Анализ на задача “Код”

CodeIT, 2014-2015, Round6

Повечето състезатели усетиха, че задачата се решава посредством алчна стратегия.

Нека решаваме за даден интервал [a,b], т.е. текущият ни стринг е S(a, b). Искаме да конструираме минималната лексикографска редица, вземайки символи само от двата края на текущия ни стринг – отляво или отдясно. Ако двата края на текущия ни стринг са различни, очевидно вземаме по-малката от двете букви. Ако са еднакви, трябва да сравним предпоследните букви в последователността. Ако те също са еднакви, може да ни се наложи да продължим сравненията и т.н.

Всичко това може да се обощи по следния начин (\*):

Ако S(a, b) < reverse(S(a, b)), вземаме първата буква от стринга, която се намира в край а

Иначе, S(a, b) > reverse(S(a, b)), вземаме последната буква от стринга (в край b)

и продължаваме аналогично докато има още букви, които трябва да вземем.

Ограниченията бяха така подбрани, че сравнението на S с reverse(S) да може да бъде извършено линейно, което води до обща сложност О(N2). Задачата можеше да бъде дадена и с ограничения до N = 100,000 (примерно) и един от подходите би бил да се направи хеширане отпред назад и отзад напред и да се използва двоично търсене, за да се намери първата буква, която различава префикса от суфикса на текущия ни стринг (и така да се реализира сравнението на S с reverse(S)). Това води до сложност O(N log N). Друг подход, който води до същата сложност, е да се използва суфиксен масив.

Твърдението (\*) може да звучи интуитивно, но формалното доказателство не изглежда да е толкова тривиално.

Автор: Момчил Пейчев