|  |  |  |  |
| --- | --- | --- | --- |
| 2022/2 『자료구조』실습 보고서 | | | |
| 제목 | 8장 실습( O ) 과제( ) | 제출일자 | 2022.  11 .    10 . |
| 학번 | 201911608 | 이름 | 김 지환 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. 프로그램 8.1 - 링크로 생성된 이진트리를 구현하고 그 이진트리를 그리기  #include <stdio.h>  #include <stdlib.h>  typedef int element;  typedef struct TreeNode {  element data;  struct TreeNode\* left, \* right;  } TreeNode;  int main(void)  {  TreeNode\* n1, \* n2, \* n3;  n1 = (TreeNode\*)malloc(sizeof(TreeNode));  n2 = (TreeNode\*)malloc(sizeof(TreeNode));  n3 = (TreeNode\*)malloc(sizeof(TreeNode));  n1->data = 10;  n1->left = n2;  n1->right = n3;  n2->data = 20;  n2->left = NULL;  n2->right = NULL;  n3->data = 10;  n3->left = NULL;  n3->right = NULL;  printf("\t%4d\n", n1->data);  printf("\t%d%4d\n", n2->data, n3->data);  return 0;  }  실행결과      - > 포화 이진트리  2. Quiz (266쪽) 01, 02 문제 풀이  01. 다음과 같은 트리를 배열 표현법으로 저장된 모습을 그려라. 각 노드는 어떤 인덱스에 저장되는가?    ->   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | 10 |  | 20 |  |  |  | 30 |  |  |  |  |  |  | 25 |  |   0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15  02. 위의 트리가 링크 표현법으로 저장된 모습을 그려라.    프로그램 8.3 - 링크로 생성된 이진트리의 순회  #include <stdio.h>  #include <stdlib.h>  #include <memory.h>  typedef int element;  typedef struct TreeNode {  element key;  struct TreeNode\* left, \* right;  } TreeNode;  TreeNode n1 = { 1, NULL, NULL };  TreeNode n2 = { 4, &n1, NULL };  TreeNode n3 = { 16, NULL, NULL };  TreeNode n4 = { 25, NULL, NULL };  TreeNode n5 = { 20, &n3, &n4 };  TreeNode n6 = { 15, &n2, &n5 };  TreeNode\* root = &n6;  void inorder(TreeNode\* root) {  if (root) {  inorder(root->left);// 왼쪽서브트리 순회  printf("[%d] ", root->key); // 노드 방문  inorder(root->right);// 오른쪽서브트리 순회  }  }  void preorder(TreeNode\* root) {  if (root != NULL) {  printf("[%d] ", root->key);  preorder(root->left);  preorder(root->right);  }  }  void postorder(TreeNode\* root) {  if (root != NULL) {  postorder(root->left);  postorder(root->right);  printf("[%d] ", root->key);  }  }  int main(void) {  printf("중위 순회 = ");  inorder(root);  printf("\n");  printf("전위 순회 = ");  preorder(root);  printf("\n");  printf("후위 순회 = ");  postorder(root);  printf("\n");  return 0;  }  실행결과      중위 순회 - 왼쪽 노드에 있는 값들부터 출력    전위 순회 - 현재 노드부터 출력한 이후 왼쪽 노드, 오른쪽 노드 탐색    후위 순회 - 왼쪽 노드, 오른쪽 노드 탐색 후 출력, 즉 모든 자식 출력 후 부모 출력  4. Quiz(276쪽) 01 문제 풀이    01. 위의 트리를 전위, 중위, 후위 순회활 경우, 노드의 방문 순서는? |