08. 트리

8.1 트리란

계층적인 자료의 표현에 적합한 자료 구조
탐색트리, 우선순위큐를 위한 힙트리, 인공지능 문제에서 다루는 결정트리
Node와 edge를 이용해서, 사이클을 이루지 않도록 구성한 데이터 구조

트리의 용어

노드 : 트리에서 데이터를 저장하는 기본요소 (데이터와 다른 연결된 노드에 대한 edge 정보를 포함)
루트노드 : 트리 맨위에 있는 노드
루트노드 : 트리 맨위에 있는 노드
간선 혹은 에지 (edge) : 노드와 노드를 연결하는 선
형제노드 : 동일한 Parent Node를 가진 노드
조상 노드와 자손 노드 : 어떤노드에서 루트노드 까지의 경로상에 있는 모든 노드들과
어떤 노드 하위에 연결된 모든 노드

단말노드 : 자식노드가 없는 노드, 자식이 있으면 비단말 노드
노드의 차수 : 어떤 노드가 가지고있는 자식의 수
트리의 차수 : 트리가 가지고 있는 노드의 차수 중에서 가장 큰 차수
레벨 : 트리의 각층에 번호를 매기는것, 루트의 레벨은0이되고 한층씩 내려갈수록
1 증가
트리의 높이 : 트리가 가지고 있는 최대레벨
포레스트 : 트리들의 집합

일반 트리의 구현방법

연결된 구조에서의 노드표현

노드는 항목을 저장하는 데이터 필드와 자식 노드를 가리키는 여러개의 링크를가짐

일반트리 : 임의의 개수의 자식을 가질 수있는 트리

일반트리의 표현방법

두개의 링크를 갖도록 하는 방법

하나는 첫번째 자식을 가리키고, 다른 하나는 다음형제를 가리키게 함

임의의 일반트리 표현 가능하지만, 표현이 복잡해고 성능이 안좋음

2이진트리

· 공집합이거나, 루트와 왼쪽 서브트리, 오른쪽 서브트리로 구성된 노드들의 집합

· 이진트리의 서브트리들은 모두이진트리이어야함

· 모든 노드의 차수가 2이하 즉 최대 2개까지 갖을수있음

· 자식들 사이에 순서 존재

왼쪽노드는 해당 노드보다 작은값, 오른쪽 노드는 해당노드보다 큰값

이진트리의 종류

· 포화이진트리 : 트리의 각레벨 에 노드가 꽉찬 이진트리

· 포화 이진트리에서는 각노드에 순서대로 번호를 붙일수있음

· 높이 k인 포화 이진트리의 전체 노드 개수: $2^{1-1}+2^{2-1}+\dots+2^{k-1}=\sum_{i=0}^{k-1}2^i=2^k-1$

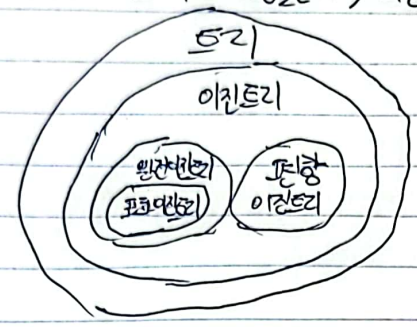
· 완전 이진트리 : 높이가 k인 트리에서 레벨 1부터 k-1 까지는 노드가 모두채워져있고 마지막 레벨에서는 노드가 왼쪽부터 오른쪽으로 채워진 이진트리

· 마지막 레벨은 꽉차있지않아도되지만 중간에 빈곳이 있으면 안됨

편향 이진트리 : 높이가 k인 트리에서 k개의 노드를 가지면서 모든 노드가 왼쪽에서 오른쪽중 한 방향으로만 서브트리를 가지고있는트리

이진트리의 성질

노드의 개수가 n개이면 간선의 개수는 n-1
높이가 h이면 노드의 개수는 h이상 2^h-1개 이하의 노드를가짐
n개 노드의 이진트리 높이는 $\log_2(n+1)$ 이상임 n이하



이진트리 표현 방법 : 배열표현법
- 배열로 표현하기 위하여 인덱스 사용
1) 트리의 높이를 구해 배열을 할당
2) 포화 이진트리 번호를 인덱스로 사용하여 배열에 노드들을저장
- 트리가 불균형 하다면 메모리 낭비가 심함
- 노드 i의 인덱스를 안다면 다른 노드들의 인덱스를 구할수있음
- 배열로 표현법의 단점
- 기억공간의 낭비, 표현할수있는 높이가 배열의 크기에따라 제한됨

노드 i 의
- 부모노드 인덱스 = i/2
- 왼쪽자식 노드인덱스 = 2i
- 오른쪽 자식 노드인덱스 = 2i+1

이진트리의 표현 방법 ; 링크표현법
연결된 구조 이진트리표현
왼쪽 링크와 오른쪽 링크는 각각 왼쪽 자식, 오른쪽 자식을 가리킴

트리의 모든 노드를 방문하는 방법 ; 순회
- 트리에 속하는 모든 노드를 한번씩 방문 하는 것
- 선형 자료구조 순회가 단순
- 트리는 다양한 방법이 있음
- 이진트리의 기본순회는 3가지
- 루트와 왼쪽 서브트리, 오른쪽 서브트리를 어떤 순서로 방문하느냐로 나눔

