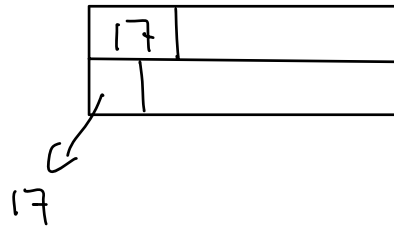
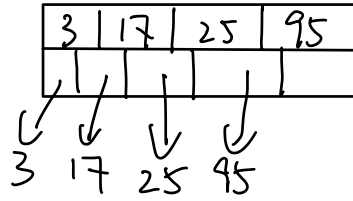


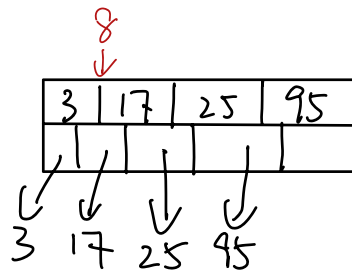
Insertion: 17



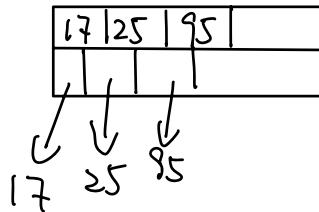
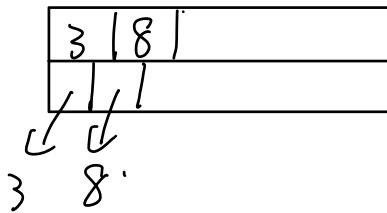
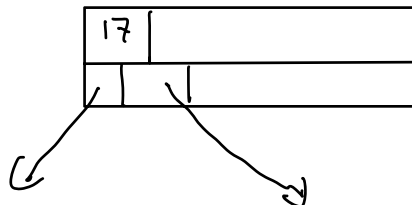
3, 25, 95

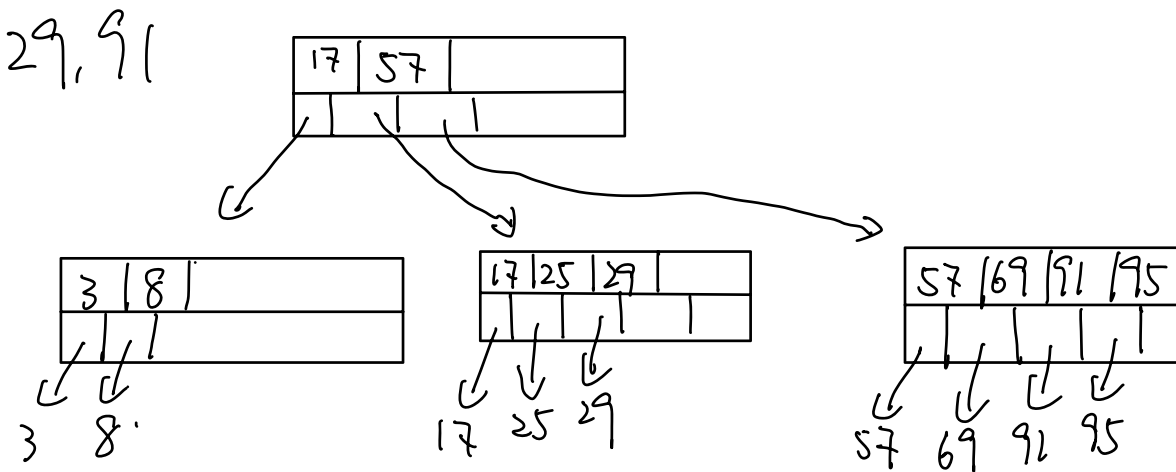
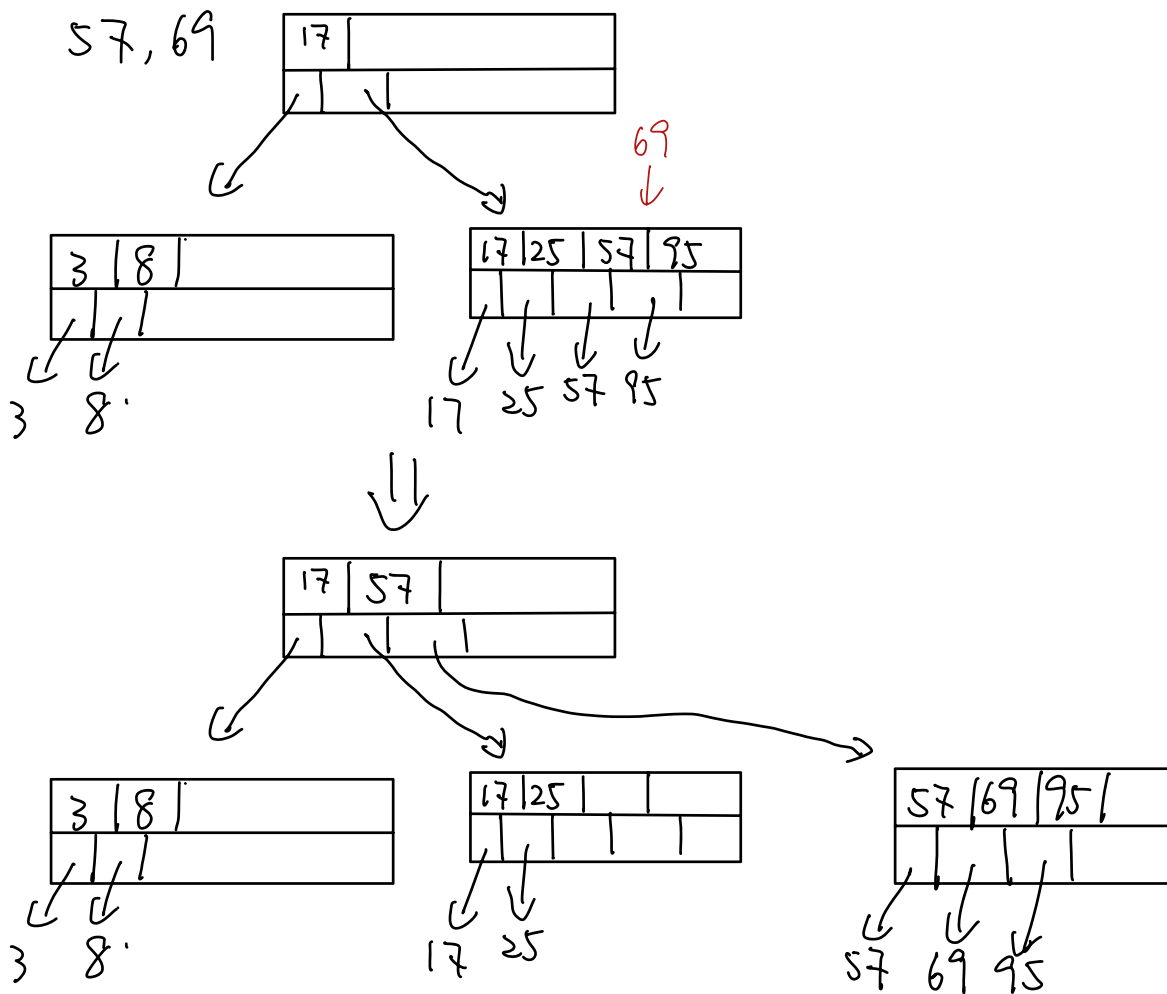


