3

Ретроспекция

Анализ на итер. алг.

- Сумиране

$$\sum_{i=1}^{n} i^a \approx n^{a+1}$$

$$\sum_{\substack{i=n \\ i/=2}}^{i=1} \frac{n}{i} \approx 2n-1$$

$\theta(n)$

- Анализ на алг за търсене
- Анализ на алг. за сортиране

Инверсия

arr[1...n] $i, j \in \{1...n\}$

$i < j \wedge arr[i] > arr[j]$

Алг за сортиране

- Времева сложност
- Пространствена сложност
- Устойчивост / стабилност

Метод на мехурчето

простр. сложност $\Theta(1)$

- устойчивост

$arr[1\ldots n] \xrightarrow{\text{сортиране}}$

$\pi(i)$ – индексът на i-тия елемент от arr в сортирания масив.

$arr = [5, 6, 2] \to [2, 5, 6]$

$\pi(1) = 2$ $\qquad$ $\pi(3) = 1$

$\pi(2) = 3$

Ще казваме, че алгоритъм за сортиране е устойчив, ако

$\forall i\ \forall j \in \{1\ldots n\}\quad i<j \wedge arr[i] = arr[j]$
$\Rightarrow \pi(i) < \pi(j)$

$\boxed{2 \qquad 2} \to \boxed{2\,2}$

Запазва относителната наредба

пример

Студенти в курса по ДАА

име, ФН, група

⋮ ⋮ ⋮

1) Сортиране по ФН.

2) устойчиво сортиране по група

име, ФН, група

сортирани по ФН { 1, 1, 1

сортирани по ФН { 2, ⋮, 2

сортирани по ФН. { 3, ⋮, 3

? Методът на мехурчето устойчив ли е?

$\boxed{\text{ДА!}}$

$\boxed{\ldots\ 2\ 2\ \ldots}$

swap? arr[i] > arr[i+1] ?

· false

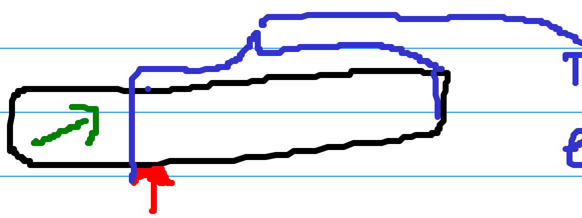
Сортиране с пряка селекция (Selection sort)

Анализ:

1) Времева сложност: $\Theta(n^2)$

2) Простр. сложност $\Theta(1)$

3) Устойчивост: Не!

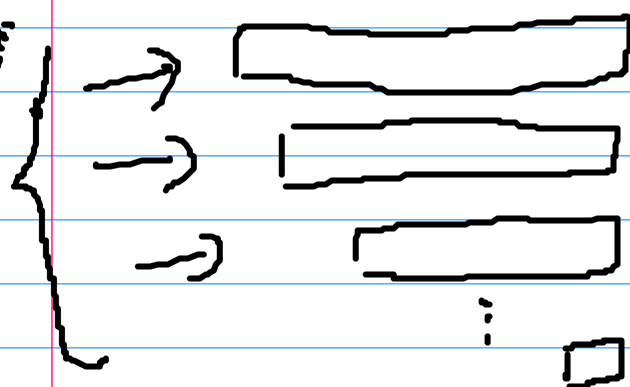


Търсим най-малкият елемент за ↑

1) Най-добър: $\Theta(n^2)$

Среден сл.: $\Theta(n^2)$

Най-лош сл.: $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$ $\Theta(n^2)$



търсим мин. елемент

[n елемента]

[n-1 елемента]

[n-2 елемента]

⋮

1 елемент

2) Пространствена сложност

$\Theta(1)$

3) Устойчивост: Не

3 2 3 1 → 1 2 3 3

swap

пример

Предимство?

! • Минимален брой размени/swap-ове

Домашно: Анализ на алгоритъма сортиране чрез вмъкване (Insertion Sort)

Анализът ще е подобен на анализа на метода на мехурчето.

