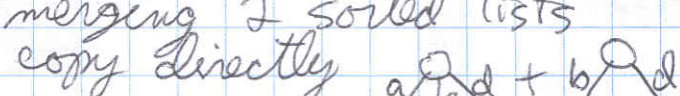
Merging odd & even suffixes:

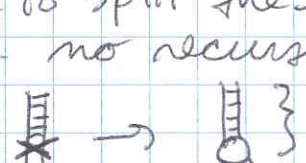
6.897
L11.4
Apr. 2, 2003

- build suffix trees out of 2 suffix arrays
- "overlay" trees to try to merge
  - overlay roots — like merging 2 sorted lists
  - uncommon branches — copy directly



  - common branches:
    - would be easy if uncompacted
    - compacted. need to know how far the edges match (after trimming to shorter length)

    - if match all the way, cool — just recurse
    - if not, need to split the edge early & then no recursion — just copy



- one way to tell whether edge labels completely match:
  - preprocess to store fingerprint($T[i:]$) $\forall i$
  - ⇒ can compute $fp(T[i:j])$ as $\frac{fp(T[i:]) - fp(T[j:])}{|\Sigma|^j}$
  - test fingerprints of edge labels for equality.
- still don't know length of match — but do recover topology of suffix tree
- compute suffix links via lca's
  - ⇒ get edge labels lengths ⇒ get edge labels
  - Also compute orders of node splits by comparing labels
  - ⇒ get suffix tree.
    (convert to suffix array if not last step)