实验四

#include<stdio.h>

#include <stdlib.h>

/\*树的数据类型\*/

typedef char TElemType;

/\*定义二叉树的结构\*/

typedef struct BiTNode

{

TElemType data; //数据

struct BiTNode \*left,\*right; //左右孩子节点

}BiTNode,\*BiTree;

void createBiTree(BiTree\* T)

{

char ch;

scanf("%c",&ch);

if(ch ==' ')

{

\*T = NULL;

}

else

{

\*T = (BiTNode\*)malloc(sizeof(BiTNode));

if(!\*T) exit(-1);

(\*T)->data = ch;

createBiTree(&(\*T)->left);

createBiTree(&(\*T)->right);

}

}

void preOrder(BiTree T)

{

if(T==NULL) return;

printf("%c",T->data);

preOrder(T->left);

preOrder(T->right);

}

void InOrder(BiTree T)

{

if(T==NULL)return ;

InOrder(T->left);

printf("%c",T->data);

InOrder(T->right);

}

void PostOrder(BiTree T)

{

if(T==NULL)return;

PostOrder(T->left);

PostOrder(T->right);

printf("%c",T->data);

}

int main(int argc, char const \*argv[])

{

BiTree T;

createBiTree(&T);

printf("先序遍历:\n");

preOrder(T);

printf("\n中序遍历:\n");

InOrder(T);

printf("\n后序遍历:\n");

PostOrder(T);

//ABC DE G F

return 0;

}

总结：本次实验要求掌握二叉树的建立，难点在于二叉树三种遍历的代码实现，需要理解三种遍历的原则，参考网络代码自己大致理解了三种遍历如何用代码实现。