



Златопольский Дмитрий Михайлович
*Кандидат технических наук, доцент кафедры
 информатики и прикладной математики
 Московского городского педагогического университета.*

Ещё раз о «счастливых» билетах

В статье [1] был приведён ряд вариантов программы решения задачи подсчёта так называемых «счастливых» билетов (автобусных или других билетов, в шестизначных номерах которых сумма трёх первых цифр равна сумме трёх последних). В результате был получен вариант программы, решающий задачу за время, примерно в 4 раза меньшее, чем первоначальный вариант. Оказывается, имеется возможность решить задачу за ещё более короткое время. Методика, реализующая такую возможность, описывается в данной статье.

Шестизначный номер билета состоит из двух частей: левой (3 цифры) и правой (тоже 3 цифры), причём в обеих частях номера «счастливого» билета сумма цифр одинакова. В каждой части сумма цифр может быть от 0 до 27. Обозначим эту сумму – *сум*. Некоторое значение *сум* могут давать в общем случае несколько трёхзначных чисел. Например, сумму, равную 2, – числа 002¹, 011, 020, 101, 110, 200 (всего их 6). Значит, общее число «счастливых» билетов с такой суммой равно $6 \times 6 = 36$. Поэтому можем сказать, что общее количество шестизначных «счастливых» билетов

$$\text{будет равно } KЧ_0 + KЧ_1 + KЧ_2 + \dots + KЧ_{26} + KЧ_{27} = \sum_{\text{сум}=0}^{27} KЧ_{\text{сум}} \cdot KЧ_{\text{сум}},$$

где $KЧ_{\text{сум}}$ – количество трёхзначных чисел с суммой *сум*.

Но как найти значения $KЧ_{\text{сум}}$?

Начнём с более простого случая – определим $KЧ_{\text{сум}}$ для двузначных чисел. Для них значение *сум* может быть равно 0, 1, 2, ..., 18. Обсудим, какой может быть последняя цифра двузначного числа при известной сумме цифр. Для этого составим и исследуем таблицу:

¹ Примем, что номера могут иметь начальные нули и что возможен также номер билета 000000.

Таблица 1

Сумма двух цифр (<i>сум</i>)	Возможные значения последней цифры
0	0
1	0 1
2	0 1 2
...	
9	0 1 2 ... 9
10	1 2 ... 9
11	2 3 ... 9
...	
17	8 9
18	9

Из таблицы 1 можно установить минимальное и максимальное значения последней цифры по следующему правилу:

```
если сум > 9
то
    мин_посл = сум - 9
    макс_посл = 9
иначе
    мин_посл = 0
    макс_посл = сум
все
```

Общее число возможных значений последней цифры равно $\text{макс_посл} - \text{мин_посл} + 1$. А так как каждому из возможных значений соответствует только одна оставшаяся цифра, то общее число двузначных чисел, сумма которых составляет *сум*, и будет равно значению последнего выражения.

Вернёмся к основной задаче – определению количества *трёхзначных* чисел с суммой цифр *сум*. Здесь также можно установить¹ правило определения минимального и максимального значений последней цифры в зависимости от *сум*:

```
если сум > 9
то
    макс_посл = 9
иначе
    макс_посл = сум
все
если сум > 18
то
    мин_посл = сум - 18
иначе
    мин_посл = 0
все
```

Определив все возможные значения последней цифры, можно для каждой из них найти соответствующее число вариантов двузначных чисел и просуммировать все найденные числа. Для этого можно применить приём, который в программировании называется «рекурсией» [2–3].

