Algorithmen & Datenstrukturen

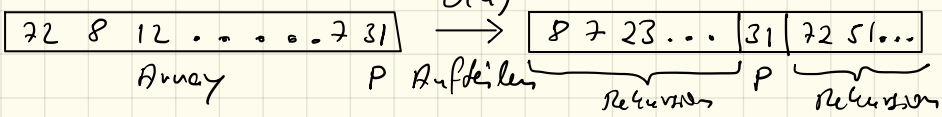
Herbst 2021

Vorlesung 15

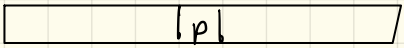
Auswahlproblem, Median

Quicksort nochmal:

Sortiere Array mit n Zahlen



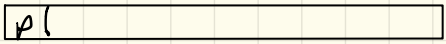
Ideal: p ist Median



← eine Hälfte kann eins kleiner sein, aber egal

$$T(n) = 2\,T(n/2) + c \cdot n \quad \Rightarrow \quad O(n \log n)$$

Worst case: z.B. immer



$$T(n) = T(n-1) + c \cdot n \quad \Rightarrow \quad O(n^2)$$

sehr unwahrscheinlich ⇒ average case $O(n \log n)$
(wenn zufälliges p) (Analyse: nächstes Semester)

Frage: Wie teuer wäre es den Median zu berechnen? z.B. Median in $O(n)$
⇒ Quicksort immer $O(n \log n)$

Auswahlproblem (Selection problem)

gegeben: Array $A[1], \ldots, A[n]$

gesucht: i-kleinstes Element Spezialfall: Median $= \lceil n/2 \rceil$-kle'instes

Idee 1: i mal Minimum entfernen $\Theta(in)$

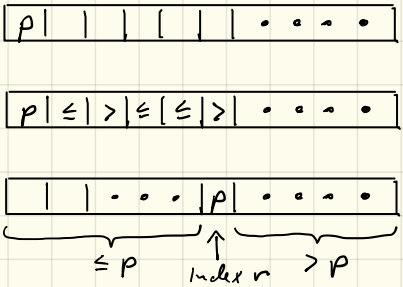
Idee 2: Sortieren, i-tes ausgeben $\Theta(n \log n)$

Idee 3: Pivotieren, rekursiv

$O(1)$ 1.) wähle Pivot (z.B. erstes)

$O(n)$ 2.) zähle Elemente $\leq p$ ($= r$ viele)

$O(n)$ 3.) teile A



4.) $i = r$: return p
$i < r$: suche i-tes links
$i > r$: suche $(i-r)$-tes rechts

Heißt "quickselect": wie quicksort aber Rekursion nur auf einer Hälfte

Laufzeit: worst case $T(n) = T(n-1) + \Theta(n)$

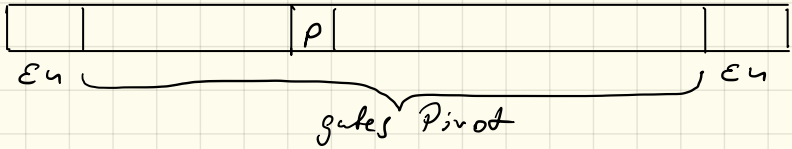
$\Rightarrow T(n) = \Theta(n^2)$

aber average case $T(n) = \Theta(n)$ (ohne Beweis)

Was ist ein gutes Pivotelement?

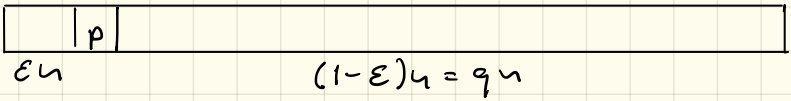
Antwort: reduziert Länge des Arrays in jeder Iteration mindestens um Faktor $q < 1$

Bild: Nach Aufteilen:



$0 < \varepsilon < \frac{1}{2}$ fix!
z.B. $\varepsilon = 0.1$

also schlimmstenfalls:



Dann dann: $T(n) \leq T(qn) + c \cdot n$, c konstant
$(q = 1-\varepsilon < 1)$

Teleskopieren: $\leq T(q^2 n) + cqn + cn$

. . . . ($\log_{1/q}(n)$ Schritte)

$\leq T(1) + n \cdot c \cdot$ geometrische Reihe in q

$= \mathcal{O}(n)$

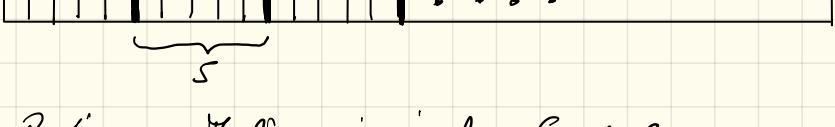
Wie schaffen wir das deterministisch?
(d.h. immer, nicht nur im average case)

Algorithmus "Median der Medians"
(Blum, Floyd, Pratt, Rivest, Tarjan 1973)

Idee: clevere Pivotwahl (neuer Schritt 1.)