8

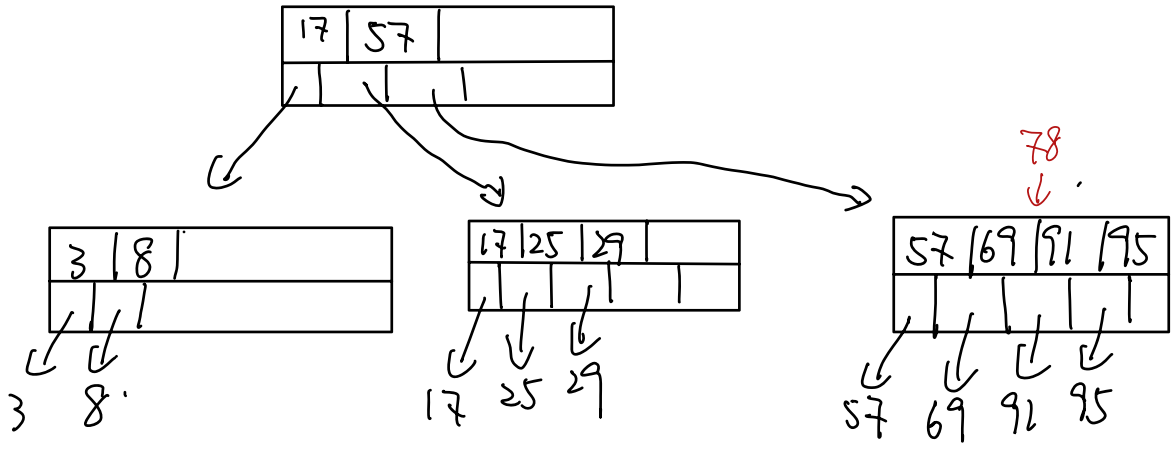


leaf node: middle key is copied to the new leaf on right

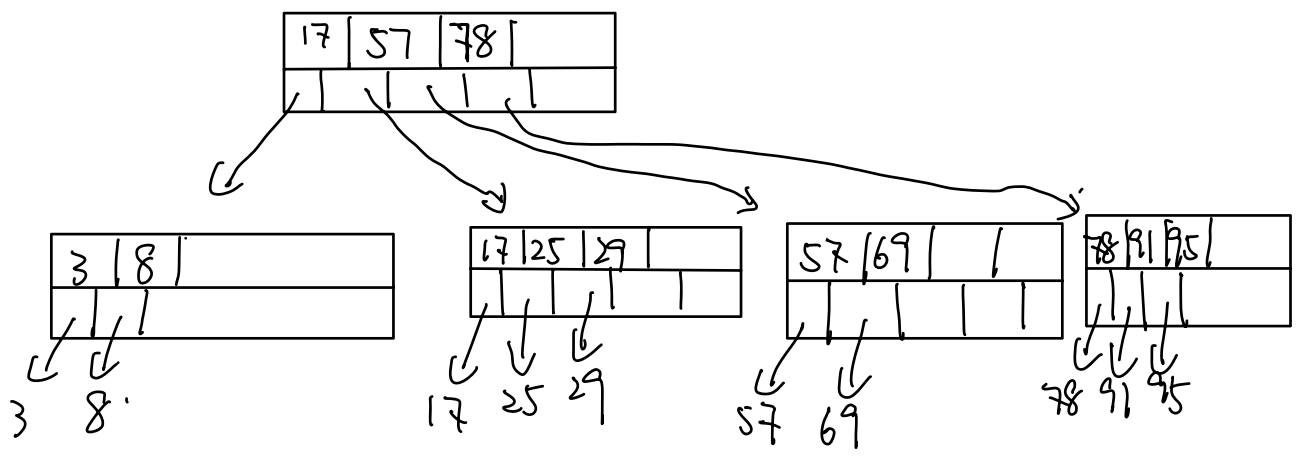




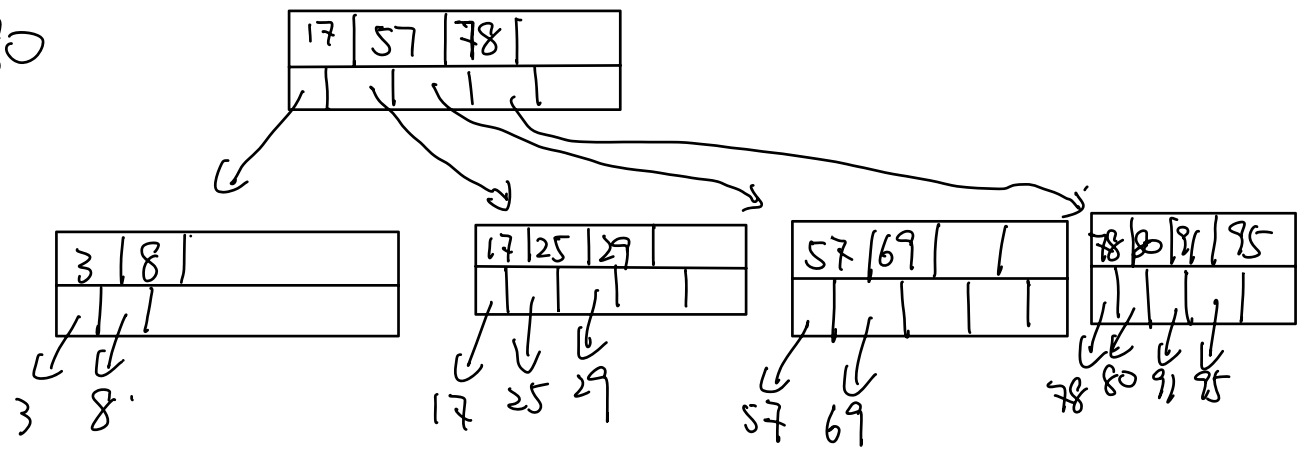
78.



