Exp (9): write a c program to implement the tree traversals (inorder, preorder, postorder)

Aim: To write a c program to implement the tree traversals (inorder, preorder, postorder)

Algorithm:

*. Start.
*. Create a binary tree.
*. Implement recursive functions for inorder, preorder, and postorder traversals.
*. Display the traversal.
*. Stop.

program:

```
#include <stdio.h>
#include <stdlib.h>
struct node {
        int data;
        struct node * left, * right;
};
struct node* newNode(int x) {
  struct node * n = malloc(sizeof(struct node));
  n->data = x;
  n->left = n->right = NULL;
  return n;
}
void inorder(struct node * r){
  if(!r) return;
  inorder(r->left);
  printf("%d", r->data);
  inorder(r->right);
}
void preorder(struct node * r) {
    if(!r) return;
    printf("%d", r->data);
    preorder(r->left);
    preorder(r->right);
}
```

```
void postorder (struct node* r) {
    if (!r) return;
    postorder (r->left);
    postorder (r->right);
    printf ("%d", r->data);
}

int main () {
    struct node* root = newNode (1);
    root->left = newNode (2);
    root->right = newNode (3);
    root->left->left = newNode (4);
    root->left->right = newNode (5);
    printf ("Inorder: "); inorder (root);
    printf ("\nPreorder: "); preorder (root);
    printf ("\nPostorder: "); postorder (root);

    return 0;
}
```

output:

```
Inorder : 4 2 5 1 3
Preorder : 1 2 4 5 3
Postorder : 4 5 2 3 1
```

Result: Thus, the program executed successfully