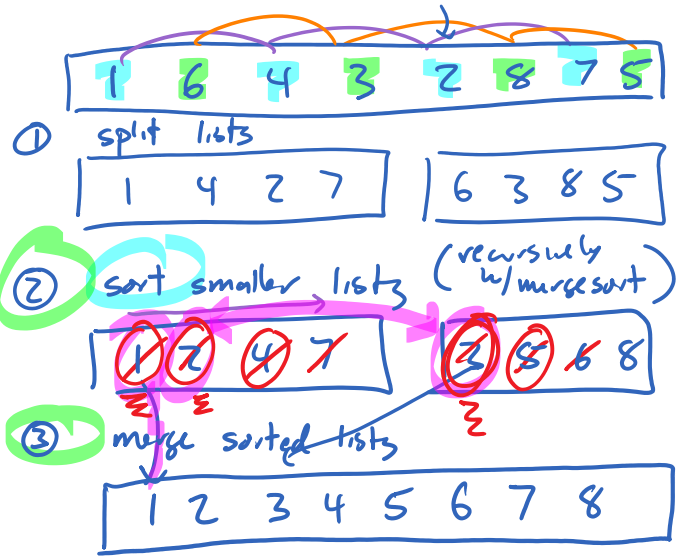
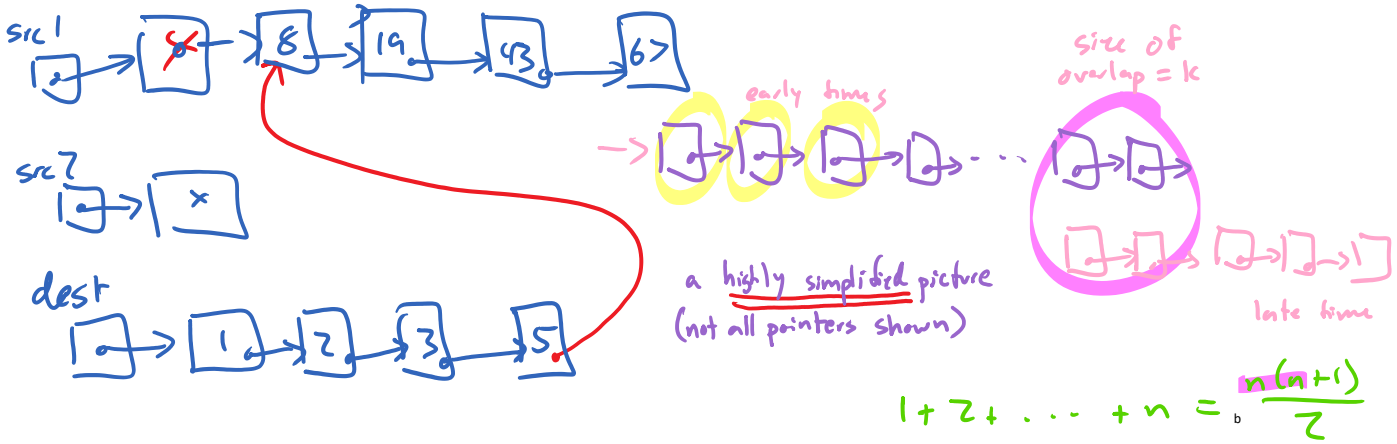
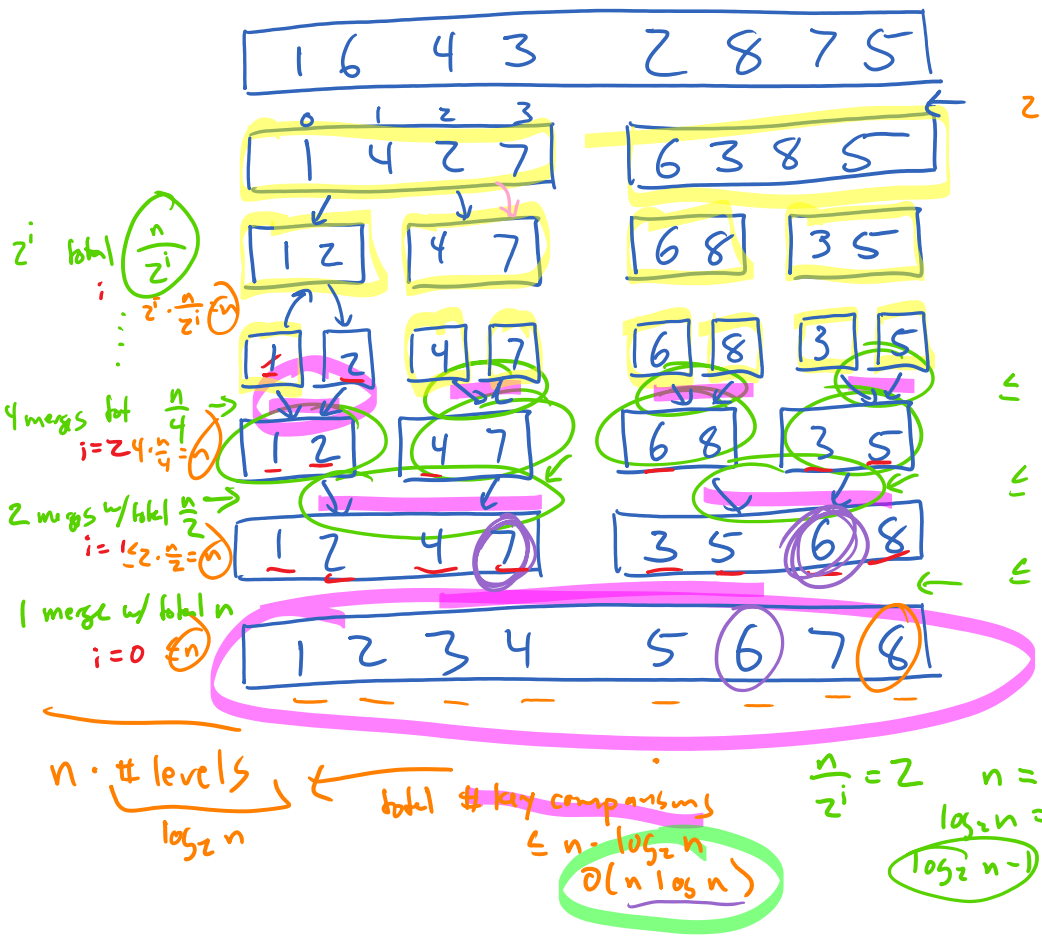
Mergesort






```
array list                                                    linked list
O(n) iters  for (size_t i = 0; i < location_list_size(l); i++)  O(n) iters
            {
O(1)            location *loc = location_list_get(l, i);      O(n)
O(1)            // get lat, lon and print them                  ---
            }                                                  O(n²) total
---
O(n) total
```