Рекурсивная, то есть использующая рекурсию, функция КЧ, возвращающая количество двух- или трёхзначных чисел, сумма цифр которых составляет *сум*, может быть оформлена следующим образом (на школьном алгоритмическом языке):

¹Предлагаем читателям убедиться в справедливости приведённого далее правила самостоятельно (см. также [1]).

```
алг цел КЧ(арг цел k, сум)
  | k - количество разрядов в числе (2 или 3)
  | сум - сумма цифр числа
нач цел мин_посл, макс_посл, посл, всего
  | посл - последняя цифра числа
  | всего - меняющееся количество искомых чисел
  если k = 2
    то
      | Определяем границы значений последней цифры
      если сум > 9
        то
          мин_посл := сум - 9
          макс_посл := 9
        иначе
          мин_посл := 0
          макс_посл := сум
      все
      | Сразу можем определить значение функции
      знач := макс_посл - мин_посл + 1
    иначе | k = 3
      | Определяем границы значений последней цифры
      если сум > 9
        то
          макс_посл := 9
        иначе
          макс_посл := сум
      все
      если сум > 18
        то
          мин_посл := сум - 18
        иначе
          мин_посл := 0
      все
      всего := 0
      | Для всех значений последней цифры
      нц для посл от мин_посл до макс_посл
      | рекурсивно вызываем эту же функцию КЧ,
      | но для k = 2 и для оставшейся суммы (сум - посл),
      | и прибавляем значение функции
      | к «старому» значению величины всего
      всего := всего + КЧ(2, сум - посл)
      кц
      | Значение функции для трёхзначного числа
      знач := всего
  все
кон
```

С использованием этой функции основная часть программы решения подсчёта коли-

чества «счастливых» шестизначных билетов оформляется очень кратко:

```
алг Число_шестизначных_счастливых_билетов
нач цел кол, сумма_цифр | кол – искомое количество
кол := 0
| Для всех возможных значений суммы цифр
| вызываем функцию КЧ и, используя формулу,
| приведенную в начале статьи, определяем
| новое значение искомой величины кол
Нц для сумма_цифр от 0 до 27
    кол := кол + КЧ(3, сумма_цифр) * КЧ(3, сумма_цифр)
кц
вывод нс, кол
кон
```

Использованный подход к решению задачи (который, как показывают расчёты, приводит к значительному уменьшению времени работы программы) может быть использован для решения задачи в общем виде – определения количества k -значных «счастливых» номеров билетов. Для

общего случая рекурсивная функция КЧ должна быть немного изменена. Изменения следующие:

1) вместо случая $k = 3$ должен быть рассмотрен случай $k > 2$;

2) минимально возможные значения последней цифры должны рассчитываться по правилу:

```
если сум > (k - 1) * 9
то
    мин_посл = сум - (k - 1) * 9
иначе
    мин_посл = 0
все
```

3) при рекурсивном вызове должно использоваться не значение 2 (см. выше), а $k - 1$.

При этих изменениях завершающая часть функции примет вид:

```
...
если k = 2
то
    ...
иначе | k > 2
    | Определяем границы значений последней цифры
    если сум > 9
        то
            макс_посл := 9
        иначе
            макс_посл := сум
    все
    если сум > 18
```

```
    то
        мин_посл := сум - 18
    иначе
        мин_посл := 0
все
    всего := 0
нц для посл от мин_посл до макс_посл
    |Рекурсивно вызываем эту же функцию,
    |но для  $k$ , уменьшенного на 1,
    |и для оставшейся суммы ( $\text{сум} - \text{посл}$ )
    всего := всего + КЧ( $k - 1, \text{сум} - \text{посл}$ )
кц
    |Значение функции
знач := всего
все
кон
```

В основной части программы целесообразно использовать дополнительные переменные величины:

– *разрядность_номера* – число разрядов в номере билета (4, 6, 8

или 10);

– *разр* – число разрядов в половине номера.

Вся основная часть программы имеет вид:

```
алг Число_счастливых_билетов
нач цел разрядность_номера, разр, кол, сумма_цифр
    |Вводим разрядность номера и рассчитываем значение разр
вывод нс, "Задайте число разрядов в билете (4, 6, 8 или 10)"
ввод разрядность_номера
    разр := div(разрядность_номера, 2)
    |Определяем сумму квадратов значений функции КЧ
    кол := 0
нц для сумма_цифр от 0 до 9 * разр
        кол := кол + КЧ(разр, сумма_цифр) * КЧ(разр, сумма_цифр)
кц
    |Выводим ответ
вывод нс, "Количество билетов с ", разрядность_номера,
    "– значными' счастливыми' номерами равно ", кол
кон
```

Задания для самостоятельной работы

1. Разработав программу на языке программирования, которым вы владеете, определите количество «счастливых» четырех-, шести-, восьми- и десятизначных номеров.

