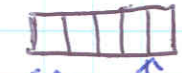
6.897
L 21.3
Apr. 26, 2005

Rank [Jacobson – FOCS 1989]

- can use lookup table for bitstrings of len. $\frac{1}{2}\lg n$
  $\Rightarrow O(\sqrt{n}\ \lg n\ \lg\lg n)$ space (bits)
  ($\sqrt{n}$: bitstring; $\lg n$: query i; $\lg\lg n$: answer)
- but if we split into $n/\frac{1}{2}\lg n$ pieces, need $\Theta(n)$ space
- instead split into $n/\lg^2 n$ pieces:

$\Rightarrow O(\frac{n}{\lg^2 n}\cdot\lg n) = O(\frac{n}{\lg n})$ bits

- split a $(\lg^2 n)$-size chunk into $(\frac{1}{2}\lg n)$-size subchunks



$\Rightarrow O(\frac{n}{\lg n}\cdot\lg\lg n)$ bits

- at bottom level, use lookup table

Select [Clark & Munro – Clark's PhD 1996]

- store index of every $(\lg n\ \lg\lg n)$th 1 bit
  $\Rightarrow O(\frac{n}{\lg n\lg\lg n}\cdot\lg n) = O(\frac{n}{\lg\lg n})$ bits
- within group of $\lg n\ \lg\lg n$ 1 bits, say range r
- if $r \geq (\lg n\ \lg\lg n)^2$ then write down 1-bit indices
  $\Rightarrow O(\frac{n}{(\lg n\lg\lg n)^2}\cdot(\lg n\lg\lg n)\cdot\lg n) = O(\frac{n}{\lg\lg n})$ bits
- else store relative index of every $(\lg r\ \lg\lg n)$th bit ($\geq \lg n\lg\lg n$) — $\lg\lg n$ bits each
  $\Rightarrow O(\frac{n}{(\lg\lg n)^2}\cdot\lg\lg n) = O(\frac{n}{\lg\lg n})$ bits
- like first step: repeat others to reduce range to $(\lg\lg n)^4$.
- lookup table on $\frac{1}{2}\lg n$-bit words →
  $\Rightarrow O(\sqrt{n}\ \lg n\cdot\lg\lg n)$ bits
  ($\sqrt{n}$: #words; $\lg n$: i; $\lg\lg n$: answer)