

OPTIMAL MERGE PATTERNS

We want to **merge some sorted files** where the number of records are {12,34,56,73,24,11,34,56,78,91,34,91,45}. What is the optimal way to merge them?

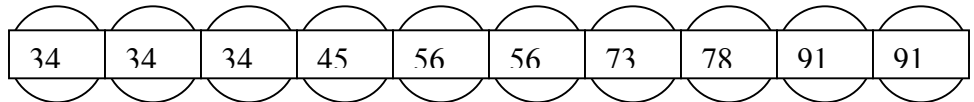
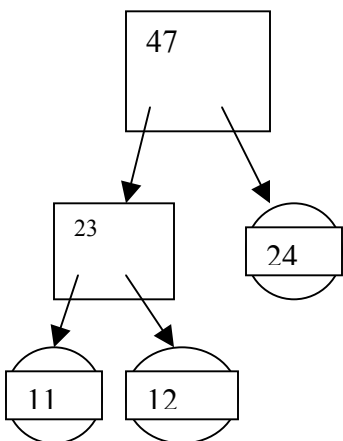
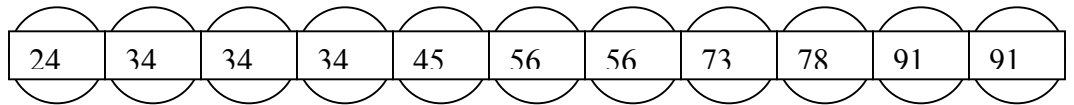
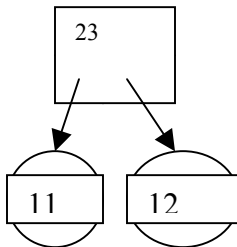
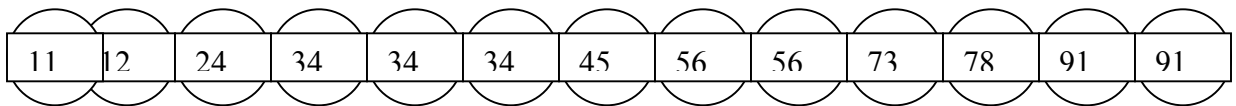
SOLUTION: At every stage we merge the files of the least length.

ALGORITHM:

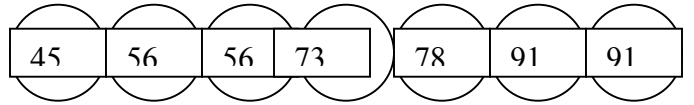
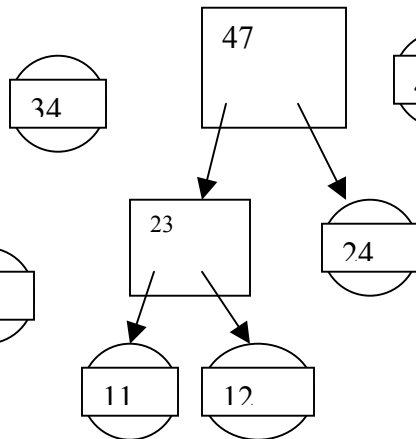
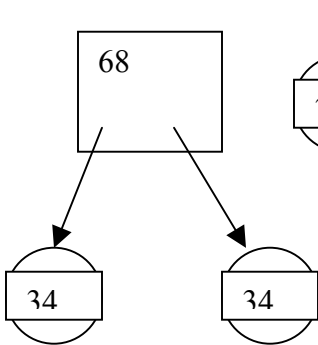
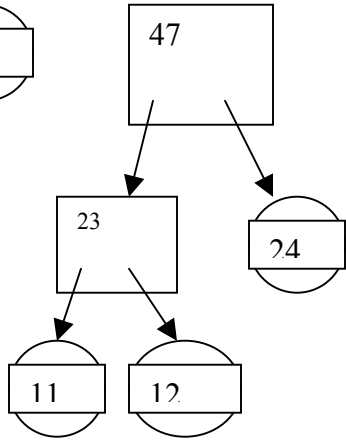
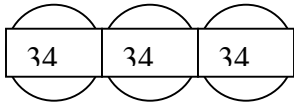
```

Create a min heap of the set of elements.
While(heap has more than one element){
    Delete a minimum element from the heap, and store it in min1;
    Delete a minimum element from the heap, and store it in min2;
    Create a node with the fields (info, left_link, right_ink);
    Let info.node = min1 + min2;
    Let left_link.node = min1;
    Let right_link_node = min2;
    Insert node with valued info into the heap;
}
    
```