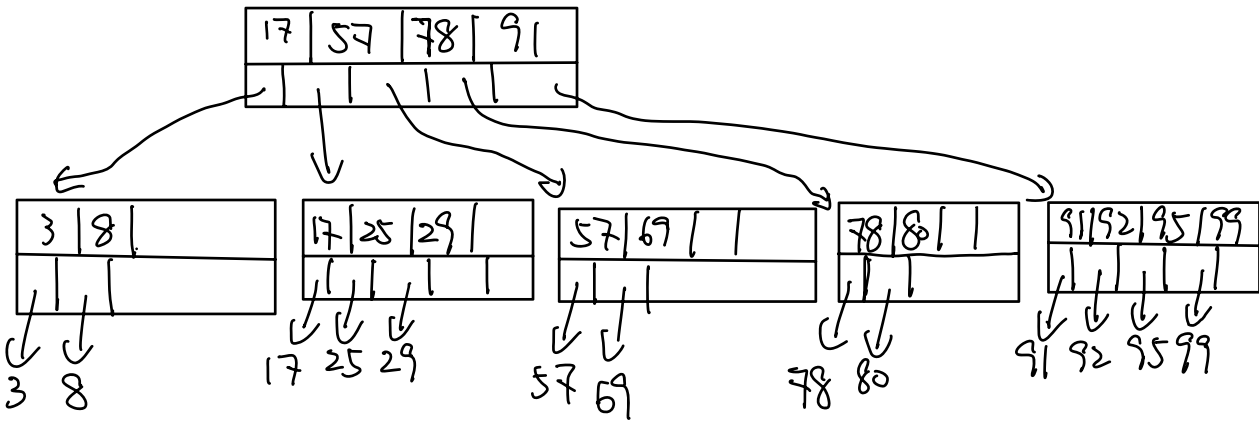
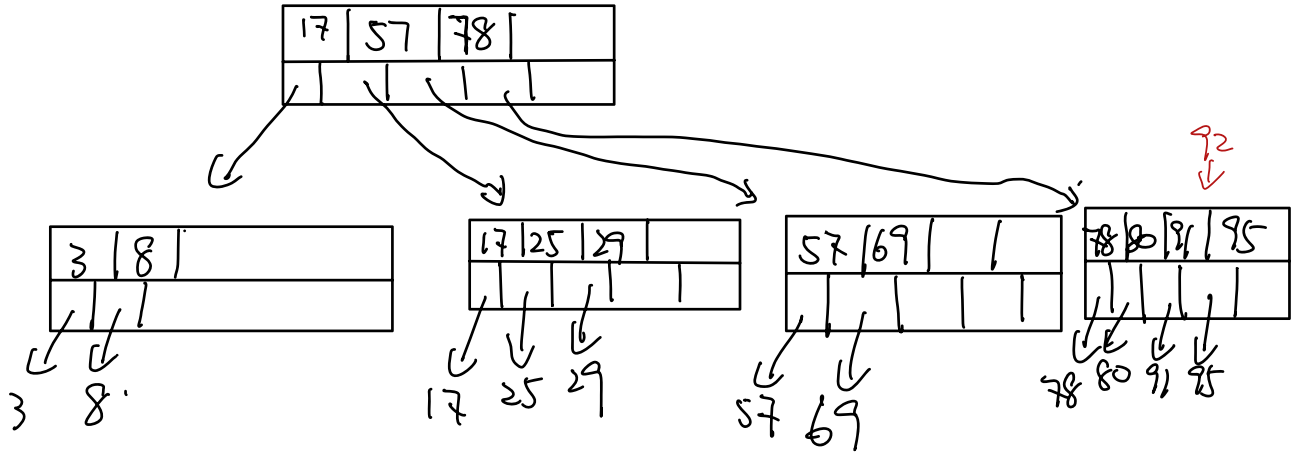
⇓