Зад 1 Каква е времевата сложност на следния алгоритъм

$$\underbrace{n \to \frac{n}{2} \to \frac{n}{4} \cdots \to 1}_{\log(n)}$$

$\Theta(\log(n))$

⚠ $n \,\&\, (n-1) == 0$

Ако n е степен на 2

10000000 &
1111111
00000000

Зад 3 Каква е сложността на алгоритъма F

$F(n)$

$g(1) \quad g(2) \quad g(3) \; \ldots - g(n)$

g най-добрия сл. $\Theta(1)$

най-лошия сл. $\Theta(n)$

F най-лошия случай

$n * \Theta(n) = \Theta(n^2)$ Не!

Амортизиран анализ

n извиквания на g $\quad$ $n - \log(n)$

↳ "леки" $\Theta(1)$

↳ "тежки" $\Theta(n)$ $\quad$ $\log(n)$

$$n \geq k = \sum_{\substack{i=1 \\ i \text{ не степен на } 2}}^{n} 1 + \sum_{\substack{i=1 \\ i \text{ е степен на } 2}}^{n} i =$$

$$k + \sum_{i=1}^{n} i =$$

n извиквания на g

i е степен на 2

$$= \underbrace{k}_{\leq n} + \underbrace{1 + 2 + 4 + 8 \ldots \frac{n}{4} + \frac{n}{2} + n}_{2n-1} \leq$$

$$\leq 3n - 1$$

Колко на струва едно извикване на g?

$$\frac{3n-1}{n} \leq \boxed{3}$$

константа

$\Rightarrow$ Сложността на f:

$$\sum_{i=1}^{n} \theta(1) = \boxed{\theta(n)}$$

- Подробно на анализ на добавяне на елемент във вектор

• Пример за амортизирана сложност

↳ добавяне на елемент в черно-червено дърво

⋮

Анализ на рекурсивни алгоритми

• решаване на рекурентни ур-я

$a_0 \quad a_1 \cdot a_2 \; \ldots \; a_n$

$a_0 = 1$

$a_1 = 2$

$a_n = a_{n-1} + 3 a_{n-2}$

• Решаване на рек. ур-я

1) Налучкване + доказване

2) Развиване

3) метода с характеристичното ур-е

4) Master th.

5) други методи

Решаване на рек. ур-е
намиране на обща ф-ла
за n-ия член. (без рекурсия)

$a_n = 2^n + 9^n + n^2$ ⟵ пример

! При анализа на рек. алгоритми не ни интересува точното решение на ур-то, а неговата асимптотика!

Зад 4 Сложност на алгоритъма?

1) Съставяне рек ур-е

$T(n) = T(n-1) + T(n-2) + 1$

2) Решаване го

$\underbrace{T(n) - T(n-1) - T(n-2)}$ $= 1$

$\lambda^2 - \lambda - 1$ $\quad\quad P(n).a^n$

$\downarrow$ $\quad\quad n^0 . 1^n$

$$\lambda_1 = \frac{1-\sqrt{5}}{2} \qquad \lambda_2 = \frac{1+\sqrt{5}}{2}$$

$$C_n^k a^n$$

полином

$$\left\{ \frac{1-\sqrt{5}}{2}, \frac{1+\sqrt{5}}{2} \right\}_m$$

от вида в множеството $k+1$ пъти !