1.) wähle Pivot wie folgt:

O(1) a.) Betrachte Array in Fünfergruppen:



O(n) b.) Bestimme Median in jeder Gruppe

O(n) c.) A' = Array der Gruppenmediane (Länge $\lceil \frac{n}{5} \rceil$)

d.) Bestimme Median von A': Median = Pivot p

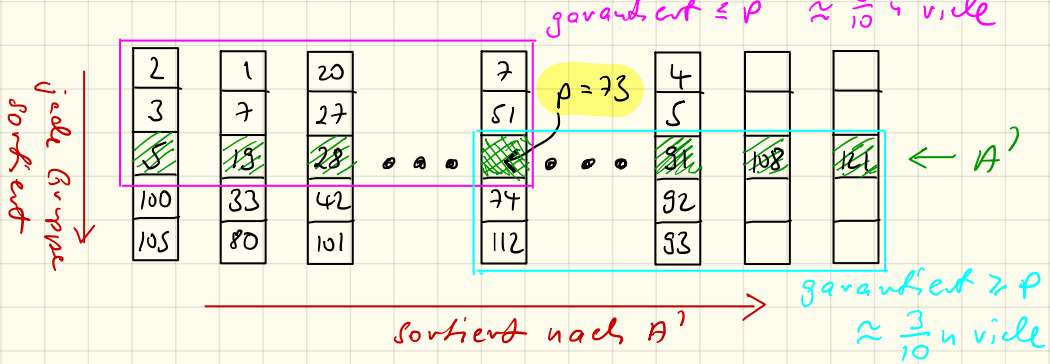
wie? Rekursion: $\lceil \frac{1}{2} \lceil \frac{n}{5} \rceil \rceil$ - tes Element von A'

2.) - 4.) wie vorher (= Aufteilen, Rekursion links oder rechts)

Also zwei rekursive Aufrufe!
p ist Median der Gruppenmediane

Wie gut ist p? (= wieviele Elemente in A sind garantiert kleiner und größer?
= was ist ε von voriger Seite?)

Bild: wir malen A so: (A ist nicht so sortiert)



Laufzeit: (Skizze) $T(n) \leq T(\frac{n}{5}) + T(\frac{7}{10} n) + cn$

$n - \frac{3}{10} n$

Zeige: $T(n) \leq an$ per Induktion, a konstant

Anfang: $T(n) \leq a \cdot n \quad \forall\, n \leq n_0$ (z.B. $n_0 = 100$)
wähle a so daß das gilt und $a > 10c$

Schritt: Annahme $T(k) \leq ak \quad \forall\, k < n$

$$\Rightarrow T(n) \leq a \frac{n}{5} + a \frac{7}{10} n + cn$$

$$= a \cdot \frac{9}{10} n + cn$$

$$\leq an$$

□

Insbesondere: Medianen geht in $O(n)$

In der Praxis: zufälliges Pivot funktioniert gut

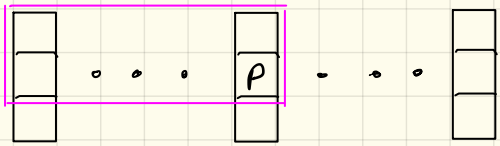
Warum Fünfergruppen?

$$T(n) \le T\left(\frac{n}{5}\right) + T\left(\frac{7}{10}n\right) + cn$$

Summe $= \frac{9}{10}n < n$ Wichtig im Beweis!

Dreiergruppen:

$\approx \frac{1}{3}n$

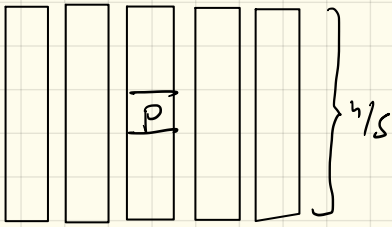


Rekurrenz $T(n) \le T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + cn$

Summe $= n$

Vierergruppen: $T(n) \le T\left(\frac{n}{4}\right) + T\left(\frac{3n}{4}\right) + cn$

Ab $g \ge 5$ klappt es, g konstant!

$n/5$ - Gruppen: $T(n) \le 5\,T\left(\frac{n}{5}\right) + T\left(\frac{7}{10}n\right) + cn$

zu teuer!