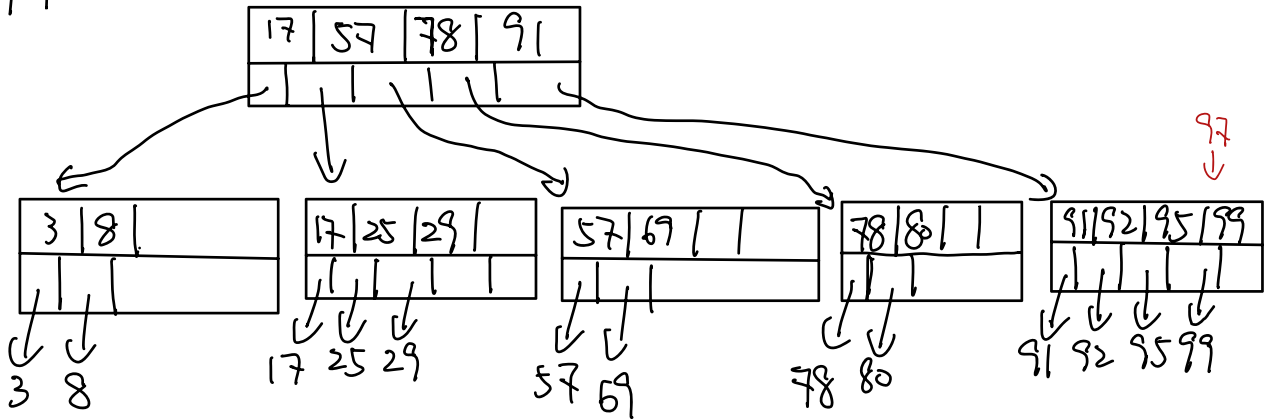
80



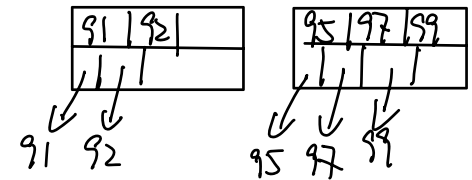
92, 99



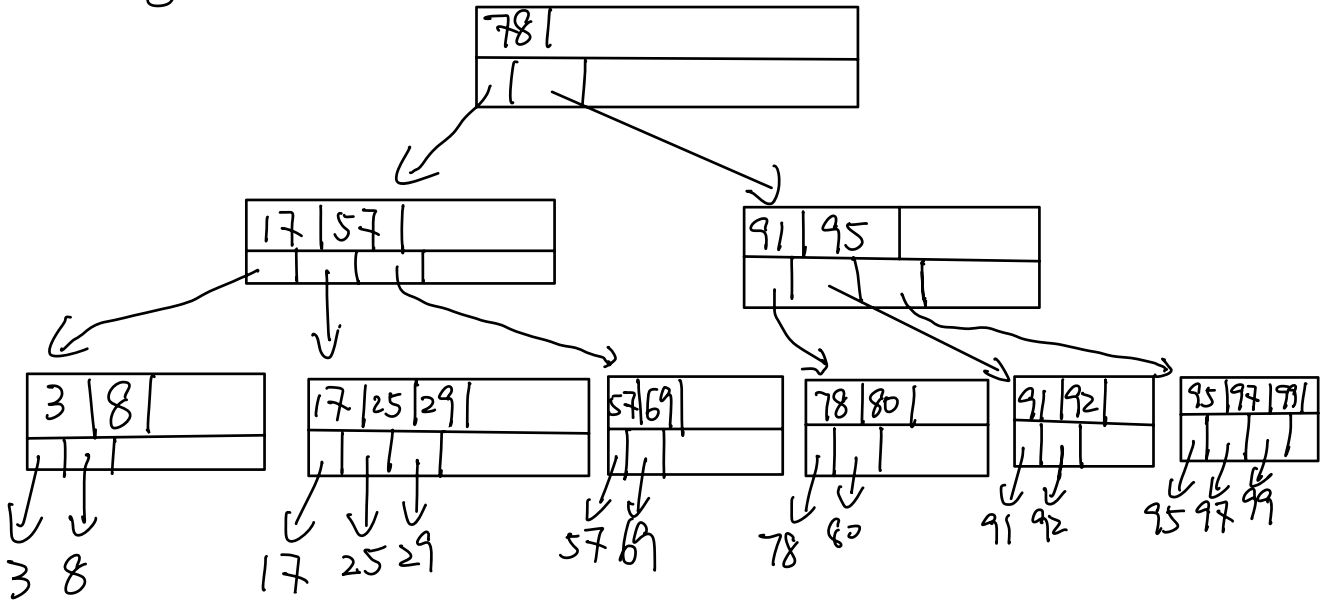
97



not enough parent pointer!
split internal node

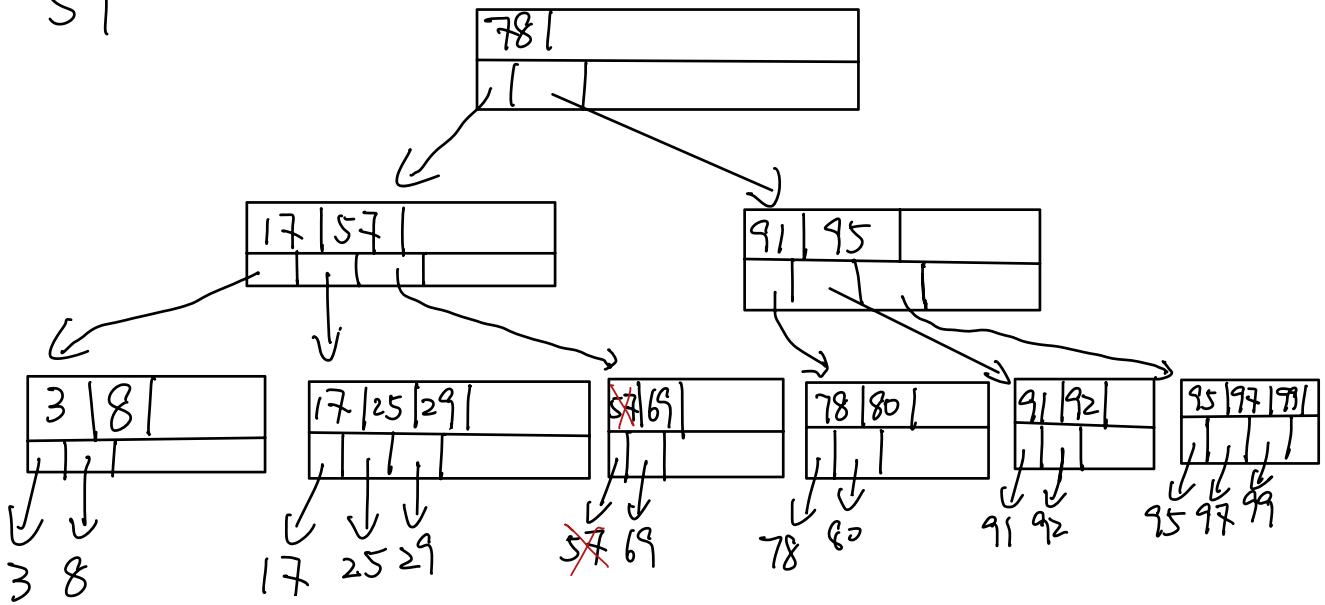


When an internal node is split, we do not need to copy the middle key to the right, only insert it in parent

