Thm: For set cover greedy picks at most $O(\lg n)$ of OPT.

Pf. Suppose OPT is k.

OPT has k sets $\Rightarrow \exists S$ s.t. $|S| \geq n/k$.

(Otherwise OPT can not cover all of the elements)

Therefore in the first step our algorithm will cover at least n/k elements.

Fact: More generally, for any $A \subseteq [1,\dots,n]$ of elements, $\exists S$ s.t. S covers at least $1/k$ fraction of elements in A: $|S \cap A| \geq \frac{|A|}{k}$.

BC. OPT covers A using k sets.

In the second step we cover at least $1/k$ fraction of remaining elements, i.e, at least $\frac{n - n/k}{k}$

remaining elements after step 2 $\leq n - n/k - \frac{n - n/k}{k} = n(1-1/k)^2$.

Supp we run for $t = k \ln n$ step.

remaining elements is $\leq n(1-1/k)^t = n(1-1/k)^{k \ln n}$

$= n\left(\underbrace{(1-1/k)^k}_{\leq 1/e}\right)^{\ln n} \leq n\, e^{-\ln n} = 1.$

IH: $x_i \cdots x_j$ is max of $x_1 \cdots x_{n-1}$

$x_k \cdots x_{n-1}$ is max suff

IS. Supp OPT is $x_a \cdots x_b$.

- Case 1: $b < n$: Then $x_a \cdots x_b$ is also OPT of $x_1 \cdots x_{n-1}$

  must have $a = i$ $b = j$.

Case 2: