

Семинар 1 - Част 2

Пример - Binary search

* best case - елементът е в средата $\Rightarrow \Theta(1)$

* worst case - елементът го няма \Rightarrow Колко стъпки са ни необходими, за да го установим?

0 итер. \Rightarrow вход n ($n/2^0$)

1 итер. \Rightarrow вход $n/2$ ($n/2^1$)

2 итер. \Rightarrow вход $n/4 = n/2^2$

\vdots

k итер. \Rightarrow вход $n/2^k \approx 1 \Rightarrow 2^k \approx n \Rightarrow k \approx \log n \Rightarrow \Theta(\log n)$
(поне n)

Алгоритми за сортиране

1) Свойства

* стабилност - запазва се относителната наредба на елементите с еднакви стойности на ключа за сортиране
пр.: студенти с оценки

Алекс - 5

Боби - 3

Вили - 4

Тери - 6

Дани - 5

\Rightarrow

Тери - 6

Алекс - 5

Дани - 5

Вили - 4

Боби - 3

} запазили са
относителната
наредба

* адаптивност - взима се предвид текущата наредба на елементите (напр. ако масивът е почти сортиран, ще се направят по-малко стъпки)

* in-place - не се използва допълнителна памет за сортирането.

2 Selection sort

* best case - сортиран вход $\Theta(n^2)$

$i=0 \Rightarrow n-1$ стъпки

$i=1 \Rightarrow n-2$ стъпки

$i=2 \Rightarrow n-3$ стъпки

....

$i=n-1 \Rightarrow 1$ стъпка

$$\Rightarrow 1+2+\dots+(n-2)+(n-1) = \frac{n(n-1)}{2} \approx n^2$$

* worst case - сортиран вход в обратен ред $\Theta(n^2)$
аналогично на best case

* стабилен - НЕ \rightarrow пр. $2' 2'' 1 \Rightarrow 1 2'' 2'$

* адаптивен - НЕ \rightarrow не е бърз от наредбата на входа

* in-place - ДА

3 Insertion sort

* best case - сортиран вход $\Theta(n)$

* worst case - сортиран вход в обратен ред $\Theta(n^2)$

$i=1 \Rightarrow 1$ стъпка

$i=2 \Rightarrow 2$ стъпки

....

$i=n \Rightarrow n$ стъпки

общо $1+2+\dots+n \approx n^2$ стъпки

* стабилен - ДА

* адаптивен - ДА \rightarrow приемаме, че на i -тата стъпка масивът е сортиран от 0 до $i-1$ позиции и така за почти сортиран вход се намалява бр. стъпки

* in-place - ДА

4) Bubble sort

* best case - сортиран вход $\Theta(n)$

$i=0 \Rightarrow n-1$ стъпки \rightarrow last swapped Index = 0 \Rightarrow stop / няма размяствания /

* worst case - сортиран вход в обратен ред $\Theta(n^2)$

$i=0 \Rightarrow n-1$ стъпки

$i=1 \Rightarrow n-2$ стъпки

...

$i=n-1 \Rightarrow 1$ стъпка

общо $\frac{n(n-1)}{2} \approx n^2$ стъпки

* стабилен - DA \rightarrow търсим строго неравенство м/у всеки два съседни елемента ($arr[j] > arr[j+1]$)

* адаптивен - зависи (како разбъркане под почти сортиран масив)

пр. 1) 1 2 3 5 4 $\rightarrow \Theta(n)$ стъпки

2) 2 3 4 5 1 $\rightarrow \Theta(n^2)$ стъпки

* in-place - DA

Алгоритъм	Сложност	Стабилност	Адаптивност	In-place
Selection sort	$\Theta(n^2)$	X	X	✓
Insertion sort	$\Theta(n^2)$	✓	✓	✓
Bubble sort	$\Theta(n^2)$	✓	зависи	✓