Example



worst case # key comparisons in merge operation

2 lists w/ total n elements
worst-case # comparisons
$n-1$ $\leq n$

$\leq 2 + 2 + 2 + 2 \leq 8$

$\leq 4 + 4 \leq 8$

$\leq 8 \leq 8$

$\leq 3 \cdot 8$

$n \cdot$ # levels $\leftarrow$ total # key comparisons $\leq n \cdot \log_2 n$

levels $= \log_2 n$

$O(n \log n)$

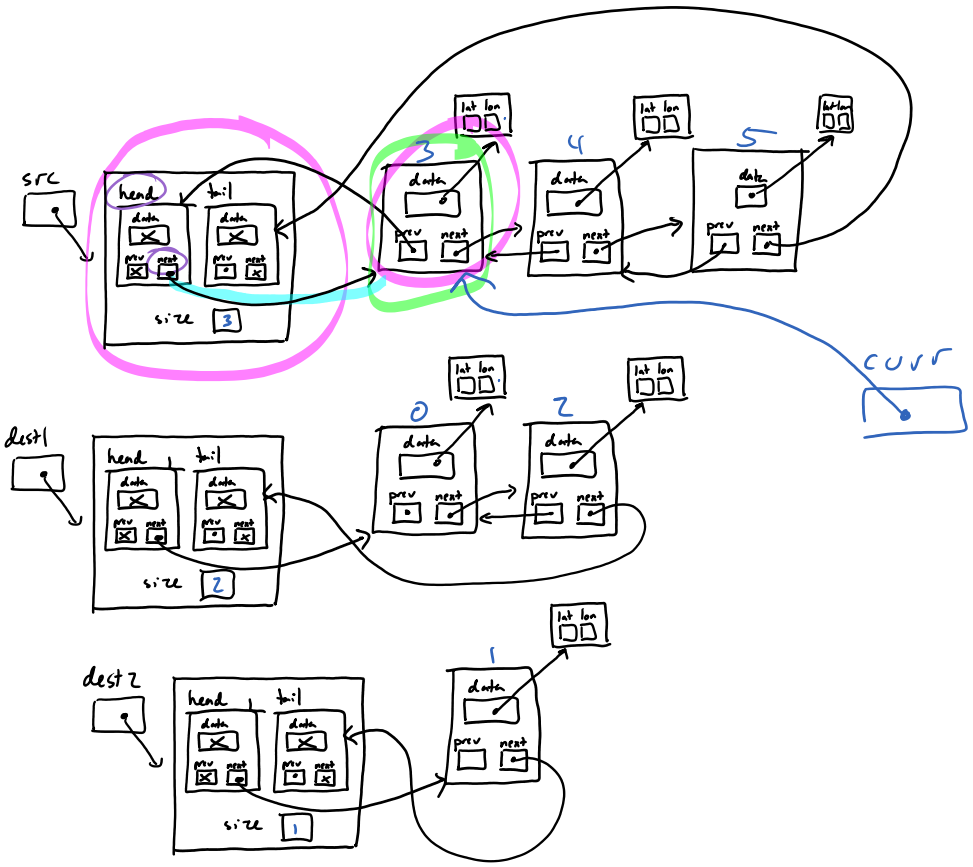
