АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА ПОСЛЕДОВАТЕЛНОСТ

Означаваме с len(i) броя на цифрите в записа на i-тия елемент на редицата, а с cnt(i) броя на цифрите в десетичния запис на числото i. Очевидно е в сила рекуренцията len(i) = cnt(i) + 2 . len(i – 1). Намирането на K-тата цифра ще става чрез итеративно „движение по числото”, но не цифра по цифра, а с определена стъпка, която ще определяме чрез len(i). Започваме от i=N.

1. Ако K ≤ cnt(i), то намираме цифрата директно и я отпечатваме, като след отпечатването нулираме K и приключваме итерацията. В противен случай намаляваме K с cnt(i) и преминаваме към стъпка 2.
2. Ако K > len(i – 1), намаляваме K с len(i – 1) и i с 1.
3. Ако i=0, приключваме итерацията. В противен случай се връщаме към стъпка 1.

В края ако K > 0, отпечатваме -1.

Ето демонстрация на алгоритъма за N = 6 и K = 50:

6 5432112113211211432112113211211 5 432112113211211 4 3211 2113211211

(оцветените низове представляват „скоковете” по числото, които правим)

В началото ни остават K = 50 цифри и i = 6.

1. cnt(6) = 1 < K = 50, т.е. прескачаме 6-цата. Остават ни K = 50 – 1 = 49 цифри.
2. len(5) = 31 < K = 49, т.е. прескачаме целия низ, съответстващ на първото срещане на 5тия елемент от редицата (зеления). Остават ни K = 49 – 31 = 18 цифри, i = 6 – 1 = 5.
3. cnt(5) = 1 < K = 18, т.е. прескачаме 5-цата. Остават ни K = 18 – 1 = 17 цифри.
4. len(4) = 15 < K = 17, т.е. прескачаме целия син низ. Остават ни K = 17 – 15 = 2 цифри, i = 5 – 1 = 4.
5. cnt(4) = 1 < K = 2, т.е. прескачаме 4-ката. Остават ни K = 2 – 1 = 1 цифра.
6. len(3) = 7 > K = 1 и не можем да прескочим целия лилав низ, т.е. търсената цифра е някъде в него.
7. cnt(3) = 1 = K, т.е. търсената цифра е първата цифра в записа на числото 3, т.е. 3. Нулираме K и приключваме алгоритъма.

Единствената особеност, на която трябва да обърнем внимание, е че len(i) нараства много бързо. Тези N, за които len(N) ≤ 10^{15}, са всъщност само N ≤ 49. Затова за N > 49 можем да считаме, че len(N) = K + 1, тъй като при скоковете len(i) ни интересува с конкретната си стойност само, ако е по-малка или равна на K.

Сложността на алгоритъма е O(n).

Всякакви тривиални решения, симулиращи генериране на елементите на редицата, са със сложност от рода на O(len(n)), O(k) или по-голяма, т.е. не са достатъчно ефективни.

Автор: Елена Димитрова