АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА МАКСИМАЛНО ЛИЦЕ

Редица от стойности, разгледани както височини на стълбове на правоъгълници с основа 1 се нарича в литературата *хистограма*. Задачата се състои в това *да впишем правоъгълник с максимално лице в/под хистограма*. Представил съм 3 от направените 4 решения (четвъртото бе компрометирано от големите константи и засега не мога да го забързам въпреки теоретичното очакване за по-добра асимптотика от средното). Тук ги описвам накратко.

1. **Пълно изчерпване (maxarrea_Brute.cpp)**

Тривиалният алгоритъм е да се разгледа всеки интервал от стойности като основа на правоъгълник и дължината му да се умножи по минималната стойност на хистограмата в този интервал. Максимумът е най-голямото лице на правоъгълник измежду възможните $N+(N-1)+(N-2)+\cdots+1=O(N^2)$. Тъй като намирането на всеки минимум по тривиалния начин е със сложност $O(N)$, окончателната сложност на това тривиално решение е $O(N^3)$.

2. **Разделяй и владей (maxarre_DaC.cpp)**

Решението по схемата „разделяй и владей“ се основава на следното:

Твърдение 1*: Нека I е е колоната на хистограмата с най-малка височина,* $h[I]$ *е самата минимална височина, а с* $S[a,b]$ *да означим правоъгълника с максимално лице, който може да се впише в частта от хистограмата в интервала* $[a,b]$, $1\le I\le N$, $1\le a\le b\le N$. *Тогава* $S[1,N]=\max\{S[1,I-1],\ N.h[I],\ S[I+1,N]\}$.

Алгоритъмът следва твърдението на теоремата – намираме I и пресмятаме $N.h[I]$, след което рекурсивно намираме $S[1,I-1]$ и $S[I+1,N]$и определяме максимума на трите стойности. Ако намирането на минимума преди всяко рекурсивно извикване правим по тривиалния начин, сложността на алгоритъма буквално съвпада със сложността на прочутия Quick-sort, т.е. $O(N^2)$ в най-лошия случай (оценка, която е много трудно да бъде достигната на реален пример) и средна сложност $O(N.\log N)$, което го прави много по-бърз **на практика** от пълното изчерпване. Опитът да заместим тривиалното намиране на минимумите в отделните интервали с по-бързо – например чрез интервално дърво на минимумите – теоретично би трябвало да даде асимптотически по-бърз алгоритъм, но на практика големите константи на такъв алгоритъм го правят дори по-бавен от описания по-горе.

3. **Линеен алгоритъм (maxarrea_Linear.cpp)**

Линейният алгортъм се основава на следното:

Твърдение 2*: За всяко* $I\in[1,N]$ *да означим с* L_I *броя на колоните, пряко предшестващи отляво колоната* I, *чиито височини са по-големи от или равни на височината на I–тата колона, а с* R_I *– броя на колоните, пряко следващи надясно*

колоната I, чиито височини са по-големи от или равни на височината на тази на I–та колона. Тогава:

А) височината на максималния правоъгълник, в който I-тата колона участва, е $h[I]$;

Б) основата на максималния правоъгълник, в който I-тата колона участва, е $L_I + R_I + 1$.

Същността на алгоритъма е да се пресметнат за линейно време стойностите L_I и R_I, $I = 1, 2, \ldots, N$. Това се прави по един и същ начин за всяка от двете редици като реализацията може да се извърши чрез използване на стек.

Решението е минимумът от стойностите на намерените N лица.

Автор: Красимир Манев