

```
1 {
  File:    code.dpr

  Compiler. DELPHI
```

Задача. В наши дни в космосе находятся сотни спутников, и все они обмениваются данными. При этом система распознавания сигналов работает по схеме <Свой-Чужой>. Один из спутников отправляет запрос другому спутнику в формате двух целых чисел, а второй спутник отвечает первому так же двумя целыми числами. Первые два числа первого спутника представляют собой сумму цифр и количество цифр тех двух чисел, которыми должен ответить второй спутник. При этом в качестве ответа должны получиться числа, представляющие наибольшее и наименьшее возможные значения, которые могут быть сформированы по описанному выше методу.

Во входном файле INPUT.TXT записаны 2 натуральных числа S и K, представляющих сумму и количество цифр соответственно ( $K \leq 100$ ).

При этом гарантируется, что возможно составить хотя бы одно K-значное число, сумма цифр которого равна S.

В выходной файл OUTPUT.TXT выведите два числа – ответ второго спутника. При этом следует помнить, что все числа не имеют лидирующих нулей.

Input.txt	Output.txt
1 3	100 100
3 4	3000 1002

Решение. Калмыков Вадим (ProCrypt),  
г. Сургут, ЦНИТ "Северная Звезда",  
12:00, 04.06.2008

Source : <http://acm.dvpion.ru>

Editor. MSP, 06.12.09, 21:56, LIST 6.0

```
}
```

```
36 Program Code;
```

```
37 Const
```

```
38   Size = 100;
```

```
39 Var
```

```

40     S,           // сумма цифр чисел
41     K : Word;   // число цифр в числах

42     u,           // заполненная часть массива
43     n,           // число одинаковых цифр в последовательности
44     i, j : byte; // счетчики циклов

45     Min,
46     Max : array [1..Size] of byte; // числа, которыми ответит спутник

47 Begin
48     Assign (Input, 'input.txt');
49     Assign (Output, 'output.txt');

50     // инициализация массивов
51     for i := 1 to Size do
52         begin
53             Max [i] := 0;
54             Min [i] := 0;
55         end;

56     Reset (Input);
57     Read (S, K);
58     Close (Input);

59     {

```

Нам дана сумма цифр и длина числа. Другими словами, нам нужно разложить число  $S$  в сумму слагаемых, при этом из данной последовательности слагаемых получить наибольшее (наименьшее) число.

Чем больше цифра в старшем разряде, тем больше число, поэтому мы будем проверять, сколько раз может участвовать каждая цифра в разложении числа  $S$ , и заполнять в такой последовательности массив `Max`. Например, найдем максимальное разложение числа 25:  $9+9+7$ . При любой длине числа (разумеется не меньшей длины разложения, в данном случае 3) будет число 99700..000.

Для минимального числа дело обстоит иначе: чем меньше цифра в старшем разряде, тем меньше само число. Поэтому, мы переворачиваем максимальное число задом-наперед: 000..00799. Число не может иметь ведущих нулей, значит занимаем у 7 единицу и имеем: 100..00699 – мы получили минимальное число.

Теперь небольшой, но очень важный штрих в нашей программе: все хорошо, когда мы имеем последовательность девяток, а если последовательность  $9+8+8$ , спросит читатель?

Очень просто. У нас есть какое – то число  $S$ . Число девяток будет выражаться числом  $N = S \text{ div } 9$ , тогда остаток в  $S$  составит  $S \text{ mod } 9$ , следовательно, остаток меньше 9 и выражается одной цифрой

десятичной системы счисления. Если же девяток не будет в последовательности, то будет лишь одна цифра.

```
}  
  
86 // получаем максимальное число  
87 u := 0;  
88 for i := 9 downto 1 do  
89   begin  
90     n := S div i; // определяем, сколько раз встречается цифра i  
91     for j := (u + 1) to (u + n) do  
92       Max [j] := i;  
  
93       Inc (u, n);  
94       S := S mod i;  
  
95       if (S = 0)  
96         then  
97           Break;  
98     end;  
  
99 // теперь ищем минимальное число  
100 for i := 1 to K do  
101   Min [i] := Max [K - i + 1];  
  
102 i := 1;  
103 while (Min [1] = 0)  
104 do  
105   begin  
106     Inc (i);  
  
107     if (Min [i] <> 0)  
108       then  
109         begin  
110           Min [1] := 1;  
111           Dec (Min [i]);  
112         end;  
113     end;  
  
114 ReWrite (Output);  
  
115 // выводим максимальное число, потом - минимальное  
116 for i := 1 to K do  
117   Write (Max [i]);  
  
118 Write (' ');  
119 for i := 1 to K do  
120   Write (Min [i]);  
121 Close (Output);  
122 End.
```

Listing данной задачи опубликован в сети Internet по адресу

<http://www.Best-Listing.ru/color-10-task-636.html>

Sergey Mitrofanov, 07.09.14, 18:46

E-mail: [infostar@mail.ru](mailto:infostar@mail.ru)

© <http://www.Best-Listing.ru/>, 2006–2014