소프트웨어학부
20162820 김영민

<1>

(1) False
(2) False
(3) False
(4) True
(5) True
(6) False
(7) False
(8) True
(9) False
(10) True

<2>

예를들어 크기가 n인 arr 배열의 각 인덱스에 담긴 값을 찾기 위한 코드가 있다

```
for(int i=0; i<n; i++){
        System.out.println(arr[n]);
        }
```

이코드에서 수행시간은 i가 0 부터 n-1까지 실행하니깐 n번 실행한다 그래서 시간 복잡도는 O(n)으로 표현가능하다 빅오인 O로 표현하는 것은 최악의 경우를 설명하는 것이므로 이렇게 표현한 것이다.

(3)

3차식 $A = 5x^4 + 2x^3 + 4$, $B = 6x^4 + 4x^2 + 7$ 이라는 식이 있다

두항을 구하기 위해선 다른 배열에 A와 B의 각 지수가 일치하는 계수의 합을 구해야한다

Ex)

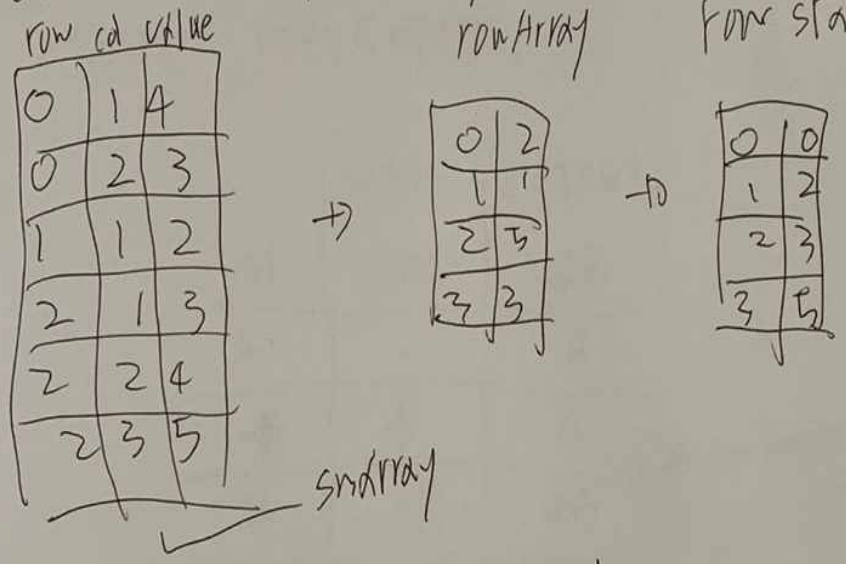
```
int C[] = new int[10]; 크기가 10인 임의의 배열
int A[] = new int[5];
int B[] = new int[5];
A = {0, 4, 0, 2, 5};
B = {0, 7, 4, 0, 6};
int apos = 0;, int bpos = 0;
for(int i=0; i<C.length; i++){
    ~~int C[i] = A[i] + B[i];~~
    if( i > A.length || i > B.length){
        break;
    }
    else{
        C[i] = A[i] + B[i];
        Apos += 1;
        Bpos += 1;
    }
}
if(Apos < A.length){
    C[Apos] = A[Apos]; }
if(Bpos < B.length){
    C[Bpos] = B[Bpos]; }
```

<3>
3-(2)

희소행렬은 smArray 라는 배열에 0이아닌 Matrix 값을 col, row, value 의 값을 지니는 배열 어레이를 만든다.

<빠르게 전치 시키는 알고리즘>

Ex) terms = 5 // smArray 크기



이렇식으로 rowstart를 만들어서.

```
for(int i=0, i<terms, i++){
    int j = rowstart[smarray[~~rowstar~~i].col] // 2
    ~~b.rows~~int k = rowArray[i];
    b.smarray[k].col = smarray[i].row
    b.smarray[k].row = smarray[i].col
    b.smarray[k].value = smarray[i].value
    ~~rowstart[i]=1~~;
    rowstart[smarray[i].col]++;
        표현한다.
```

<4>

(4-1)

A+(B×C)−D

~~A+(B×C)−D~~ 라는 중위표기식을

일단 모두 괄호를 쳐준다 ((A+(B×C)−D) 이다.

후위표기식은 ABC×+D− ~~ABC+×D−~~ 가 된다 알고리즘을

설명하면 STACK을 이용한다.

first	STACK	output
	X없음	없음
A	·	A
+	+	A
B	+(	AB
×	+(×	AB
C	+(×	ABC
)	+	ABC×
−	−	ABC×+
D	−	ABC×+D
·	·	ABC×+D−

← 사이에 (괄호 | +(| A

이런식으로 STACK의 FILO 기법 이용!

A,B,C 문자는 바로출력, 연산자, 괄호는 스택삽입!