2. Описанный в статье подход может быть реализован без использования рекурсии – расчёты значений

$\text{КЧ}_{\text{сум}}$ можно проводить от меньшего числа разрядов к большему. Посмотрите на таблицу 2. В ней приведены значения $\text{КЧ}_{\text{сум}}$ для различных значений суммы цифр и числа разрядов. Например, выделенное число 73 означает, что именно столько сущес-

твует трёхзначных чисел с суммой цифр 12.

Для однозначных чисел все значения известны (они равны 1). Для остальных каждое число в таблице получается, если просуммировать 10 элементов из столбца слева, стоящих

в одной строке с ним и сверху от него (оттенены значения, сумме которых равно число 73). Правда, есть исключение – для чисел, соответствующих сумме цифр, меньшей 9, значения определяются суммированием меньшего количества значений.

Таблица 2

Значение функции $KЧ_{\text{сум}}$

Сумма цифр	Количество разрядов в половине номера			
	1	2	3	4
0	1	1	1	1
1	1	2	3	4
2	1	3	6	10
3	1	4	10	20
4	1	5	15	35
5	1	6	21	56
6	1	7	28	84
7	1	8	36	120
8	1	9	45	165
9	1	10	55	220
10		9	63	282
11		8	69	348
12		7	73	415
13		6	75	480
14		5	75	540
15		4	73	592
16		3	69	633
17		2	63	660
18		1	55	670
19			45	660
20			...	...

После заполнения таблицы для нахождения количества «счастливых» билетов нужно просуммировать квадраты чисел, стоящих в том или ином столбце. Если нужно было

бы подсчитать количество 10-значных «счастливых» номеров, то потребовалось бы заполнить ещё один столбец до значения суммы, равного 45.

Разработайте соответствующий вариант программы.

3. Сравните время выполнения описанного в данной статье варианта программы, варианта программы согласно заданию 2 и вариантов, приведённых в [1]. Для этого используйте процедуры и функции, возвращающие текущее время, предусмотренное в системе программирования. Так как на продолжитель-

ность работы программы, кроме её особенностей, влияют и другие факторы, для каждого варианта проведите серию расчётов и используйте минимальное значение в серии.

В заключение заметим, что обсуждаемая задача может быть также решена с использованием так называемых «производящих функций», но соответствующую методику мы оставим за пределами данной статьи.

Литература

1. *Златопольский Д.М.* Считаем «счастливые» билеты, или Оптимизация программ. // Потенциал, 2016, № 3.
2. *Грацианова Т.Ю.* Эта страшная рекурсия: рекурсивные процедуры и функции. // Потенциал, 2014, № 5.
3. *Златопольский Д.М.* Программирование: типовые задачи, алгоритмы, методы. – М.: Бинوم. Лаборатория знаний, 2007.

Калейдоскоп

Калейдоскоп

Калейдоскоп

Чудо-звонок

В университетском городке Оксфорда есть не-большой музей приборов, в котором хранится немало уникальных экспонатов, например, первый британский гелиево-неоновый лазер, прибор для первых опытов по магнитной ориентации атомных ядер и другие, но наибольший интерес посетителей музея вызывает обычно самый простой из них – электрический звонок-долгожитель

Вот уже 175 лет он тихо и непрерывно позванивает с тех пор, как его купил (уже не новым) оксфордский профессор физики В. Уолтер в лондонской лавке научных приборов. Этот «вечный двигатель» устроен так. Под стеклянным колпаком находятся две цилиндрические батареи («вольтовые столбы» – по две тысячи пар электродов из фольги), залитые для герметичности серой и соединённые между собой. К ним присоединены стоящие внизу чашечки звонка, а между ними подвешен на нити металлический шарик. Плавнo покачиваясь, он ударяется то об одну, то о другую чашечку, поскольку они соединены с разными полюсами батареи. И эти его колебания, повторяясь, не прекращаются. Предполагается, что звон будет длиться ещё не одно столетие.