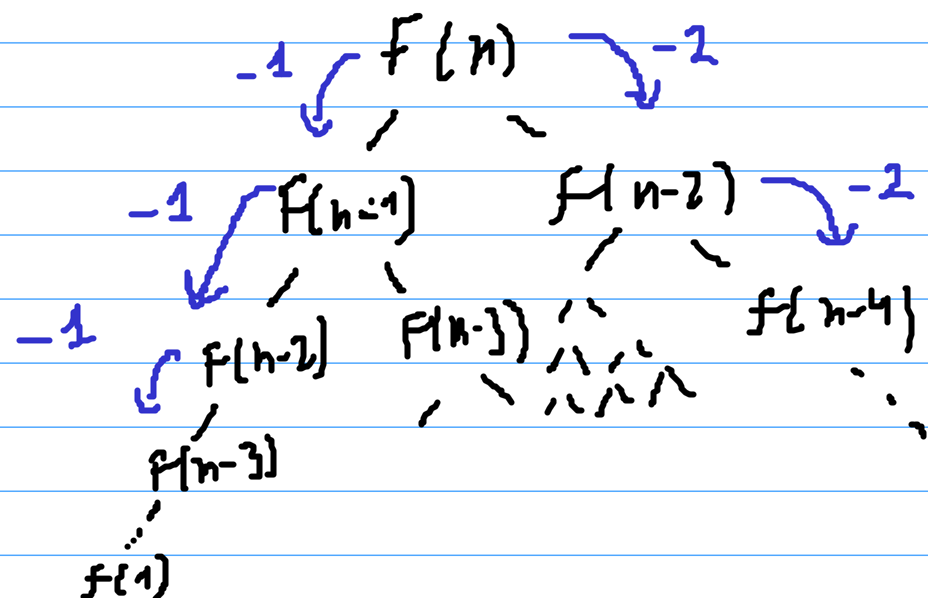
$$\cup \ \{1\}_m$$

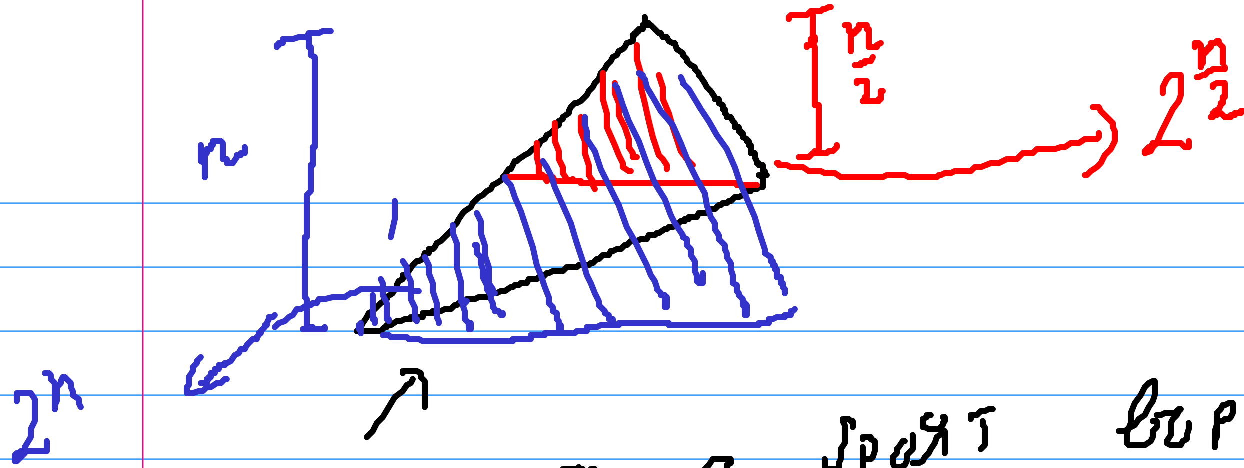
$$\left\{ \frac{1-\sqrt{5}}{2}, \frac{1+\sqrt{5}}{2}, 1 \right\}_m$$

$$\underline{A} \cdot \left(\frac{1-\sqrt{5}}{2}\right)^n + B \cdot \left(\frac{1+\sqrt{5}}{2}\right)^n + C \cdot 1^n$$