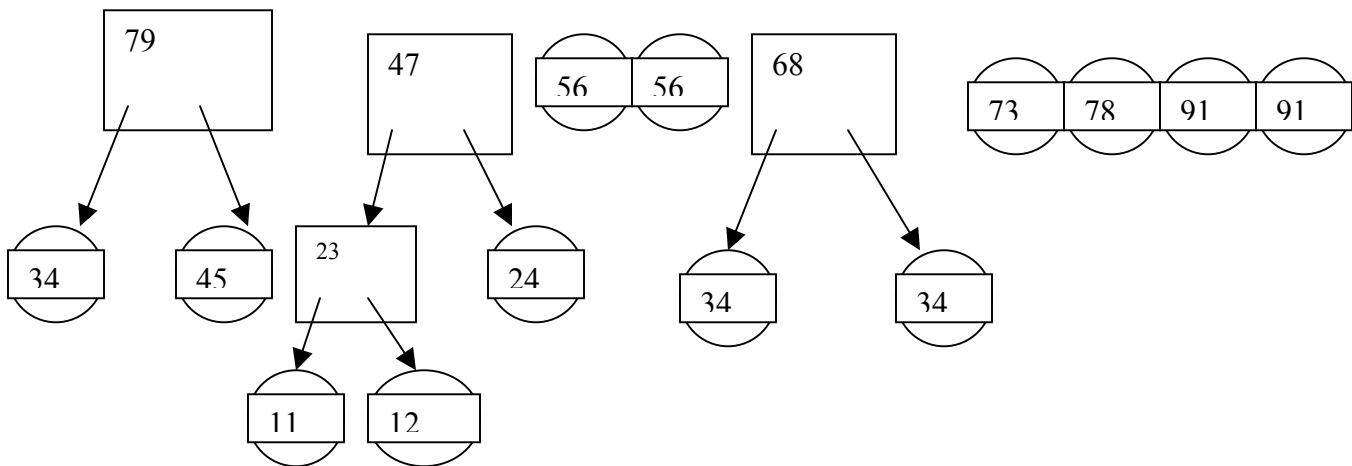
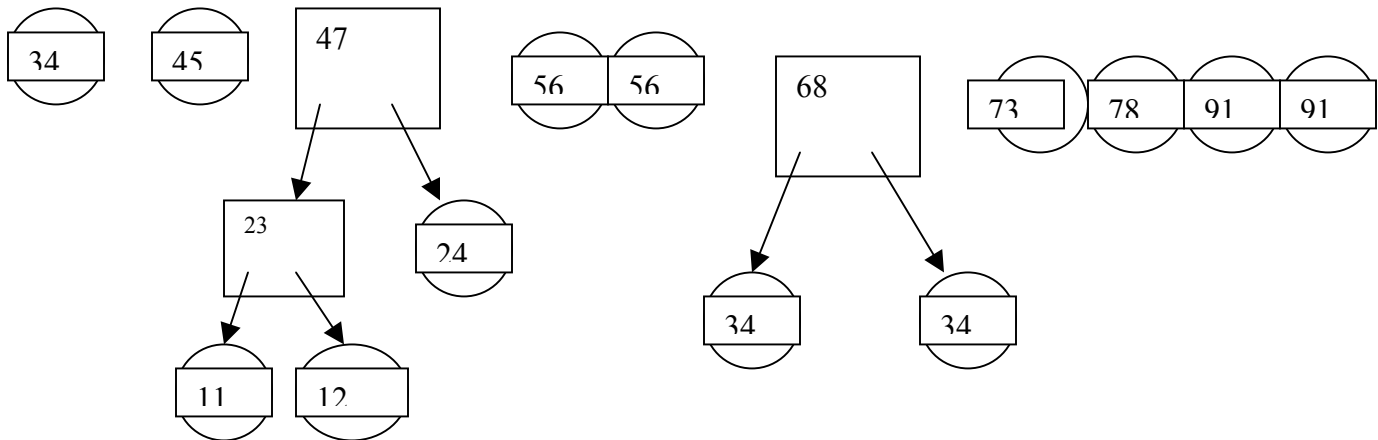
TRACING THE ALGORITHM:



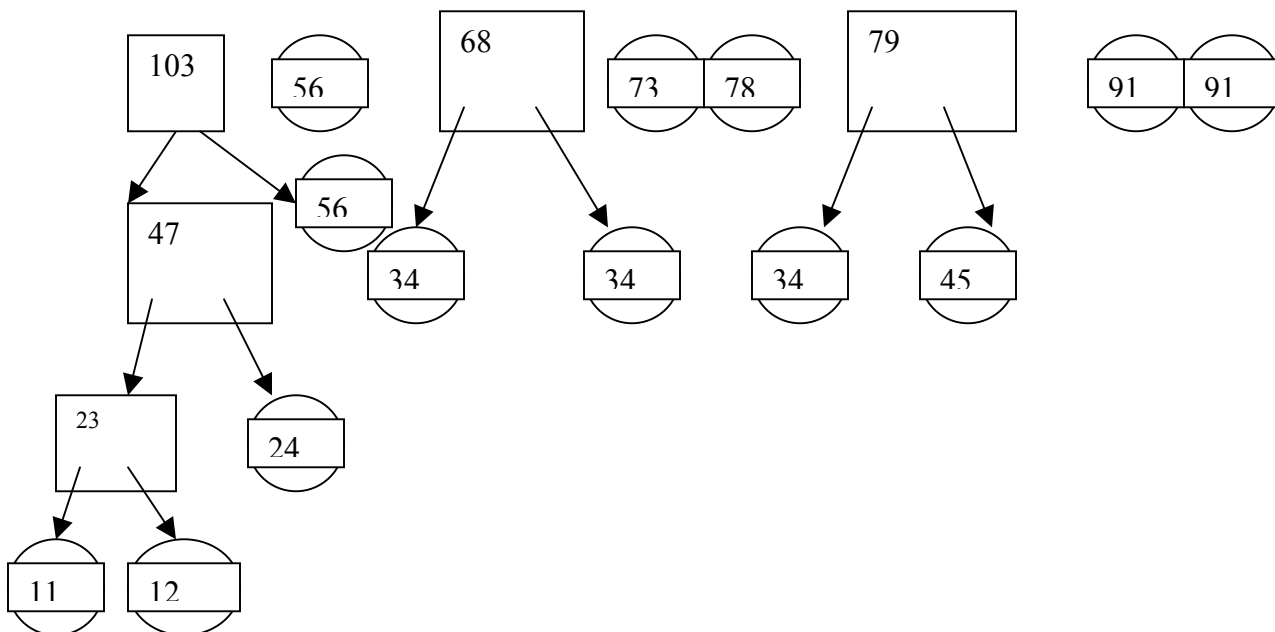
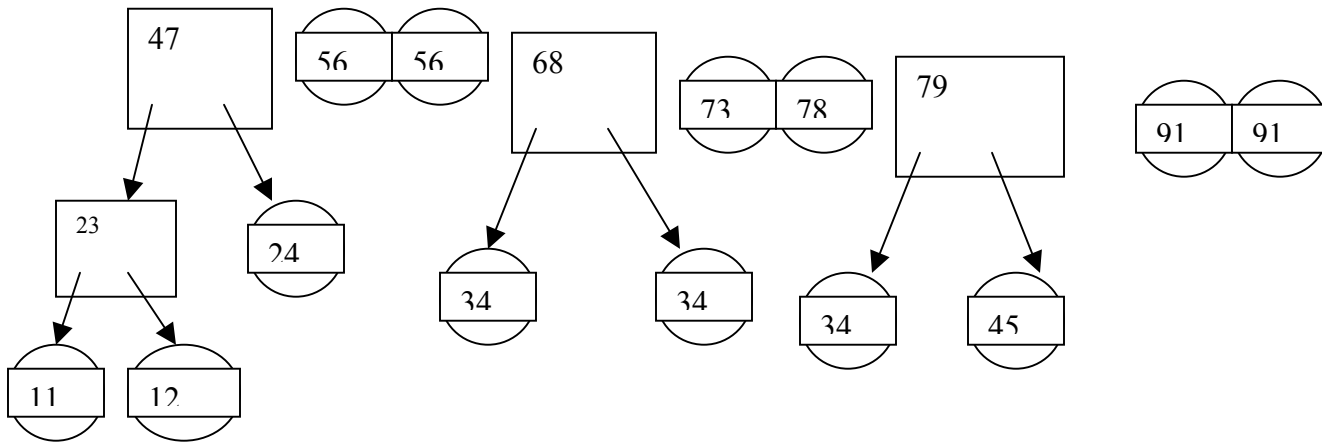
47 SHOULD BE INSERTED INTO THE MIN HEAP



