Анализ на задача „Числото на звяра”

Динамичен подход #1 – сложност O(NK) – 50 точки

Нека дефинираме следната подзадача: колко са n-цифрените числа, които завършват с точно k (k ≤ K) поредни шестици и които **не** съдържат K поредни шестици (освен последните k шестици, когато k = K). Нека броят им е dp[n][k]. Стойността на dp[n][k] може да се пресметне чрез следната рекурентна зависимост:

dp[n][k] = dp[n – 1][k – 1], ако k > 0 (към числата с дължина n–1 и завършващи с k–1 шестици можем да добавим още една шестица) и  
dp[n][0] = в случая, когато последната цифра не е шест (към числата с дължина n – 1 и завършващи с i (за i < K) на брой шестици можем да добавим всяка цифра без шест),  
като граничните случаи са dp[1][0] = 8 и dp[1][1] = 1 (нулата не може да е първа цифра)

Отговорът на оригиналната задача е . Всяко n-цифрено число, което завършва с точно K шестици (и не съдържа други K поредни шестици преди тях), може да бъде допълнено до N на брой цифри по какъвто и да е начин (т.е. всяка от останалите N–n цифри може да е от 0 до 9). Така получените N-цифрени числа ще отговарят на условието на задачата, но което е по-важното – няма да могат да бъдат образувани по този начин за никое друго n, поради което не съществува опасност да преброим едно и също число два пъти. Още повече, обединениено на така получените числа за всяко n дава именно числата, които отговарят на условието.

Друг начин да гледаме на това решение е следният: започваме да градим числото цифра по цифра и в момента, в който се окажем с K поредни шестици, прекъсваме граденето и прибавяме 10N-n към отговора, тъй като можем да довършим числото до N цифри по 10N-n начина (приемайки, че дотук числото е n-цифрено).

Динамичен подход #1.1 – сложност O(N) – 100 точки

От горната зависимост се вижда, че dp[n][k] = dp[n – 1][k – 1] = dp[n – 2][k – 2] = … = dp[n – k][0] за k > 0. Използвайки това във формулата за k = 0, получаваме dp[n][0] = . Така може да разглеждаме едиствено случая, когато k = 0, свеждайки сложността по памет до линейна. Вижда се, че сумата в рекурентната зависимост за dp[n][0] всъщност представлява сумата па предходните K члена, която не е необходимо да бъде пресмятана на ново всеки път, а може да бъде поддържана с константна сложност при итеративна реализация на алгоритъма. Така сложността по време също става линейна.