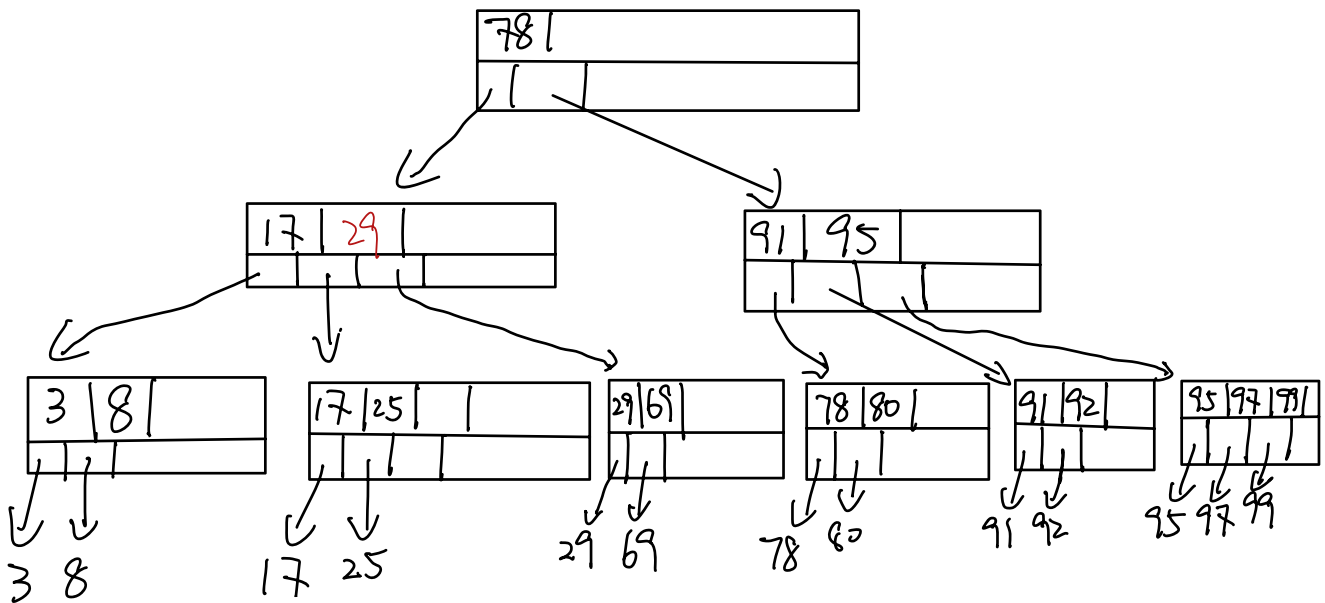


Deletion:

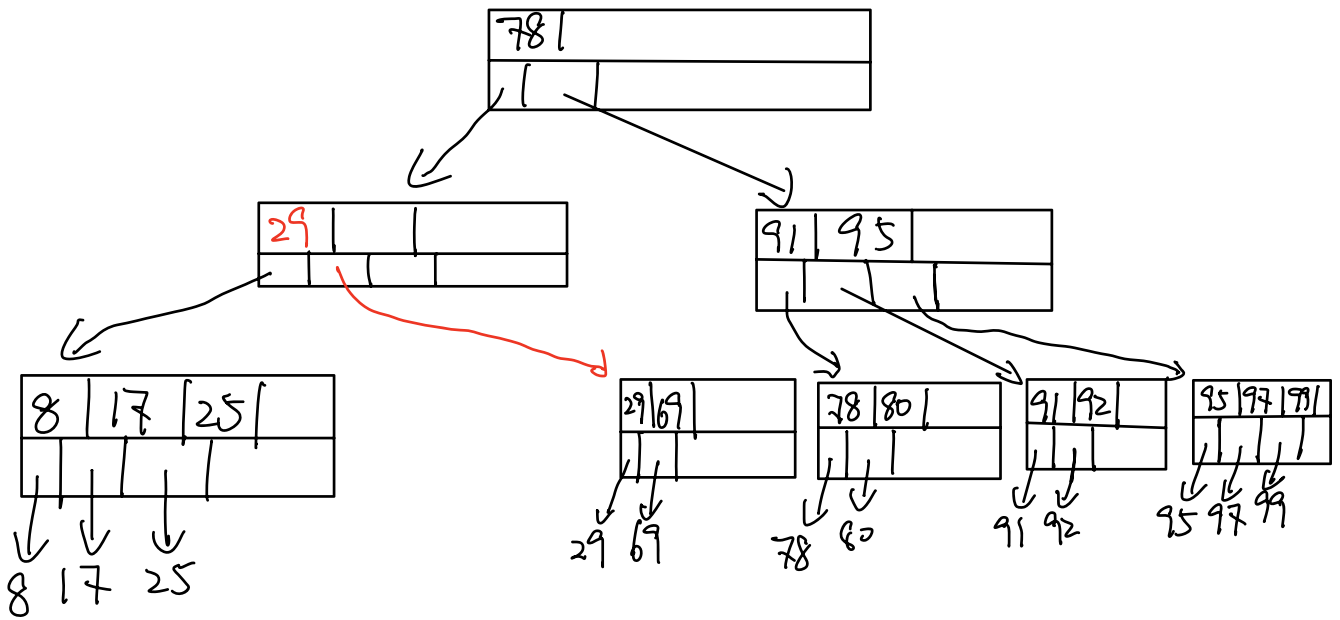
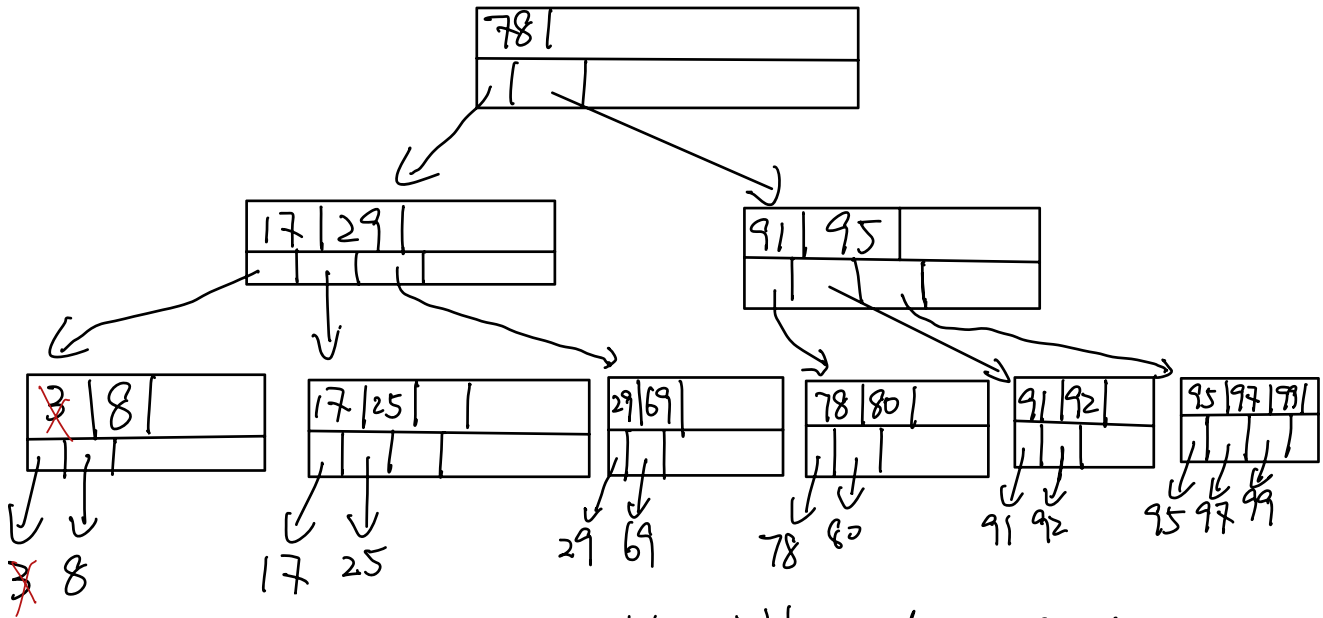
57