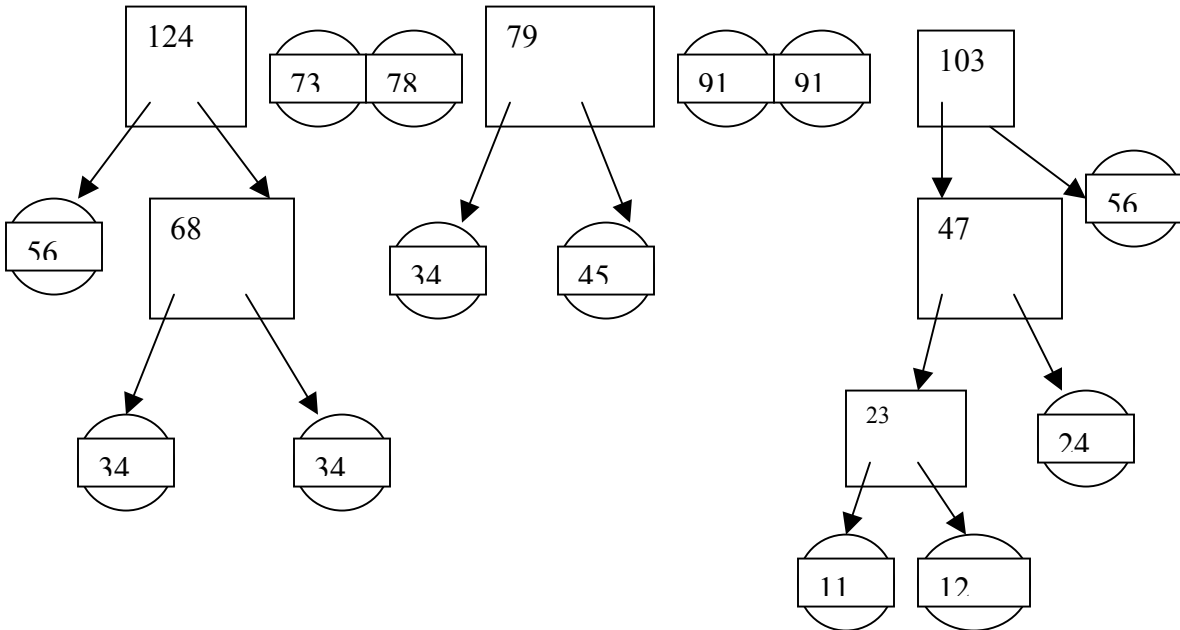
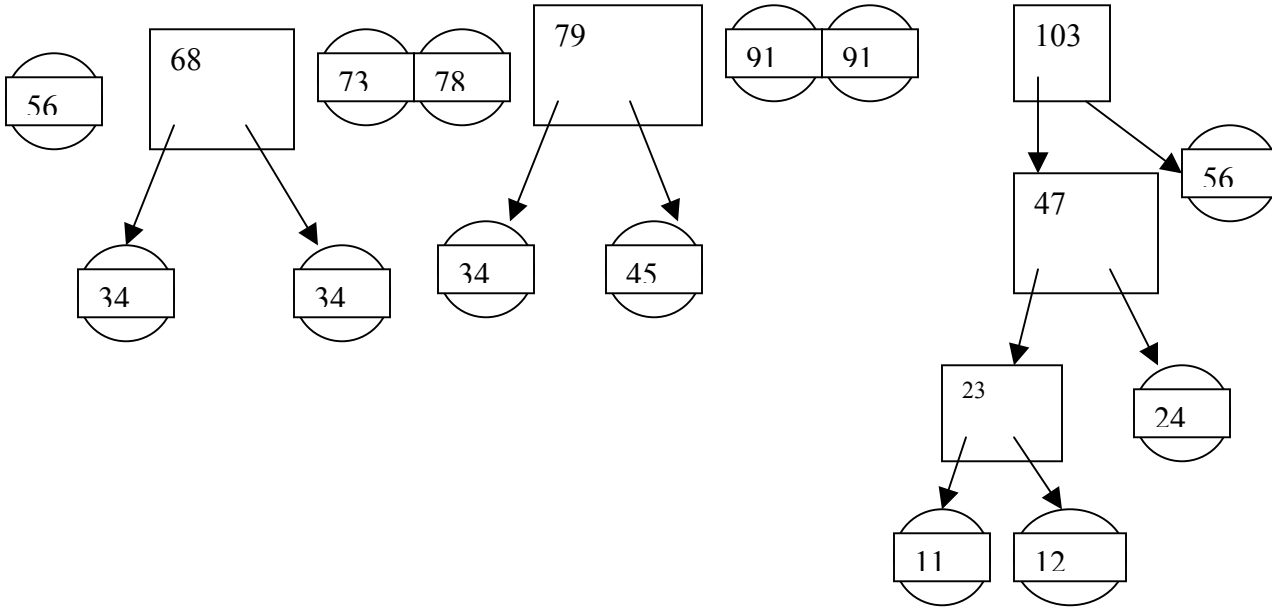
68 HAS TO BE INSERTED INTO THE HEAP



