления одинаково: увеличивается самый младший (самый правый) разряд. Когда число в самом младшем разряде максимально для данной системы исчисления (для десятичной системы это число 9), то оно

сбрасывается на минимальное, а в следующем по старшинству разряде число увеличивается на 1. То есть чтобы увеличить число 538 на 1, увеличиваем младший разряд, получаем 539. Чтобы увеличить еще на единицу, придется девятку (максимальная цифра в десятичной системе исчисления) заменить на ноль (минимальная цифра) и увеличить на единицу следующий по старшинству разряд, т. е. тот, где расположена цифра 3. Таким образом, увеличив 539 на единицу, получим 540.

Если следующий по старшинству разряд также содержит максимальную цифру, то мы сбрасываем в минимальную цифру и его и переходим к следующему по старшинству разряду. Например, увеличив число 599 на единицу, получим 600.

Если мы дошли до самого старшего разряда, но и он содержит максимальную цифру — правило по-прежнему соблюдается. Например, увеличивая число 999 на единицу, мы получим 1000. На самом деле, мы неявно выполнили операцию увеличения числа 0999 на единицу и увеличили цифру 0 в самом старшем разряде.

Вернемся к системам исчисления. Обычно произносят «десятичная система исчисления» или, например, «двоичная система исчисления». Число, которое называют применительно к выражению «система исчисления», называется *основанием системы исчисления* и фактически означает, сколько символов доступно в этой системе исчисления для описания чисел. Например, в двоичной системе исчисления доступно всего два символа: 0 и 1. Правило, по которому возрастают числа, то же самое.

Допустим, имеется число 0 в двоичной системе исчисления. Увеличиваем его на единицу — получаем 1. Увеличиваем снова. Единственный разряд, который является пока по совместительству и самым старшим, и самым младшим, уже содержит максимальную цифру (не забываем, что система исчисления двоичная и цифра 1 — это максимальная цифра в этой системе), поэтому руководствуемся тем же правилом: представляем себе не 1, а 01, затем увеличиваем старший разряд и сбрасываем в минимальную цифру младший. Получаем 10. Увеличиваем снова на единицу — получим 11. В табл. 1 отображены 16 чисел в десятичном и двоичном эквиваленте.

Нули перед двоичными числами здесь дописаны специально для облегчения понимания, на самом деле такой необходимости нет, хотя и принципиально ничего не меняется — ведь все равно, 5 рублей у вас в кошельке или 05.

Следующий вопрос, в котором необходимо разобраться, — сколько всего чисел можно описать, имея известное число разрядов

Таблица 1

Десятичное и двоичное представления чисел

Десятичная	Двоичная	Десятичная	Двоичная
0	0000	8	1000
1	0001	9	1001
2	0010	10	1010
3	0011	11	1011
4	0100	12	1100
5	0101	13	1101
6	0110	14	1110
7	0111	15	1111

в известной системе исчисления? Например, сколько всего чисел мы можем описать, имея 3 разряда в десятичной системе исчисления? Правильный ответ — 1000 (от 0 до 999). А если разрядов 5? Получаем ответ 100000 (от 0 до 99999). А сколько всего чисел можно описать, имея 4 разряда в двоичной системе исчисления? Правильный ответ — 16, вы можете сами в этом убедиться, глядя на табл. 1: в ней описаны 16 чисел и максимальное число 1111 нельзя увеличить, не потребовав новый, 5-й разряд.

Закономерность довольно очевидна: количество чисел, которое можно описать, имея заданное количество разрядов в известной системе исчисления, равно основанию системы исчисления в степени количества разрядов. Распишем вышерассмотренные примеры и убедимся в этом:

$$1000 = 10^3;$$

$$100000 = 10^5;$$

$$16 = 2^4.$$

Многие представляют числа в других системах исчисления как что-то особенное, необычное. На самом деле, все довольно просто. Система исчисления — это лишь способ описать какое-либо число, которое само по себе от этого не меняется. Например, у вас есть 5 яблок. Независимо от того, в какой системе исчисления представить это число, количество яблок, которое у вас есть, не изменится. И неважно, как сказать: есть у вас 5 яблок или есть у вас 101 яблок (в двоичной системе исчисления).

Просто мы с детства привыкли оперировать в десятичной системе исчисления и поэтому все другие кажутся нам чужими. На самом деле, если бы нас с детства учили оперировать числами в двоичной системе исчисления, то мы бы легко сейчас говорили на рынке «Вудьте добры, 101 яблоко», а число 5 казалось бы нам чужеродным.

Поговорим теперь о том, какие системы исчисления используются в цифровой технике. Все довольно просто: любая цифровая техника реально работает только с числами в двоичной форме и никакой другой. Мы немного позже рассмотрим, почему сложилось именно так, а не иначе. Все числа в других системах исчисления, которые вы вводите в компьютер или другое цифровое устройство, на самом деле за вас переводятся в двоичную форму.

Также упомянем шестнадцатеричную систему исчисления, которая иногда нам будет нужна. В ней, как следует из названия, имеется 16 символов на описание цифр и для описания эквивалентов чисел 11, 12, ..., 15 используются латинские буквы А, В, ..., F. Эти числа представлены в табл. 2.

Таблица 2

Десятичное и шестнадцатеричное представления чисел

Десятичная	Шестнадцатеричная	Десятичная	Шестнадцатеричная
0	00	18	12
1	01	19	13
2	02	20	14
3	03	21	15
4	04	22	16
5	05	23	17
6	06	24	18
7	07	25	19
8	08	26	1A
9	09	27	1B
10	0A	28	1C
11	0B	29	1D
12	0C	30	1E
13	0D	31	1F
14	0E	32	20
15	0F	33	21
16	10	34	22
17	11	35	23

Если вы когда-нибудь в пространстве цифровой техники встретите строку, где нет ни одной буквы «старше» F, то, скорее всего, перед вами реально не текстовая строка, а число в шестнадцатеричной форме, например A8B9EEFAC718B.

На случай, если вам понадобится перевести число из одной системы исчисления в другую, можно пользоваться встроенным в Windows калькулятором: «Пуск» → «Стандартные» → «Калькулятор». В меню «Вид» его нужно перевести в режим «Инженерный». Появятся новые функции, нас будет интересовать окошко, которое на рис. 1 обозначим прямоугольником.