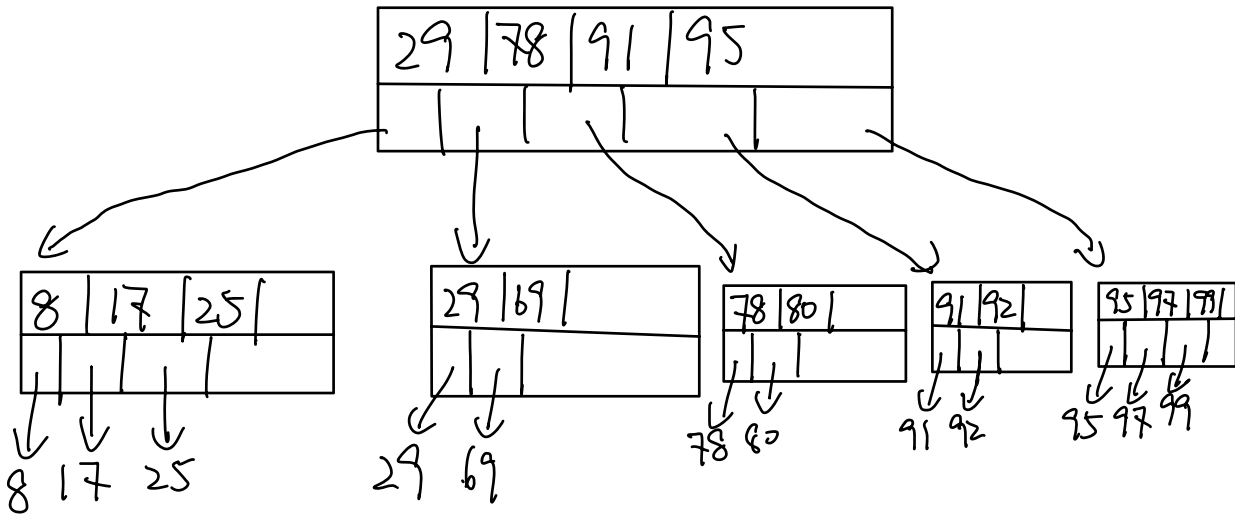
neighbor has extra
borrow from neighbor



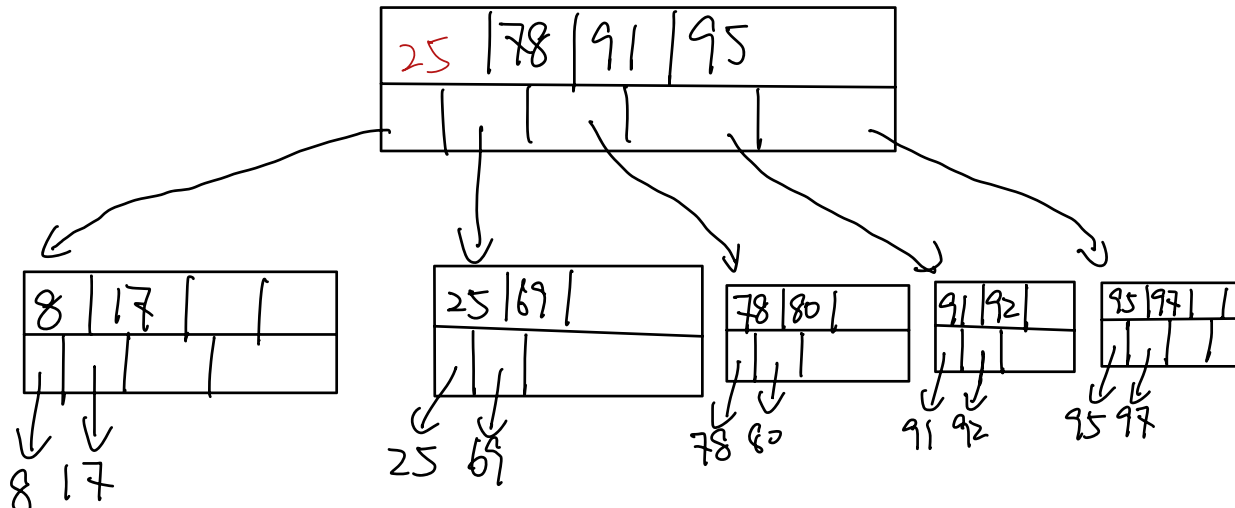
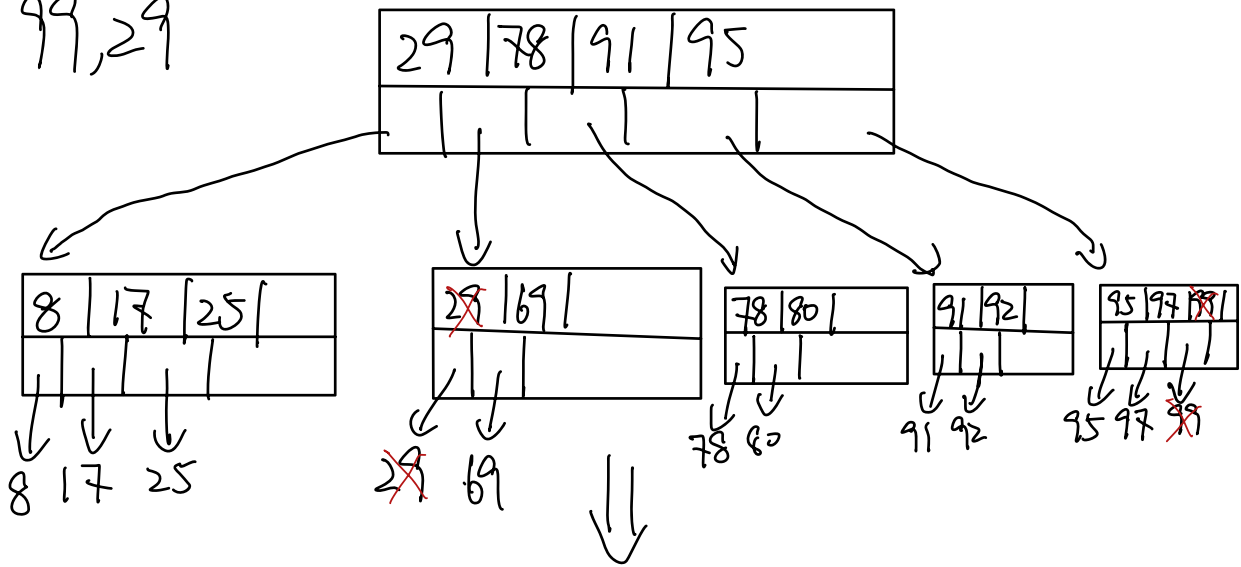
3



