ДАА

упр.№8, 8 май 2020г.

Hello, world!

$$T(n) = 2\left(2T\left(\frac{n}{4}\right) + \frac{\frac{n}{2}}{\lg\frac{n}{2}}\right) + \frac{n}{\lg n} = 4T\left(\frac{n}{4}\right) + \frac{n}{\lg n - 1} + \frac{n}{\lg n} = 4\left(2T\left(\frac{n}{8}\right) + \frac{\frac{n}{4}}{\lg\frac{n}{4}}\right) + \dots$$

$$= 8T\left(\frac{n}{8}\right) + \frac{n}{\lg n - 2} + \frac{n}{\lg n - 1} + \frac{n}{\lg n} = \dots =$$

...последно за МТ

$$= n.T\left(\frac{n}{n}\right) + \frac{n}{2} + \frac{n}{3} + \dots + \frac{n}{\lg n} = n.\Theta(1) + n.\sum_{i=2}^{\lg n}\frac{1}{i} \asymp n.\Theta(1) + n.\lg\lg n \asymp n\lg\lg n$$

- с полагане: $T(n) = 2T(\sqrt{n}) + \lg n$

$T(n) = T(2^m) = S(m) \Leftrightarrow n = 2^m \Rightarrow \sqrt{n} = 2^{\frac{m}{2}} \Rightarrow T(\sqrt{n}) = T(2^{\frac{m}{2}}) = S(\frac{m}{2})$

$\Rightarrow S(m) = 2S(\frac{m}{2}) + m \Rightarrow k = 1$; сравн. $m^k = m$ с $f(m) = m \Rightarrow$ II сл. МТ $S(m) \asymp m\lg m$

$\Rightarrow T(n) = \lg n.\lg\lg n$

$T(n) = T(2^{2^m}) = S(m) \Leftrightarrow n = 2^{2^m} \Leftrightarrow m = \lg\lg n \Rightarrow \sqrt{n} = \sqrt{2^{2^m}} =$

$= 2^{\frac{2^m}{2}} = 2^{2^{m-1}} \Rightarrow S(m) = 2S(m-1) + 2^m.1 \Rightarrow S(m) = A.2^m + B.m.2^m$

- потенциално нерешима: $T(n) = 2T\left(\frac{n}{2}\right) + \frac{n}{\lg n}$

$\asymp m.2^m$

$\Rightarrow T(n) \asymp \lg\lg n.\lg n$

$\Rightarrow k = \log_2 2 = 1 \Rightarrow$ Сравн. n с $\frac{n}{\lg n}$

? I сл. $\exists ? \varepsilon > 0$: $n^{1-\varepsilon} \geq \frac{n}{\lg n} \Leftrightarrow \lg n \stackrel{?}{\geq} n^{\varepsilon} \Rightarrow$ НЕ!

? II сл. $f(n) \stackrel{?}{\asymp} n.\lg^t n$ за $t \geq 0 \Rightarrow$ НЕ!

? III сл. $\exists ? \varepsilon > 0$: $n^{1+\varepsilon} \leq \frac{n}{\lg n} \Rightarrow$ НЕ! Защото $n^{1+\varepsilon} > n > \frac{n}{\lg n} \Rightarrow$ НЕ!

$n = 2^m \Rightarrow S(m) = 2S(m-1) + 2^m.\frac{1}{m}$

$\Rightarrow$ НЕ!

Ако не знаеш какво да правиш с данните, сортирай ги

		Име	T(n)	M(n)	Бележки
С директни сравнения	Наивни	Insertion	n^2	1	стабилен; най-добър за малки ⇒ дъно на рек. алг.
		Selection	n^2	1	min. размествания
		Bubble	n^2	1	?
	Бързи	Heap	$n\lg n$	1	нестабилен; in-place
		Merge	$n\lg n$	n	подлежи на паралелизиране
		Quick	$n\lg n$ - средно n^2 - най-лошия	$\lg n$ - средно n - най-лошия	in-place; най-добър общо; варианти
Други		Counting	$n+c$	c	
		Radix	$n.d$	1	на практика $n\lg n$
		BogoSort	$n!$ очаквано (expected)	1	- - - - -

Зад.1: да се намери k-то по големина число в масив

- ако k е фиксирано число (const) → k локални пром.; трябва ли сорт.

  $\Rightarrow T(n) = \Theta(n)$; $M(n) = \Theta(1)$

- иначе
  - пак по същия начин (с опашка): $T(n) = \Theta(n.k) = O(n^2)$

    $M(n) = \Theta(k) = O(n)$
  - PICK → $T(n) = \Theta(n)$

    $M(n) =$
  - QuickSelect → $\Theta(n)$ средно

    $\Theta(n^2)$ в най-лошия
  - Median-of-Medians → $\Theta(n)$

Зад.2: Липсващо число в масив

$A[1..n]$, който съдържа числа от $[0;n]$ без повторения
$\Rightarrow$ съдържа всички числа без едно
$\rightarrow$ кое?

- ако е сортиран $\Rightarrow$ binary search $\Rightarrow T(n) = \lg n \quad M(n) = 1$

0 1 2 4 5 6 7 8
1 2 3 4 5 6 7 8
0 1 2 3 4 5 6 7

- иначе

– сортираме с HeapSort + binary search $\rightarrow T(n) = \underbrace{\Theta(n \lg n)}_{\text{сорт}} + \underbrace{\Theta(\lg n)}_{\text{търсене}} = \Theta(n \lg n)$

$M(n) = \Theta(1) + \Theta(1) = \Theta(1)$

– сумата на A изваждаме от $\dfrac{n(n+1)}{2} \rightarrow T(n) = \Theta(n)$

$n < 2^{64} \Rightarrow$ получ. остатък от модул $2^{64} \rightarrow$ т.к. $n < 2^{64}$, т.е. това е еднозначно

$M(n) = \Theta(1)$

– броене $\rightarrow$ масив от броячи $\rightarrow T(n) = \Theta(n)$

$M(n) = \Theta(n)$

Зад.3: дадени n числа в [1;k] – да се направи O(n+k) preprocessing, който да отговаря на въпросите:

- колко от числата са в [a;b] (където 1≤a≤b≤k)

$A[1..k]$ - масив от бройки → попълване за $O(n+k)$

→ $B[0..k] = \quad B[i] = A[i] + B[i-1]$; $B[0] = 0$ → попълване за $O(n+k)$

преф. суми на A ; $B[i]$ = брой срещания на числа $\le i$

На заявка $[a;b]$ ще върнем $B[b] - B[a-1]$ → $O(1)$

- колко е сумата на всички числа в [a;b] (където 1≤a≤b≤k)