#include <stdio.h>

#include <stdlib.h>

struct node

{

int data;

struct node \*middle;

struct node \*left;

struct node \*right;

};

struct node \*newNode(int data)

{

struct node \*temp = (struct node \*)malloc(sizeof(struct node));

temp->data = data;

temp->left = temp->right = temp->middle = NULL;

return temp;

}

void createPrevNode(struct node \*root, struct node \*head)

{

struct node \*temp = newNode(root->data);

if(!head->left)

{

head->left = newNode(0);

head->left->right = head;

head->middle = temp;

}

}

void createNextNode(struct node \*root, struct node \*head)

{

struct node \*temp = newNode(root->data);;

if(!head->right)

{

head->right = newNode(0);

head->right->left = head;

head->middle = temp;

}

}

void createLeafNode(struct node \*root, struct node \*head)

{

if(!root->left && !root->right && !head->middle)

{

struct node \*temp = newNode(root->data);

temp->middle = head->middle;

head->middle = temp;

}

}

void TopViewDLL(struct node \*root, struct node \*head)

{

if(root->left)

{

createPrevNode(root, head);

TopViewDLL(root->left, head->left);

}

if(root->right)

{

createNextNode(root, head);

TopViewDLL(root->right, head->right);

}

createLeafNode(root, head);

}

void TopViewOfTree(struct node \*root)

{

struct node \*head = newNode(0);

TopViewDLL(root, head);

while(head->left)

head = head->left;

struct node \*temp;

while(head)

{

printf("%d\t", head->middle->data);

head = head->right;

}

}

int main()

{

struct node \*root = newNode(10);

root->left = newNode(20);

root->right = newNode(30);

root->left->left = newNode(40);

root->left->right = newNode(50);

root->right->left = newNode(60);

root->right->right = newNode(70);

TopViewOfTree(root);

return 0;

}