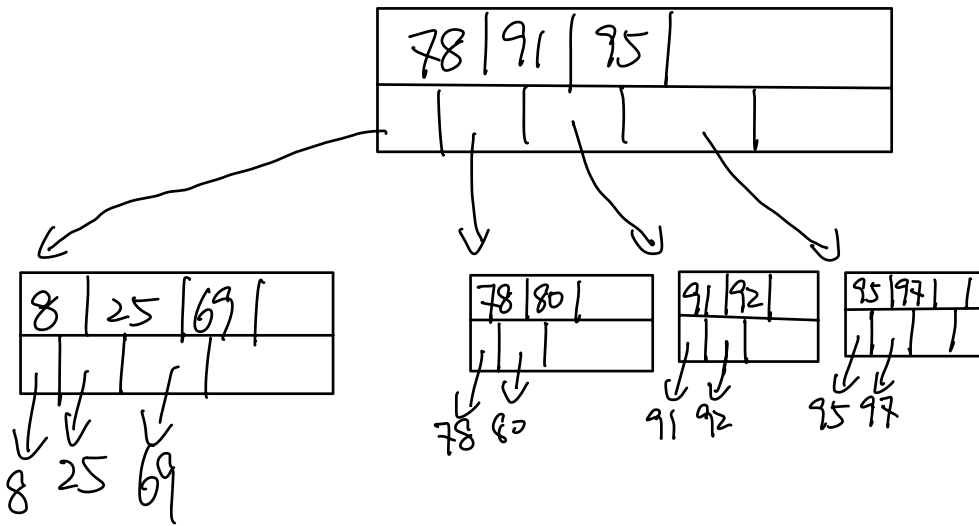
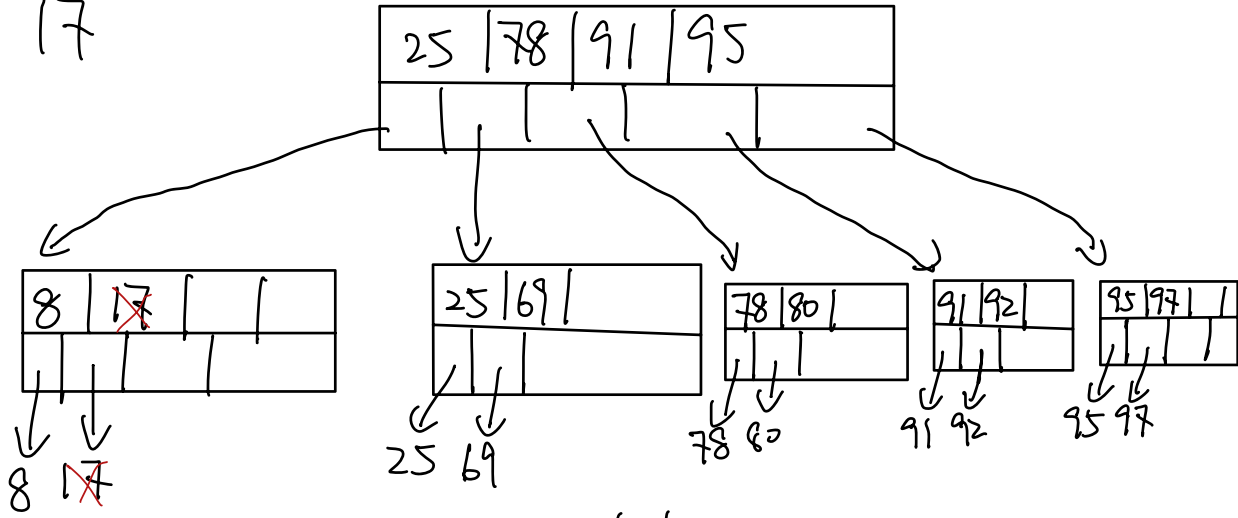
neighbor does not have extra,
merge



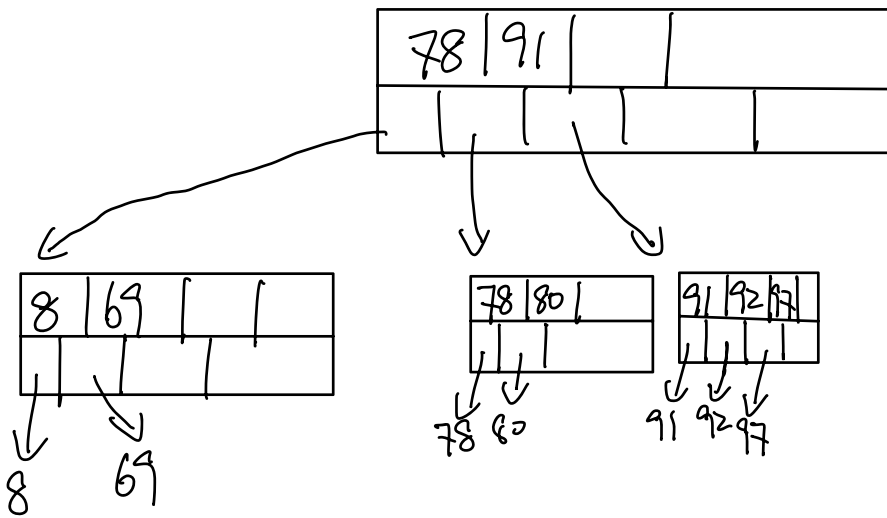
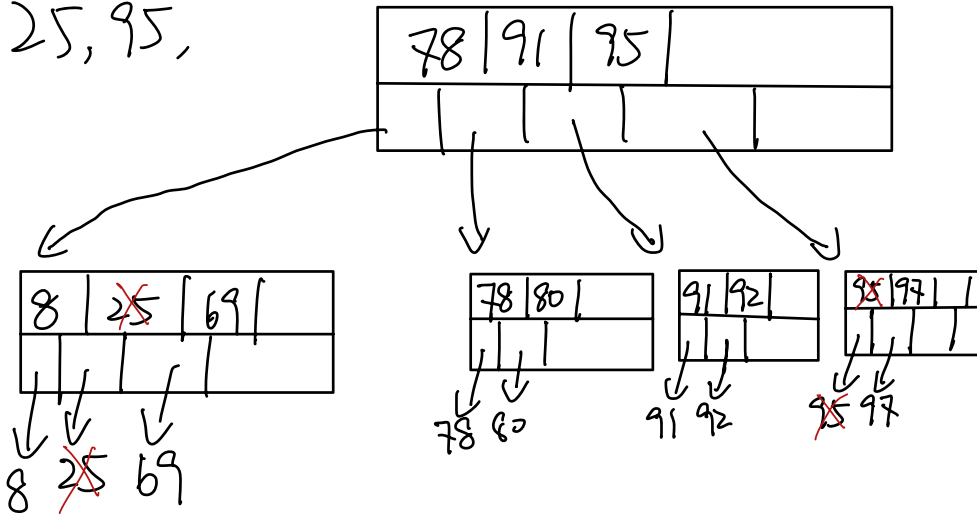
99, 29