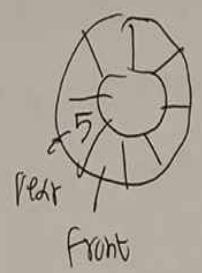
4-(2)

이런 원형큐가 있다, capacity=8 이다 (용량)

최초는 front와 rear 모두 = 0 이다.

Element들가 push 되면, rear=(rear+1) % capacity 이다

따라서

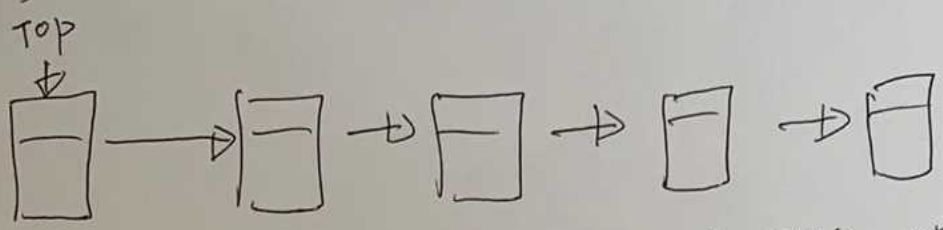


front=0, rear=1.

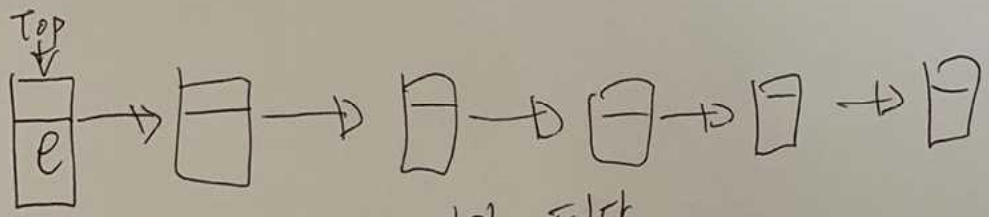
의 형식이된다. pop의 경우는 rear -=1을하여 원소를 Delete 해준다. 여기서 rear=front 이면 isEmpty 인상태가 되고 (rear+1) % capacity = front 이면 full 인상태가 된다. front는 항상 빈상태이므로 최대 가능원소는 n-1개 이다.

<5>

5-(1)



이런 chain node가 존재함에 STACK 구조는 맨앞의 노드는 Top을 가진다 그리고 push(e) e라는 원소가 push가 되면 ~~chainnod~~ newnode = ~~cha~~ new chainnode(e,Top); Top을 링크로 하는 new node가 생성된다. 그리고 Top = new node를 해주어 제일 앞에 있는 newnode를 Top으로 설정한다



그러면 이렇게 push가 된다.

POP의 경우는

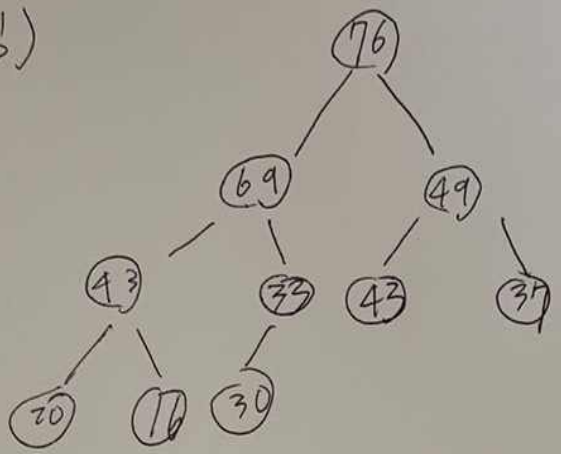
Chainnode X = Top

Top.link = Top

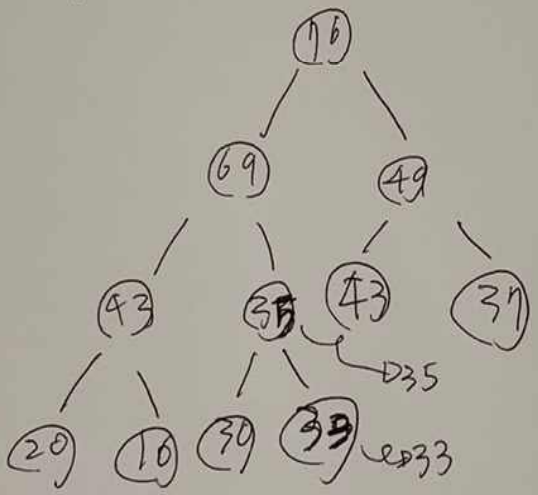
Delete X 를통해 맨앞노드를 없애준다.

5-(2)

Max Heap 그림)



① 35 원소 push



② pop이 한번 이루어진 상태