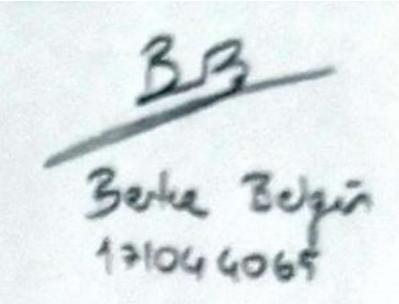
Berke Belgin 171044065

Firstly in this implementation we can hold head and tail in one variable easily and with this way especially malinhead data structures while it is taking on to add into queue, it takes of one we hold it is a circular structure since +1 index of tail is head and into queue, when we since +1 index of tail is head and into a circular structure.



private booken is leaf (Node (E) rode) [ if g node, let == null 88 node - right == null) return true; else reduce folse: public boolen is Bevanced () ( if (is Balanced (root) ==-4) return false else return true; public int is Bodonced (Node < E) node) { if (node == null) return 0; int left = is Bolonced (noole.left); < int right = 15 Bovanced (node right); if (left == -1) 11 right == -1) return -1; ance ebe if (left conght) return 1+ left + right; if! is leaf (node. right)) {! return -# ; else return 1+ left + right; Oin)

(41 SC

O-(U)

Scanned by TapScanner