

자료구조론

2014년 시행 5급(기술) 공채 제2차시험

응시번호 :

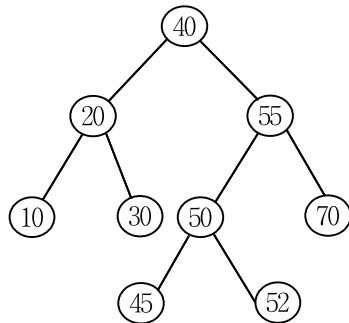
성명 :

제 1 문. 이진 탐색트리를 저장하기 위한 노드 구조는 다음과 같다. data 필드는 노드의 값이고 left와 right 필드는 각각 왼쪽 자식과 오른쪽 자식의 노드포인터이다. 노드 값은 모두 양수이고 동일한 노드 값을 갖는 복수의 노드는 존재하지 않는 것으로 가정하며, 편의상 노드 값 x를 갖는 노드를 '노드 x'로 부르기로 한다. 다음 질문에 답하시오. (총 25점)

struct tnode

struct tnode *left	int data	struct tnode *right
--------------------	----------	---------------------

(이진 탐색트리 예)



- 이진 탐색 트리 상의 임의의 노드 x에 대한 조상들을 자신을 포함하여 순서대로 출력하는 함수 void Ancestor(tnode, int)를 C 언어와 유사한 언어로 작성하시오. 예를 들어, 상기 그림의 이진 탐색 트리의 경우, Ancestor(root, 50)의 수행 결과는 40, 55, 50을 출력한다. (단, root는 루트노드포인터이고 노드 x는 트리에 반드시 존재한다고 가정한다) (7점)

(정의) **가장가까운조상**: 이진 탐색 트리 상의 노드 v, w의 가장가까운 조상 노드는 v와 w를 자손으로 갖는 가장 레벨이 낮은 노드이다. (단, 노드 x의 조상은 자기 자신을 포함한 모든 조상을 의미한다)

- 이진 탐색 트리 상의 두 노드 v와 w의 공통 조상 중 가장가까운조상의 노드 값을 구하는 재귀 함수 int NearAncestor(tnode, int, int)를 작성하시오. 예를 들어, 상기 그림의 이진 탐색 트리의 경우, NearAncestor(root, 45, 70) = 55, NearAncestor(root, 50, 55) = 55이다. 답이 없을 경우 -1을 반환한다. (단, 노드 v, w는 트리에 반드시 존재한다고 가정한다) (10점)
- 트리가 이진 탐색 트리가 아닌 일반 트리라고 가정하자. 일반 트리에서 가장가까운조상의 노드 값을 구하는 재귀 함수 int NearAncestor2(tnode, int, int)를 작성하시오. 답이 없을 경우 -1을 반환한다. (8점)

제 2 문. 아래와 같이 C 언어 형식으로 주어진 노드의 구조를 이용하여 이중 연결 리스트(doubly-linked list)를 표현하고자 한다. NodePtr 타입의 listData가 리스트의 첫 번째 노드를 가리키는 포인터라고 할 때, 다음 물음에 답하시오. (총 15점)

```
typedef struct node_tag {
    int info;
    struct node_tag *next;
    struct node_tag *prev;
} Node;
typedef Node* NodePtr;
```

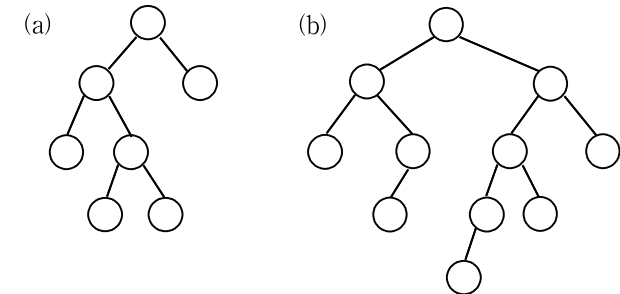
- 1) 리스트에서 노드의 info 값이 양 옆 노드에 있는 info 값의 합보다 큰 노드의 info 값을 모두 출력하는 함수 void PrintNode(NodePtr)를 재귀호출(recursive call)을 이용하여 C 언어와 유사한 언어로 작성하시오. (단, 옆에 노드가 없는 경우에는 info 값을 0으로 간주한다) (5점)
- 2) 이중 연결 리스트의 특성을 이용하여, 주어진 info 값을 가진 노드를 리스트로부터 삭제하는 함수 void DeleteNode(NodePtr, int)를 반복(iteration) 구문을 이용하여 C 언어와 유사한 언어로 작성하시오. (단, 같은 값을 갖는 노드들이 리스트에 여러 개 있을 수 있다) (10점)