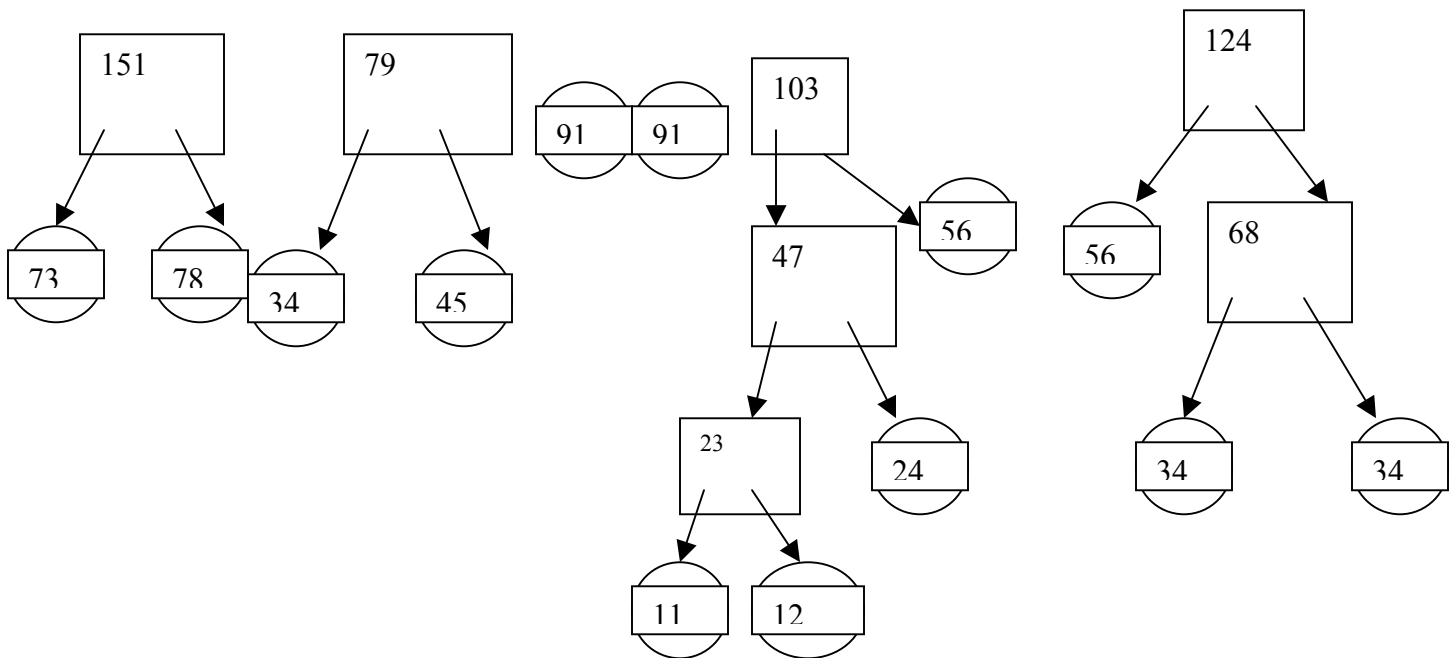
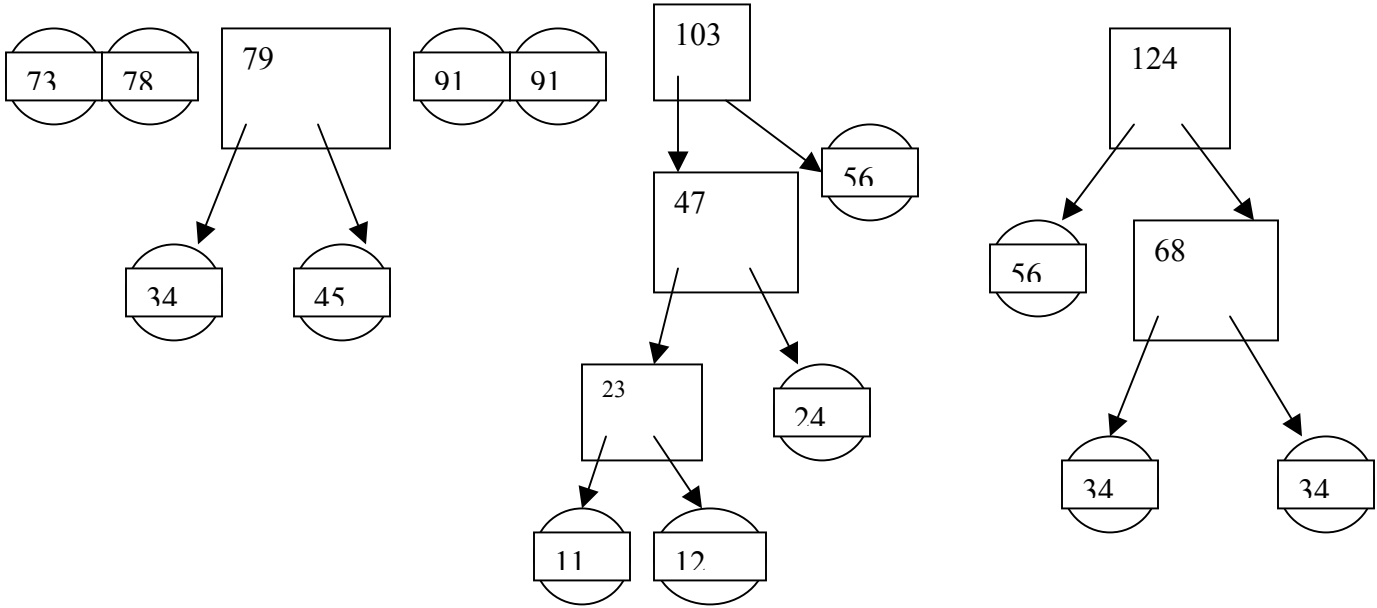
79 HAS TO BE INSERTED INTO THE HEAP IN THE PROPER POSITION



