

Дизайн на алгоритми

zag 1 Даден е масив от цели числа. Да се идексите на голямата елемента, такива че произведението е максимално.

инициализирай

SOLUTION ($A[1..n]$)

1. $s_1, s_2, b_1, b_2 \leftarrow 1$
2. for $i \leftarrow 1$ to n
3. if $A[i] > A[b_1]$
4. $b_2 \leftarrow b_1$
5. $b_1 \leftarrow i$
6. else if $A[i] > A[b_2]$
7. $b_2 \leftarrow i$
8. if $A[i] < A[s_1]$
9. $s_2 \leftarrow s_1$
10. $s_1 \leftarrow i$
11. else if $A[i] < A[s_2]$
12. $s_2 \leftarrow i$
13. if $A[s_1] * A[s_2] > A[b_1] * A[b_2]$
14. return (s_1, s_2)
15. else
16. return (b_1, b_2)

zag 2 Даден е масив с елем $\in \mathbb{N} \setminus \{0\}$. Да се идексът на най-малкото ест. число, което не е в масива

Реш. 1: sort $\Theta(n \lg n)$ + сравнение със 2 елем $\Theta(n) \Rightarrow T(n) = \Theta(n \lg n)$ $M(n) = \Theta(1)$

Реш. 2: counting sort $\Theta(n)$ + обх. за първа 0 $\Theta(n) \Rightarrow T(n) = \Theta(n)$ $M(n) = \Theta(n)$

Реш. 3: За масива 1 4 3 5 6 всички срещнати индекси умн. с -1 (всички). Индексът

SOLUTION ($A[1..n]$) на който има положително число е търсеният от нас индекс

zag 3 Даден е масив от цели числа. Да се предложат двойки алгоритми, идексащия

Index-предвар. обработка и заявкка, която трябва да е макс. бърза. Заявките

Query-заявка

са от вида: Каква е сумата на елем. от i до j , $i < j$

и i, j са валидни индекси

$$A: 1 \ 4 \ \underline{6 \ 3 \ 1 \ 5} \ 10$$

$$B: 1 \ \underline{5 \ 11 \ 14 \ 15 \ 10} \ 30$$

INDEX ($A[1..n]$)

1. $A[0] \leftarrow 0$
2. for $i \leftarrow 1$ to n
3. $A[i] \leftarrow A[i] + A[i-1]$
4. return $A[0..n]$

QUERY (i, j)

1. return $A[j] - A[i-1]$

zag.4 Даден е масив с дължина ≥ 2 . Има един "поп" елемент, като въпросът е: "попнат" елемент на индекс i ли е? След всеки въпрос "попнат" елемент се мести с ± 1 позиции (което е валидно). Предложете оптимален алгоритъм, който тамира "попни" на всяка стъпка точността на "попнат" позиция се сменя.
Ако точностите са различни и не го хваним, в ирот. случаи обхождаме 2 пати и след това идем 2 пати за идем за да сменим точността.

zag.5. Даден е масив от числа DCR, кийс по големината елемент

Решение 1: sort + A[n-k]

Решение 2: SELECT / Pick



zag.6 Масив съдържаща числата $0 \dots n$, без повторения, като има едно присвашо число. DCR, кое е то? Разлика на суми

Сортиране? Двуетно търсене

zag.7. Масив се търси възстановен, ако $A[1] < A[2] < \dots < A[i-1] < A[i] > A[i+1] > \dots > A[n]$
DCR най-големие. Двуетно търсене

zag.8 Да се предложи алгоритъм, който по подаден масив от цели числа ги подреди така, че всички отрицателни са вляво на всички положителни. Partition с pivot 0

zag.9. $A[1 \dots n]$ да се сортира. Ногава се k ($2 \leq k \leq n$) и обръща реда на елемент от $1 \text{go } k$:

пример $1643450 \quad k=3 \rightarrow 4613450$

zag.10 Дадени $A[1 \dots n]$ и $B[1 \dots m]$ сортирани DCR негататата на $A+B$