void RB\_DELETE\_FIXUP(BST \*\*T ,BST \*x)

{BST \*w;

while ((x != \*T) && (x->color == cBLACK))

{

if (x== (x->parent)->left)

{

w = (x->parent)->right;

if(w->color == cRED)

{

w->color = cBLACK; // Case 1

(x->parent)->color = cRED; // Case 1

LEFT\_ROTATE(T, x->parent); // Case 1

w = (x->parent)->right; // Case 1

}

if (((w->left)->color== cBLACK) && ((w->right)->color== cBLACK))

{

w->color= cRED ; // Case 2

x =x->parent; // Case 2

}

else

{

if ((w->right)->color== cBLACK)

{

(w->left)->color =cBLACK ; // Case 3

w->color=cRED; // Case 3

RIGHT\_ROTATE(T, w) ; // Case 3

w= ( x->parent)->right; // Case 3

}

w->color = ( x->parent)->color; // Case 4

(x->parent)->color=cBLACK ; // Case 4

(w->right)->color =cBLACK ; // Case 4

LEFT\_ROTATE(T, x->parent); // Case 4

x =\*T ; // Case 4

}

}

else //(same as then clause with "right" and "left" exchanged)

{

w = (x->parent)->left;

if(w->color == cRED)

{

w->color = cBLACK; // Case 1

(x->parent)->color = cRED; // Case 1

RIGHT\_ROTATE(T, x->parent); // Case 1

w = (x->parent)->left; // Case 1

}

if (((w->right)->color== cBLACK) && ((w->left)->color== cBLACK))

{

w->color= cRED ; // Case 2

x =x->parent; // Case 2

}

else

{

if ((w->left)->color== cBLACK)

{

(w->right)->color =cBLACK ; // Case 3

w->color=cRED; // Case 3

LEFT\_ROTATE(T, w) ; // Case 3

w= ( x->parent)->left; // Case 3

}

w->color = ( x->parent)->color; // Case 4

(x->parent)->color=cBLACK ; // Case 4

(w->left)->color =cBLACK ; // Case 4

RIGHT\_ROTATE(T, x->parent); // Case 4

x =\*T ; // Case 4

}

}

}// end while

x->color= cBLACK;

} // RB\_DELETE\_FIXUP