$\frac{n}{2^i} = 2$ $n = 2^{i+1}$

$\log_2 n = i + 1$

$\log_2 n - 1 = i$

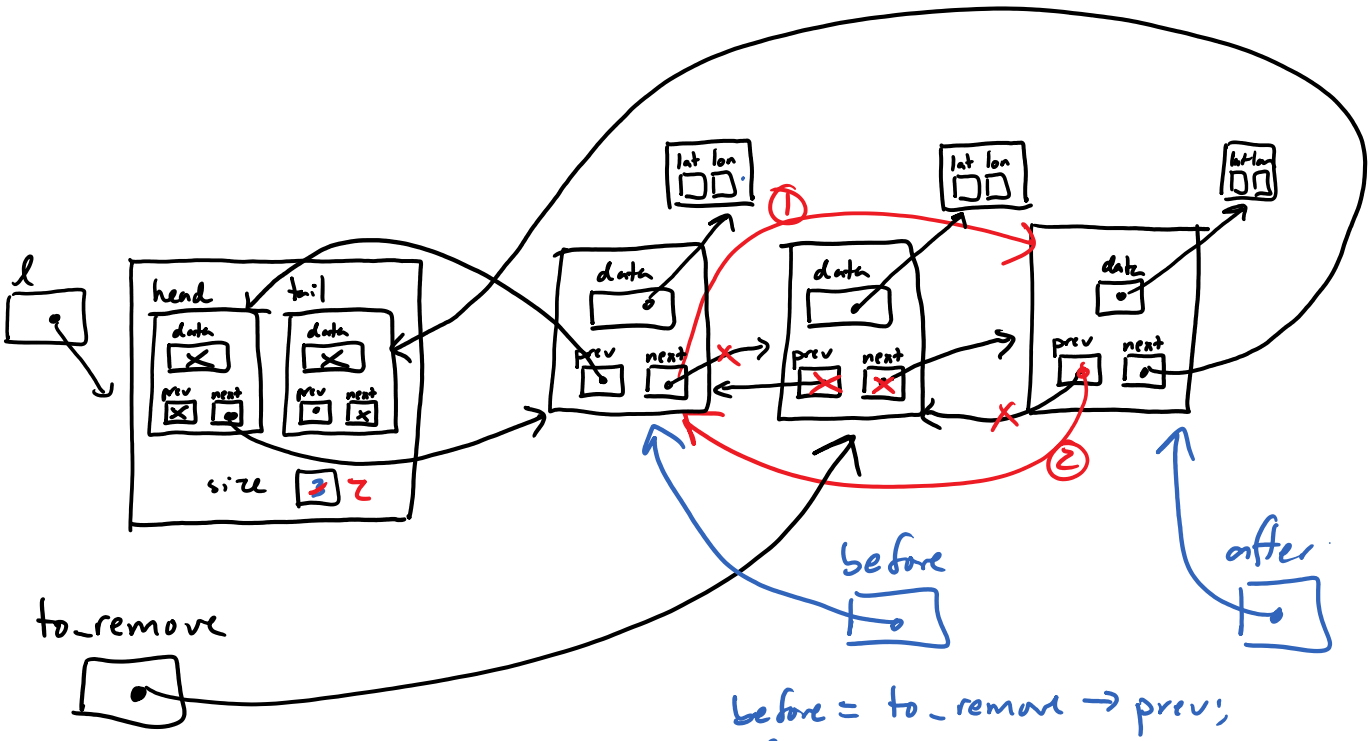
$\log_b a = \frac{\log a}{\log b}$

Split



curr = src → head . next;

Remove Node




```
before = to_remove -> prev;
after = to_remove ->next;
to_remove -> prev = NULL;
to_remove -> next = NULL;
(1) before ->next = after;
(2) after -> prev = before;
    l -> size-- ;
```