**자료구조 과제 5**

(이진트리 생성 및 중위순회)

logo.gif

|  |  |
| --- | --- |
| 과목명 | 자료구조 |
| 담당교수 | 김계영 |
| 학과 | 소프트웨어학부 |
| 학년 | 3 |
| 학번 | 20152994 |
| 성명 | 이진영 |
| 제출일 | 2017.6.12 |



**과제5 – 이진트리 생성 및 중위순회**

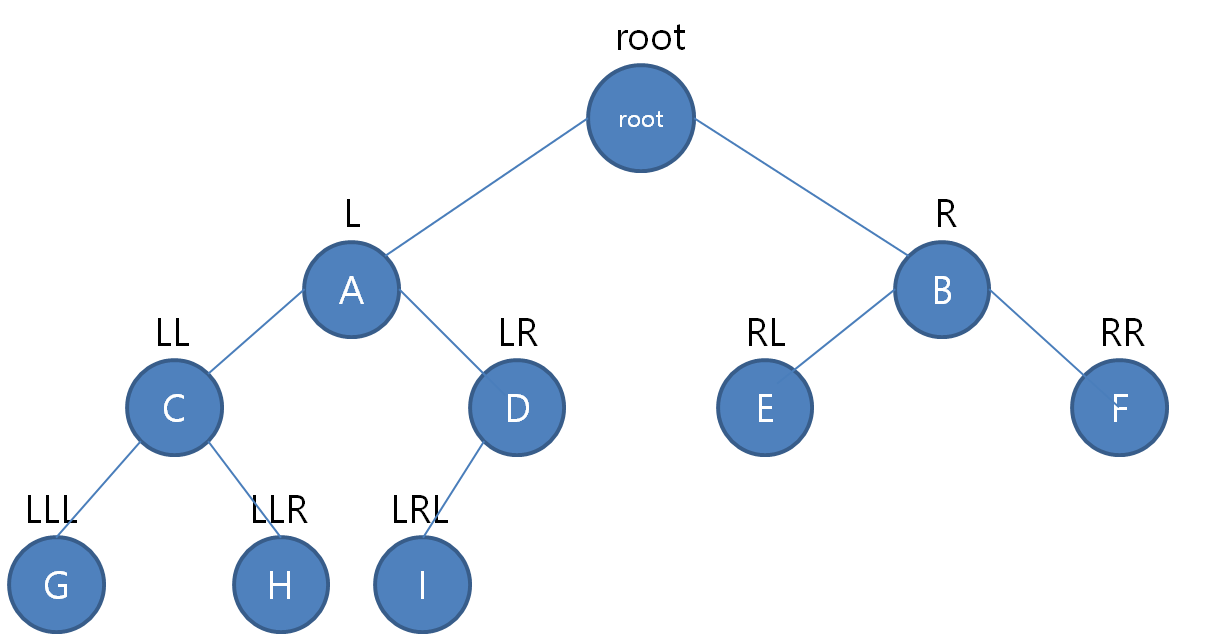
20152994 이진영

**1. 원시 코드**

|  |
| --- |
| #include <stdio.h>  #include <malloc.h>  #include <string.h>  char\* treeInfo[] = { "L.A","R.B",  "LL.C","LR.D","RL.E","RR.F",  "LLL.G","LLR.H","LRL.I" }; // 이진트리 정보  typedef struct node\* treePointer;  typedef struct node {  char\* data;  treePointer leftChild;  treePointer rightChild;  };  void inorder(treePointer ptr);  void main() {  treePointer node;  treePointer root; // 루트 노드 정의  root = (treePointer)malloc(sizeof(treePointer));  root->data = "root";  root->leftChild = NULL;  root->rightChild = NULL;  int i = 0;  while (treeInfo[i]) { // 읽어온 트리 정보에 알맞게 이진트리 생성  node = root;  for (int j = 0; j < strlen(treeInfo[i]); j++) {  if (treeInfo[i][j] == 'L') {  if (node->leftChild == NULL) {  node->leftChild = (treePointer)malloc(sizeof(treePointer));  node->leftChild->leftChild = NULL;  node->leftChild->rightChild = NULL;  if (treeInfo[i][j + 1] == '.') {  node->leftChild->data = &treeInfo[i][j + 2];  j = strlen(treeInfo[i]) - 1;  }  }  node = node->leftChild;  }  if (treeInfo[i][j] == 'R') {  if (node->rightChild == NULL) {  node->rightChild = (treePointer)malloc(sizeof(treePointer));  node->rightChild->leftChild = NULL;  node->rightChild->rightChild = NULL;  if (treeInfo[i][j + 1] == '.') {  node->rightChild->data = &treeInfo[i][j + 2];  j = strlen(treeInfo[i]) - 1;  }  }  node = node->rightChild;  }  }  i++;  }  printf("\n※중위순회 결과\n");  inorder(root); // 루트노드부터 시작하여 중위순회  printf("\n\n");  }  void inorder(treePointer ptr) // 중위순회 재귀 함수  {  if (ptr) {  inorder(ptr->leftChild);  printf("%s ", ptr->data);  inorder(ptr->rightChild);  }  } |

**2. 실행 결과**

**① 트리 그림**

****

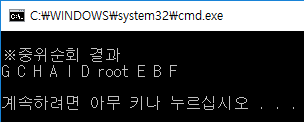
**② 입력 자료**

char\* treeData[] = { "L.A","R.B",

"LL.C","LR.D","RL.E","RR.F",

"LLL.G","LLR.H","LRL.I" };

**③ 실행 결과**

****