$$\boxed{\Theta\left(\left|\frac{1+\sqrt{5}}{2}\right|^n\right)}$$

$$1{,}618^n$$





сложността е брой върхове в дърво от 0

$$2^{\frac{n}{2}} \le 1{,}6..^{n} \le 2^{n}$$

Зад 5

$$T(n) = T(n-1) + \underbrace{1}_{\Theta(1)}$$

характеристично уравнение

$$\underbrace{T(n) - T(n-1)}_{} = 1 \qquad\qquad n^{0}\,1^{n}$$

$$\lambda - 1 = 0$$

корени

$$\{1\}_m \quad \cup \quad \{1\}_m$$

$$\{1, 1\}_m$$

$$T(n) = A n^0 1^n + B n^1 1^n$$

$$= A.1 + B.n = \Theta(n)$$

~~зад 6~~

$$T(n) = 3T(n-1) + 4T(n-2) + \underbrace{1}_{\theta(1)}$$

$$\underbrace{T(n) - 3T(n-1) - 4T(n-2)}_{\lambda^2 - 3\lambda - 4} = 1$$

$n^0 . 1^n$

$\lambda_1 = -1 \quad \lambda_2 = 4$

$\{1\}_M$

$\{-1, 4\}_M$

$\{-1, 1, 4\}_M$

$$T(n) = A(-1)^n + B\, 1^n + C\, 4^n$$

$$\theta(4^n)$$

~~зад 7~~ "оптимизация" на алг от задача 6

$\Rightarrow T(n) = T(n-1) + T(n-2) + 1$!

като фибоначи $\theta\left(\left(\frac{1+\sqrt{5}}{2}\right)^n\right)$

~~Зад 8~~

$+1$

$$T(n) = T(n-1) + T(n-2) + T(n-3) \dots T(0) =$$

$$= \sum_{i=0}^{n-1} T(i)$$

$$T(n-1) = T(n-2) + T(n-3) \dots T(0) + 1$$

$$T(n) - T(n-1) = T(n-1)$$

$$T(n) - 2T(n-1) = 0$$

$$\lambda - 2 = 0$$

$$\lambda = 2$$

$$T(n) = A \cdot 2^n \Rightarrow \boxed{\theta(2^n)}$$

~~Зад 9~~

Сложността на funcHelp(n,n)

- На всяка стъпка имаме точно две рекурсивни извиквания.
- На всяко рекурсивно извикване намаляваме точно един от двата параметъра
- Дъното е когато двата параметъра са 0

$R(0,0) = \Theta(1)$

$R(0,j) = 2R(0,j-1)+1 \rightarrow T(k) = 2T(k-1)+1$

$R(i,0) = 2R(i-1,0)+1 \rightarrow T(k) = 2T(k-1)+1$

$R(i,j) = R(i-1,j) + R(i,j-1)+1 \rightarrow T(k)=2T(k-1)+1$

Полагаме $k = i+j$

$T(k) = 2T(k-1)+1$

$T(k) - 2T(k-1) + 1$

$\lambda - 2 = 0 \qquad \{1\}_m$

$\lambda = 2 \qquad \cup$

$\{2\}_m \qquad \{1,2\}_m$

$T(k) = A1^k + B.2^k \qquad \Theta(2^k).$

$funcHelp(\underset{i}{n}, \underset{j}{n}) \rightarrow i+j = 2n \quad \Theta(2^{2n}) =$

$= \boxed{\Theta(4^n)}$

~~Зад 10~~ $T(0)=1$

$$T(n) = \underbrace{T(n-1)}_{T(n-2)+2^{n-1}} + 2^n$$

$\boxed{10.1}$ Решение с разбиване

$$\underbrace{T(n-2)}_{T(n-3)+2^{n-2}} + 2^{n-1}$$

$$T(n-3) + 2^{n-2}$$

$$\vdots$$

$$\underbrace{T(0)}_{1} + 2^1$$

$$2^n + 2^{n-1} + 2^{n-2} \dots + 2^1 + 1$$

$$= 2^{n+1} - 1 \qquad \boxed{\Theta(2^{n+1})} \quad (\doteq \Theta(2^n))$$

$\boxed{10.2}$ Решение с характ. уравнение

$$T(n) = T(n-1) + 2^n$$

$$\downarrow \qquad\qquad \downarrow \; n^0 (2)^n$$

$$\lambda - 1 = 0 \qquad \cup \qquad \{2\}_m$$

$$\{1\}_m$$

$$\{1, 2\}_m$$

$$T(n) = A \cdot 1^n + B \cdot 2^n \qquad \boxed{\Theta(2^n)}$$