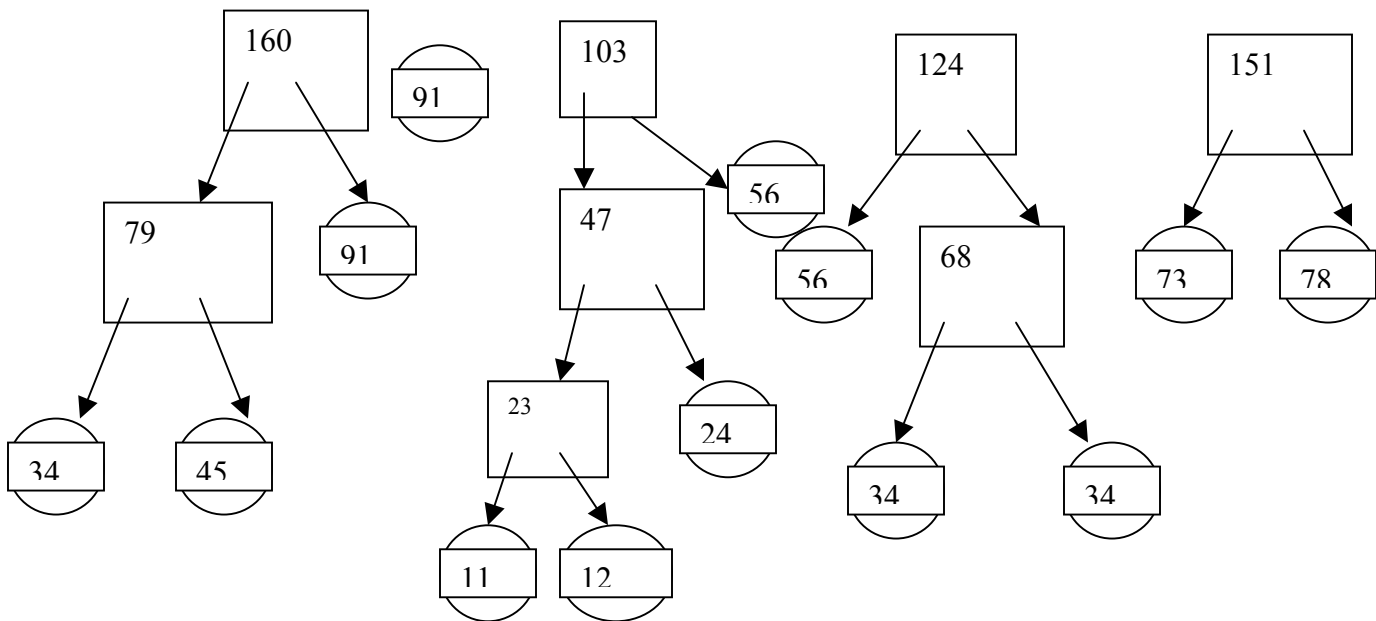
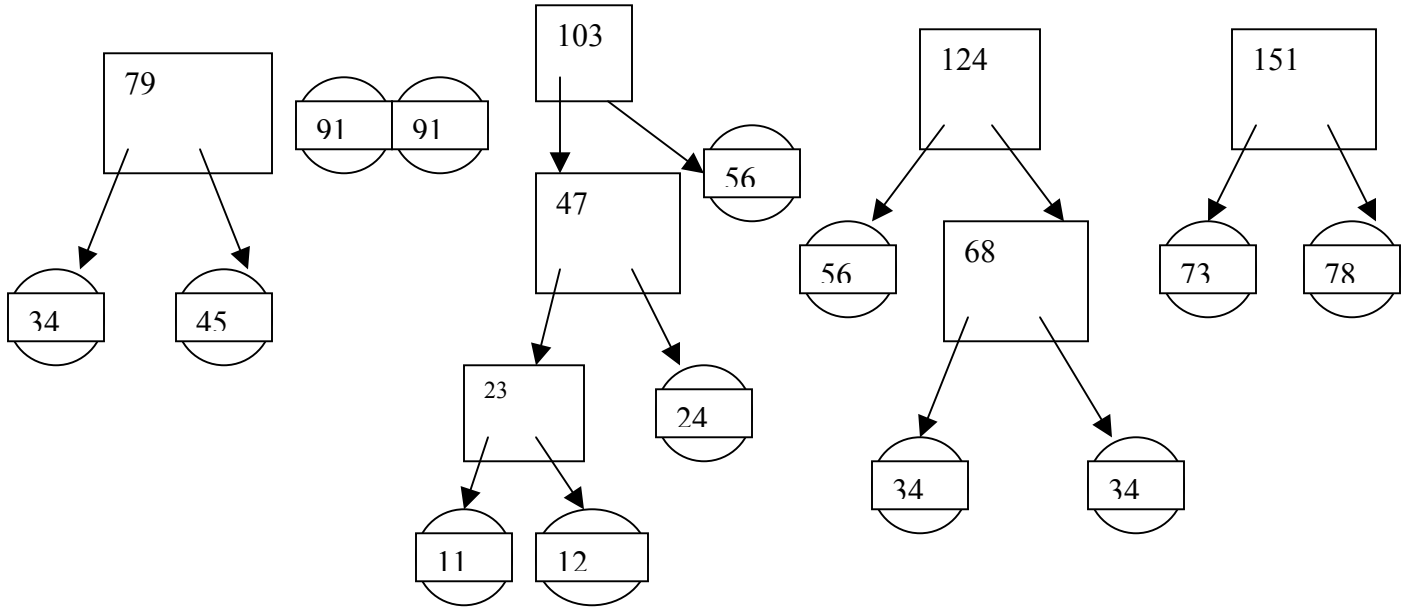
103 HAS TO BE INSERTED IN THE PROPER PLACE IN THE MIN HEAP



124 HAS TO BE INSERTED INTO THE RIGHT POSITION IN THE HEAP



151 HAS TO BE INSERTED IN THE RIGHT POSITION IN THE MIN HEAP



160 HAS TO BE INSERTED INTO THE PROPER POSITION IN THE HEAP