제 3 문. 이진 트리의 노드 x에 대해 함수 f(x)는 다음과 같이 정의된다. (단, x.left와 x.right는 각각 노드 x의 왼쪽 및 오른쪽 자식을 나타낸다) (총 30점)

$$f(x) = \begin{cases} -1, & x \text{가 null인 경우} \\ 1 + \min\{f(x.\text{left}), f(x.\text{right})\}, & x \text{가 null이 아닌 경우} \end{cases}$$

(정의) **좌편향 트리**: 이진 트리 T 상의 모든 노드 x에서 $f(x.\text{left}) \geq f(x.\text{right})$ 일 경우 T는 좌편향 트리라고 한다.

- 1) 다음 이진 트리의 각 노드 x에 대해 함수 f(x)의 값을 구하여 각 노드 위에 표시하시오. 또한 각 트리가 좌편향 트리인지 답하고 그 이유를 설명하시오. (10점)



(정의) **right path**: 이진 트리 T 상의 루트 노드에서 시작하여 각 노드의 오른쪽 자식들만으로 연결되는 최장 경로를 right path라고 한다.

- 2) right path의 길이가 0, 즉, 경로 상에 한 개의 노드만 존재할 때, 좌편향 트리이면서 최소 개수의 노드를 갖는 트리를 그리시오. 또한 right path의 길이가 1, 2인 경우 각각에 대해 좌편향 트리이면서 최소 개수의 노드를 갖는 트리를 그리시오. (6점)
- 3) 2)의 결과를 바탕으로 right path의 길이가 r인 좌편향 트리의 노드 개수에 대한 하한값을 제시하시오. (4점)
- 4) 3)의 결론을 증명하시오. (10점)

제 4 문. 다음과 같이 C 언어 형식으로 작성된 프로그램에 대하여 물음에 답하시오.
(총 15점)

```
#define SWAP(x,y,t) ((t)=(x), (x)=(y), (y)=(t))

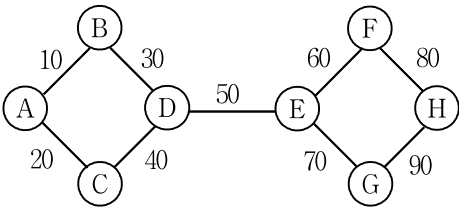
void ga(char list[], int i, int n) {
    int j, temp;
    if(i == n) {
        for(j=0; j<=n; j++)
            printf("%c", list[j]);
        printf("\n");
    }
    else {
        for(j=i; j<=n; j++) {
            SWAP(list[i], list[j], temp);
            ga(list, i+1, n);
            SWAP(list[i], list[j], temp);
        }
    }
}
```

1) 다음 main() 함수의 출력 결과를 구하시오. (10점)

```
main(){
    char s[] = {'a', 'b', 'c'};
    ga(s, 0, 2);
}
```

2) ga() 함수의 시간 복잡도(time complexity)를 함수의 매개변수를 사용하여 O-표기법으로 나타내시오. (5점)

제 5 문. 아래는 어떤 유선 통신망을 그래프로 표현한 것이다. 정점(vertex)은 통신국을, 간선(edge)은 직접 연결이 되어 있음을 나타낸다. 각 간선에 표시된 숫자는 통신선의 연결 길이이다. 다음 물음에 답하시오.
(총 15점)



- 1) 그래프의 정점들 중에서 그 정점에 부속된(incident) 모든 간선과 함께 삭제하면 더 이상 연결 그래프가 되지 않는 정점을 단절점(articulation point)이라고 한다. 위의 그래프에서 단절점을 구하시오. (3점)
- 2) 위의 그래프에는 몇 개의 서로 다른 신장 트리(spanning tree)가 존재하는가? (6점)
- 3) 모든 신장 트리 중 간선의 가중치 합이 최소인 것을 최소비용신장트리(minimum cost spanning tree)라고 한다. 위의 그래프의 최소비용신장트리를 그리고, 총 길이를 구하시오. (6점)

안전행정부 시험출제과장