1) 전위 순회 ; VLR
2) 중위순회 ; LVR
3) 후위순회 ; LRV

각 서브 트리도 이진트리이므로 동일한 순회방법이 적용

전위 순회
- 루트 노드 (V) → 왼쪽 서브트리 (L) → 오른쪽 서브트리(R)
- 모든 서브트리에 같은 일관성을 적용

중위순회
- 왼쪽 서브 트리 (L) → 루트노드 (V) → 오른쪽 서브트리 (R)
- 모든 서브트리에 같은 일관성을 적용

후위 순회
왼쪽 서브트리 (L) → 오른쪽 서브트리 (R) → 루트노드 (V)
모든 서브트리에 같은 일관성을 적용

레벨 순회
노드를 레벨 순으로 검사하는 순회 방법
V 큐를 사용해 구현
V 순환을 사용하지 않음

트리의 전체 노드 개수 구하기
모든 노드들을 한번씩은 방문해야함 → 순환사용
왼쪽 서브 트리의 노드수 + 오른쪽 서브트리의 노드수 + 루트노드의 수 (1)
후위 순회 방식의 순환 호출

트리의 전체 단말 노드 개수 구하기
· 왼쪽 자식 or 오른쪽 자식이 모두 None 인 노드
· 순환을 사용
· 두 서브트리의 단말 노드 개수를 계산하고, 이를 반환

트리의 높이 구하기
순환을 사용
후위 순회 구조 사용
루트노드를 포함한 현재 트리의 높이는 과연 서브트리의 높이들에서 더 높은 값에 1을 더함

모르스 부호
· 1830년대 미국의 사무엘 모르스
· 도트(점)과 대시(선)의 고합으로 구성된 메시지 전달 방법
· 메시지를 전선을 통해 장거리로 전송할 수 있도록 함

모르스 부호 : 인코딩
인코딩 : 알파벳을 모르스 부호로 변환
표에서 바로 찾을 수있음, key값과 1:1대응
시간복잡도 O(1)

모르스 부호 : 디코딩
디코딩 : 모르스 코드가 있을때 알파벳을 추출하는 방법
표에서 바로 찾을 수없고 순차탐색을 통해 알아야함
시간복잡도 O(n)

결정트리로 이용한 모르스코드의 디코딩
결정트리 : 여러단계의 복잡한 고려를 갖는 문제에 대해 고려와 탐색을
트리형태로 나타내면
점은 왼쪽 자식, 선(-)은 오른쪽자식
디코딩시간 복잡도 : 트리높이에 비례 -> O(logn)

BST힙트리
힙이란?
'데이'와 모습이 비슷한 완전이진트리기반의 자료구조
가장큰 또는 작은 값을 빠르게 찾아내도록 만들어진 자료구조
최대힙, 최소힙

최대힙 : 부모노드의 키값이 자식노드의 키값보다 크거나 같은 완전이진트리
최소힙 : 부모노드의 키값이 자식노드의 키값보다 작거나 같은 완전이진트리

힙의 삽입연산

삽입연산은 힙의 순서특성 (최대 or 최소 힙)과 트리의 형태적특성 만족시유지

회사의 승진과정과 유사 (Shift-UP 혹은 Up-heap)

· 회사에서 신입사원이 들어오면 일단 말단위치에 앉힘

· 신입사원의 능력을 보고 위로 승진시킴

시간복잡도 : O(logn)

1) 새로운 항목이 들어오면 힙의 마지막 노드의 다음위치에 삽입

2) 마지막에 삽입된 노드를 부모노드들과 교환하여 힙의 순서특성을 만족 시켜주는 과정을 거쳐 최대 트리의 높이만큼의 이동이 필요

힙의 삭제연산

힙에서 삭제 연산은 루트 노드를 꺼내며 힙의 상태 유지

회사에서 사장자리가 비면 말단사원을 올리고 순차적으로 강등 (배꼽잡고 웃음)

Down-heap

시간복잡도 : O(log n)

힙의 삭제연산

1) 루트노드에 마지막노드를 덮고 완전이진트리 조건은 만족 힙의 순서 특성은 만족 X

2) 루트 노드를 자식과 비교하여 계속 다운교환 자식들 중에서 더 큰 자식이 교환할 때까지 이과정을 자식이 없거나 자식이 더작을 때까지 반복

배열을 이용한 힙의 표현

힙은 보통 배열을 이용하여 구현

완전이진트리 → 각 노드에 번호 부여 → 배열의 인덱스

배열을 이용한 힙의 표현

부모 노드와 자식노드의 관계

✓ 부모의 인덱스 = (자식의 인덱스)/2

✓ 왼쪽 자식의 인덱스 = (부모의 인덱스)*2

✓ 오른쪽 자식의 인덱스 = (부모의 인덱스)*2+1

우선순위 큐 구현

힙을 이용한 우선 순위 큐의 구현과 가장 좋은 방법

허프만 코드

- 허프만 알고리즘 에 의해 최적 이진 코드
- 이진트리는 각 글자의 빈도가 알려져 있는 메시지의 내용을 압축하는 데 사용

이러한 이진트리 구 허프만 코딩트리

아스키 코드는 모든 문자를 동일한 비트수로 표현 -> 압축이 많이 되어서는 효율적이지 못함

자주 사용하는 문자 에는 적은 비트수를, 그렇게 않은 문자에는 많은 비트수를 부여 -> 효율적

고정길이 코드와 가변 길이 코드의 비교

텍스트 A ~ J로 이루어진 빈도수

1개의 문자 표현하기위해 고정 길이코드 사용시 최소 4비트 필요

가변 길이 코드 사용하면 글자마다 코드의 비트수가 달라짐

모든 가변 길이 코드가 다른코드의 첫부분이 아니어야 함