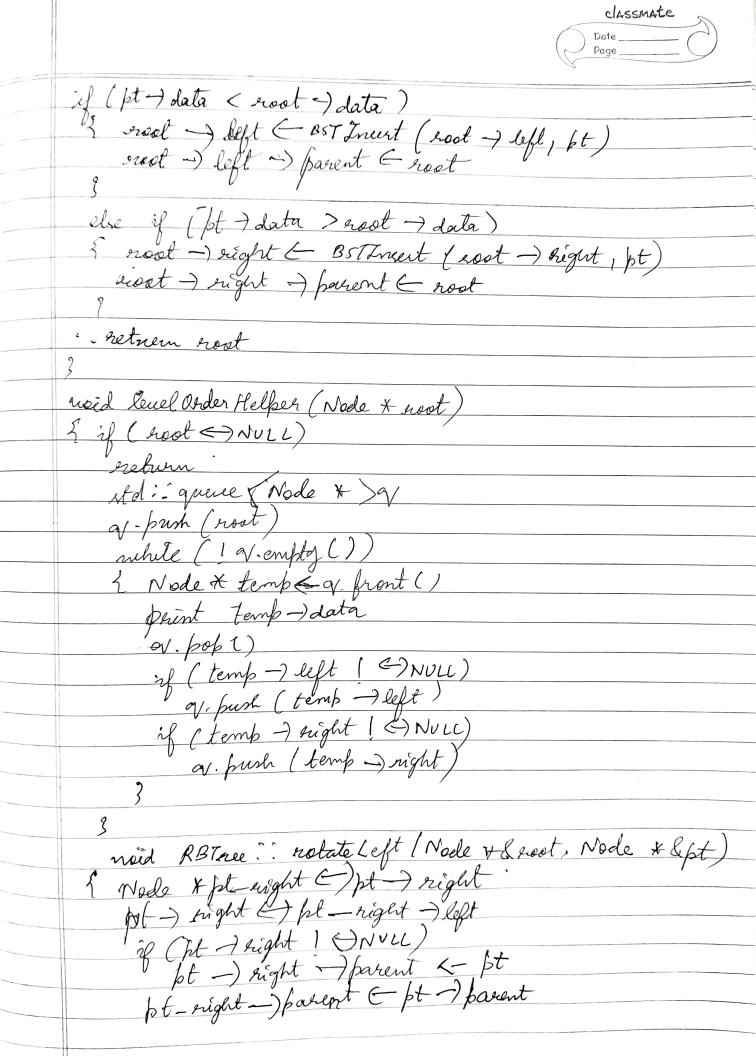
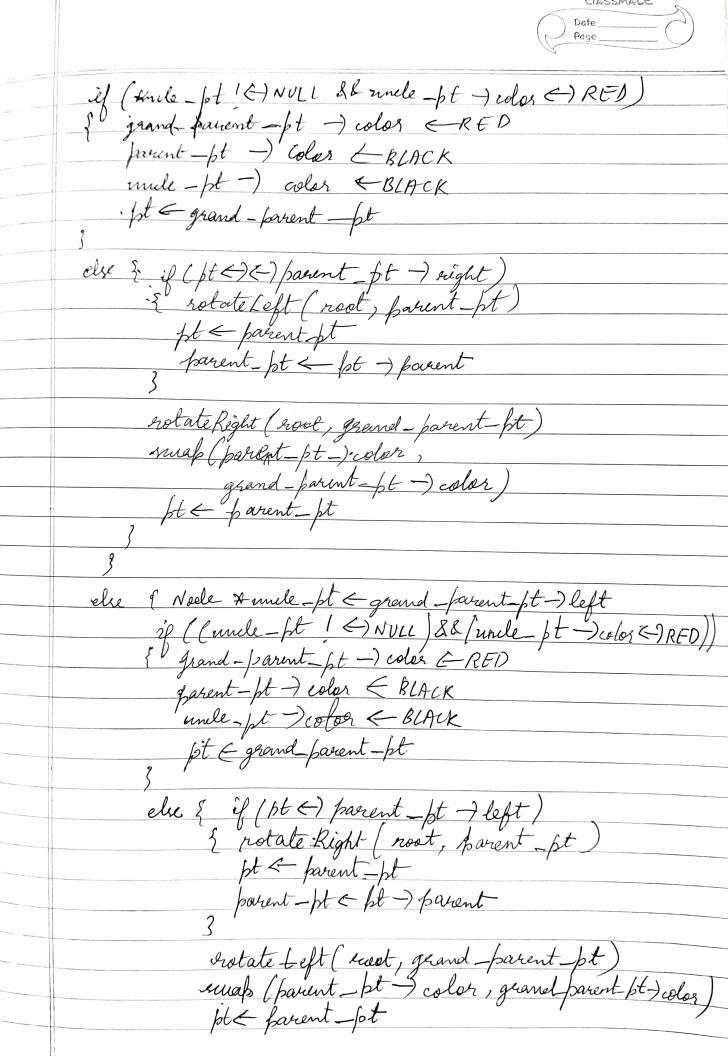
PRANAV JAGADEESH IBMISCSO71 5B BATCH 4 PROGRAM 6 REO BLACK TREES WRITE-UP enum Color & RED, BLACKS. eternt Node E int data bool wher Node & left, & sight, & farent Node (int data) & this -) data & data left & right & parent & NULL this -) eslor (- RED dan R& Tree & private: Node + nost protected noid notate Left [Node \* & , Node \*&) - rived kobate Right ( Node \* & , Node \* & noted for Violation (Node + &, Node + & public: RBTree () 3 root < NULL } noid insert (const int &n) roid inorder () noid level Order ( roid morder Helper (Nocle \* root) { if ( noot = ) NULL ) inorder Helper (hoot > left) print hoot -data marder Helper ( root -) right ) Nøde + BSTInust (Nøde + stast, Nøde \*pt) 2 if ( noot E) NUCL)



if ( ft -> parent <> NULL) good & fit - sight else if ( fit a pt faringt -) left ) pt -) parent -) left <- pt\_sight else fit - parent - sight < fit right pt\_right -) left - pt pt -) parent - pt\_right noid RBTree: notate Right [ Node \*& host, Node \* 8pt) I Noole \*fit\_left = fit ) right pt -) left = pt - keft -) night if (pt) left ( C) NULL) pt -) left -) parent ← pt pt toft -) parent ( pt -) parent if (pt -) parent ( ) NULL) nost =) pt-left du if (pt >> pt -prept -) deft) bt ) parent ) left < pt\_ left else pt-) parent -) right = pt-left ft-left - right the pt ) parent = pt \_ left ried KBTnee: fix Violation (Node & levot, Node +8pt Node & farent-pt E) NULL Node + grand-parent pt E) NULL while ((pt 1 c) Froot) ld (pt -) color 1 c) BLACK) &d (pt-) parent -) color E) RED)) parent-pt & pt -> parent grand-parent-ft & ft - ) farent - parent if ( powent - pt ( ) grand - parent - pt ) left Node & unile ft & grand-parent fft & right

Classmate



grost -) color EBLACK noid RBTree: invert (const int . & data) { Node & pt ( new Node (data) suset C-BSF Tingert ( root, ft) fix Violation (nost, pt) noid RBTree: inorder () & inorder Helper (rost)} nsid RBThee: : lendorder() { lenel Order Helper ( root) } int main () 3 RBThee tree int n <0, m <0 buint modes to be inserted n = value print Enter nodes for (int ico, icn, i++) { me value tree. trust (m) print Inorder Tueneeral free inorder () point Level Order Traverial tree level Order () return 0

Classmate