

# НАЦИОНАЛЕН ЛЕТЕН ТУРНИР ПО ИНФОРМАТИКА

Стара Загора, 4 юни 2022 г.

Група В

## Анализ на задача Морзов код

**Първа подзадача.** Нека първо да помислим как можем да намерим броя на различните разчитания на  $S$ . Тъй като в тази подзадача  $K=0$  и думите и бележката са съставени само от тирета, тя се свежда до това да намерим по колко начина можем да представим дължината на  $S$  като сбор от дължините на думите. Тази задача може да се реши, като се използва метода динамично програмиране. Означаваме с  $dp[i]$  броя начини, по които може да разчетем  $i$  знака от бележката. Ако  $l[j]$  е дължината на  $j$ -тата дума, тогава:

$$dp[i] = \sum_{j=1}^M dp[i - l[j]]$$

Остава само да разберем как да намерим в колко разчитания участва всяка дума. Нека фиксираме началото на думата  $W_j$  на позиция  $i$ . Тогава броя разчитания с нея на позиция  $i$  е равен на  $dp[i - 1] \times dp[i + l[j]]$ . Така общият брой разчитания  $cnt_j$ , които я включват са:

$$cnt[j] = \sum_{i=1}^{N-l[j]} dp[i - 1] \times dp[N - (i + l[j])]$$

Сложност –  $O(N \times M)$

**Втора подзадача.** Тук вече думите не са съставени само от тирета и затова ни трябва алгоритъм, който да намира позициите на думите в низа. В случая те са съставени от най-много 100 символа, така че може да се използва квадратен цикъл (да гледаме от всяка позиция в низа дали може да се постави дадена дума). Нека представим низа си като граф, в който върховете са позициите в низа и всяка дума, която може да се разположи на позиции от  $i$  до  $i + l[j]$ , служи като еднопосочно ребро между двата съответни върха. Така като направим едно обхождане в дълбочина (DFS), придвижвайки се по създадените ребра, можем да намерим възможното разчитане (ако има такова) и да видим кои думи участват в него. Тъй като  $K \neq 0$ , ще се наложи да пазим по време

# НАЦИОНАЛЕН ЛЕТЕН ТУРНИР ПО ИНФОРМАТИКА

Стара Загора, 4 юни 2022 г.

## Група В

на обхождането коя е последната използвана дума, за да не правим невалидни разчитания.

Сложност –  $O(N \times M \times 100)$

**Трета подзадача.** Тук няма ограничение за дължината на думите, така че трябва да направим същото като във втора подзадача, но като използваме линеен алгоритъм за откриване на дума в низ – например Хеширане, КМР или Z-алгоритъм.

Сложност –  $O(N \times M)$

**Четвърта подзадача.** Решението на тази подзадача е основна крачка към авторското решение. Тя обединява решенията в първа и втора подзадача. Отново ще представим  $S$  като граф и ще използваме квадратния цикъл за намиране на думи в него (дължината им е  $\leq 100$ ), но тъй като този път възможните разчитания са повече от 1, ще трябва да използваме динамичното програмиране, за да ги намерим. Ако си фиксираме дадена дума  $W_j$  на позиция  $i$ , за да намерим колко разчитания има, ще трябва да намерим колко са възможните разчитания на подниза преди нея и да ги умножим по възможните разчитания на този след нея. За това ни трябва две динамични – едно, което да показва възможните разчитания на позиции от началото до  $i$  и едно, което да показва възможните разчитания на позиции от  $i$  до края. В този случай:

$$cnt[j] = \sum_{i=1}^{N-l[j]} dp1[i-1] \times dp2[i+l[j]]$$

Сложност –  $O(N \times M \times 100)$

**Пета подзадача.** Разширяваме динамичното от предната подзадача, като му добавим още един параметър към стейта – предната дума за  $dp1$  и следващата дума за  $dp2$ .

Сложност –  $O(N \times M^2)$

**Шеста подзадача.** Тази подзадача представлява авторското решение. Понеже няма ограничение за дължината на думите, както в

# НАЦИОНАЛЕН ЛЕТЕН ТУРНИР ПО ИНФОРМАТИКА

Стара Загора, 4 юни 2022 г.

Група В

трета подзадача, трябва да се ползва линеен алгоритъм за намиране на дума в низ.

Сложност –  $O(N \times M^2)$

Условие и анализ: Емилиана Димитрова

Тестове и решения: Александър Гатев