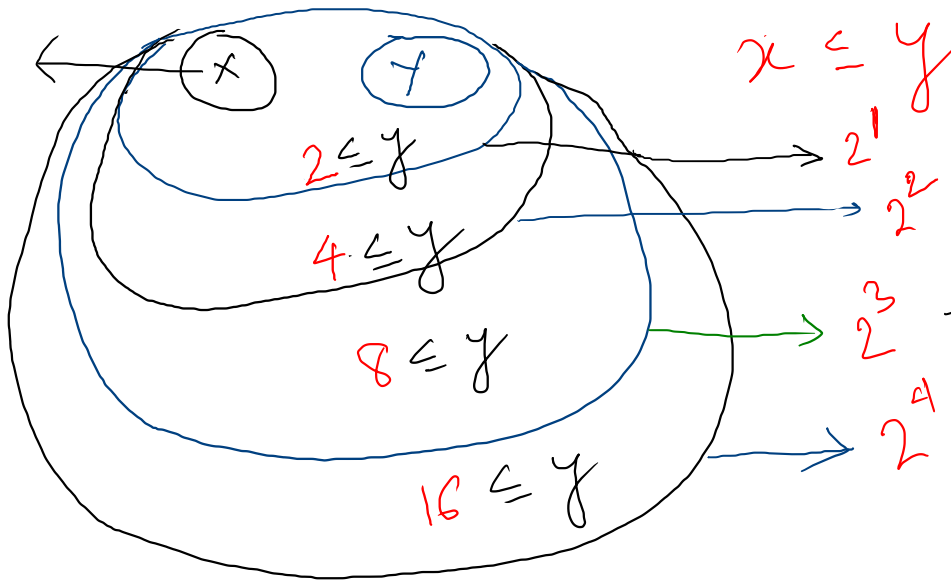


Time Complexity analysis for merging smaller list to larger list.

Assumption:

1. Let's just focus on one element i.e. x
2. x always lies in smaller list.
3. Initial size of smaller list is one.

Set size = 1



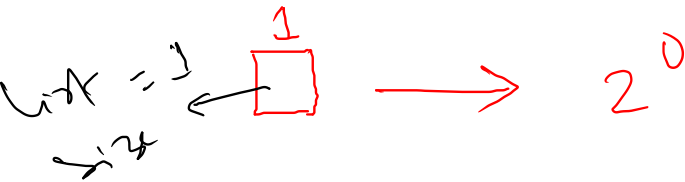
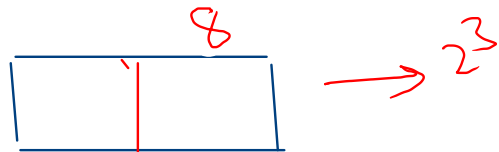
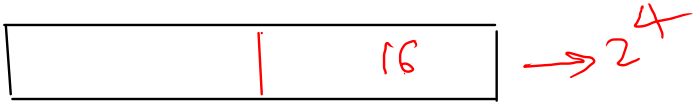
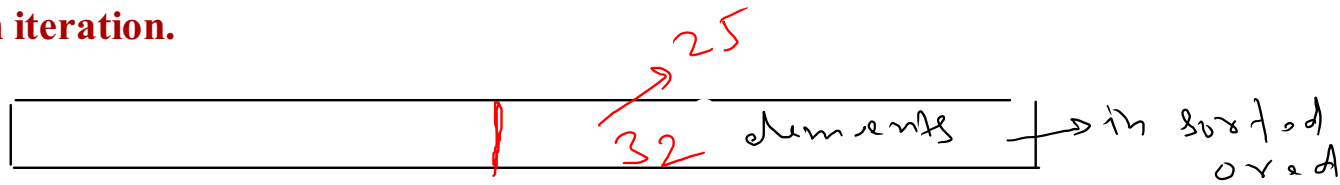
Objective for X: We need to track the copy operation for X in each merge operation.

When size of merged list reached to 8; x is copied 3 times.

number of copy operation for x = $\log_2(\text{size of merged list})$

$= \log_2 n$

Time complexity analysis for situation when working area is getting divided by half at each iteration.



We are getting list size of 1 after 5 divide operations

$$\begin{aligned} \text{No. of divide operations} &= \log_2 (\text{list size}) \\ \text{in terms of list size} &= \log_2 n \\ &= \log_2 n \end{aligned}$$