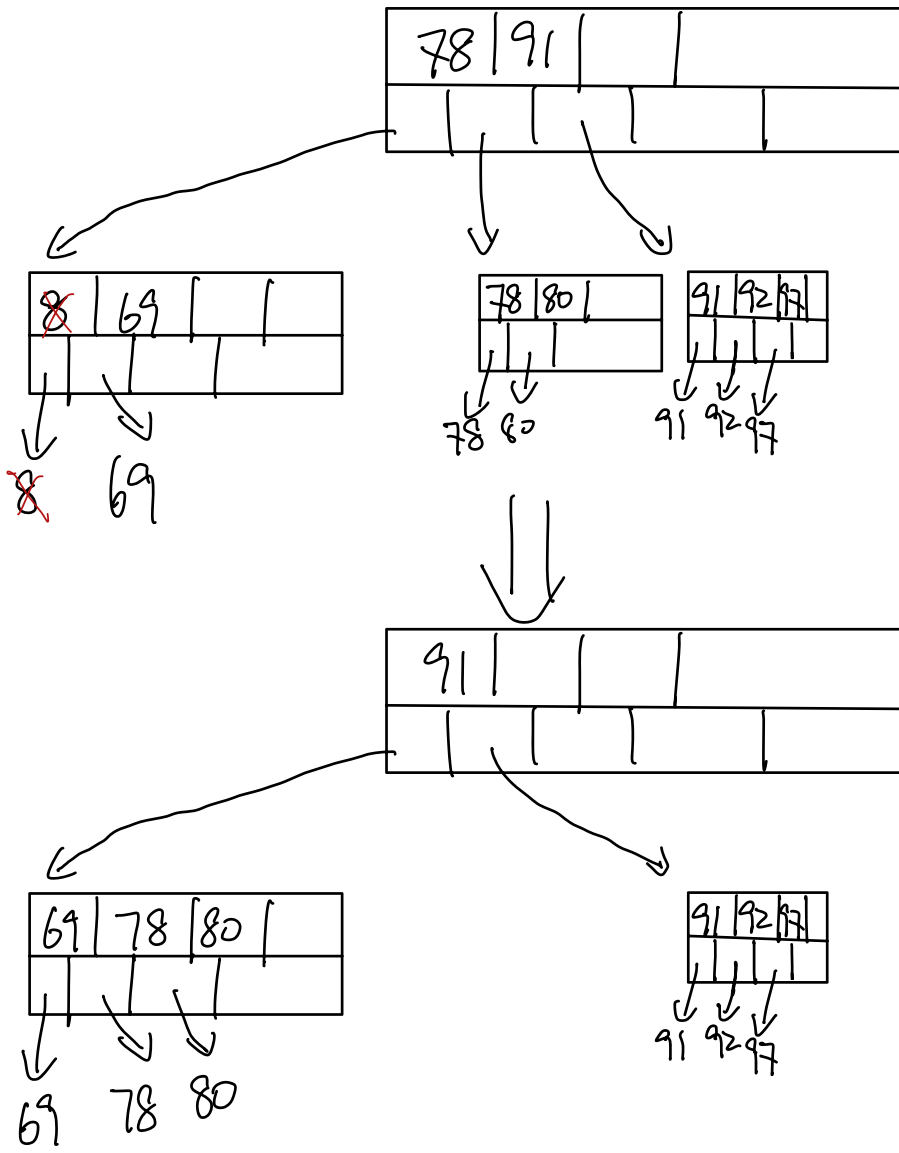
17



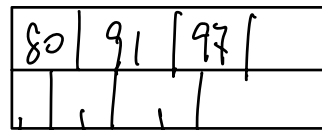
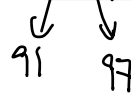
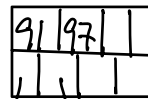
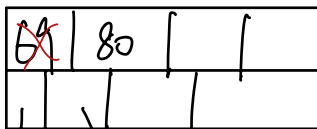
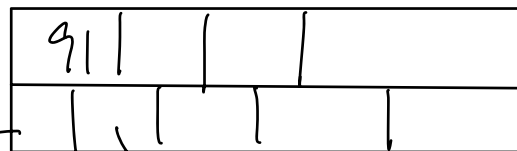
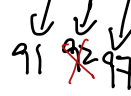
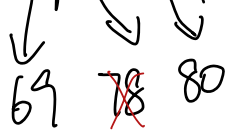
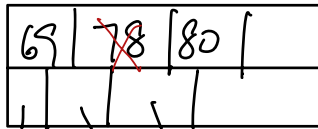
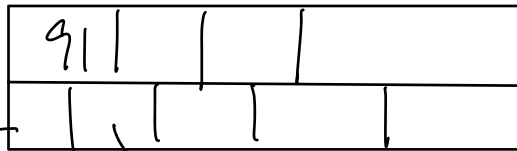
25, 95,



8



78, 92, 69



97, 91

