void Morris(struct Node\* root)

{

struct Node \*curr, \*prev;

if (root == NULL)

return;

curr = root;

while (curr != NULL) {

if (curr->left\_node == NULL) {

cout << curr->data << endl;

curr = curr->right\_node;

}

else {

/\* Find the previous (prev) of curr \*/

prev = curr->left\_node;

while (prev->right\_node != NULL && prev->right\_node != curr)

prev = prev->right\_node;

/\* Make curr as the right child of its

previous \*/

if (prev->right\_node == NULL) {

prev->right\_node = curr;

curr = curr->left\_node;

}

/\* fix the right child of previous \*/

else {

prev->right\_node = NULL;

cout << curr->data << endl;

curr = curr->right\_node;

}

}

}

}