

Stack 자료구조

1. 스택(Stack)의 개념

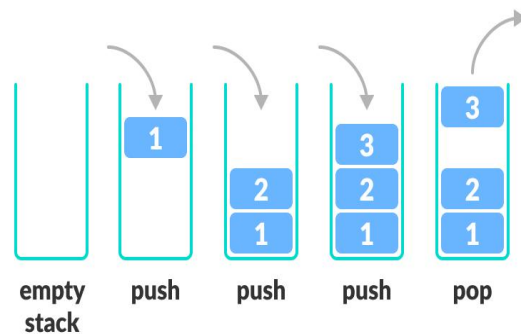
한 쪽 끝에서만 자료를 넣고 뺄 수 있는 LIFO(Last In First Out) 형식의 자료 구조이다. 즉, 가장 최근에 스택에 추가한 항목이 가장 먼저 제거되는 자료구조이다.

2. 스택(Stack)의 연산(자바 기준)

- 1) push(item): item 하나를 스택의 가장 윗 부분에 추가한다.
- 2) pop(): 스택에서 가장 위에 있는 항목을 제거한다.
- 3) peek(): 스택의 가장 위에 있는 항목을 반환한다.
- 4) empty(): 스택이 비어 있을 때에 true를 반환한다.

3. 자바스크립트에서 스택사용하기

```
function solution(){
    stack=[]
    stack.push(1);
    stack.push(2);
    stack.push(3);
    console.log(stack.pop());
    console.log(stack[stack.length-1]) //스택의 가장 위에 있는 항목 확인
    console.log(stack.length==0) //스택이 비어 있는지 확인
    console.log(stack.pop());
    console.log(stack.pop());
    console.log(stack.length==0)
    return stack;
}
console.log(solution());
```



올바른 괄호

괄호 문자열이 입력되면 올바른 괄호이면 "YES", 올바르지 않으면 "NO"를 출력합니다.
(())() 이것은 괄호의 쌍이 올바르게 위치하는 거지만, (()())은 올바른 괄호가 아니다.

입출력 예:

s	result
"(())()"	"YES"
"(()())"	"NO"
"()())"	"NO"
"()())"	"NO"

제한사항:

- 문자열 s의 길이는 100을 넘지 않습니다.

프로그램 기본 구조

```
function solution(s){  
    let answer="YES";  
  
    return answer;  
}  
console.log(solution("(()())()"));
```

괄호문자제거

입력된 문자열에서 소괄호 () 사이에 존재하는 모든 문자를 제거하고 남은 문자만 출력하는 프로그램을 작성하세요.

입출력 예:

s	result
"(A(BC)D)EF(G(H)(IJ)K)LM(N)"	"EFLM"
"((AF)B)(D)E(F)"	"E"
"(Aab)(EsG)FFD"	"FFD"
"(YYYDD)sh(kkf(hfs))"	"sh"

제한사항:

- 문자열 s의 길이는 100을 넘지 않습니다.

연속된 문자 지우기

문자열 s 가 주어지면 이웃한 두 개의 문자가 같으면 두 문자를 제거합니다. 이 과정을 반복해서 최종적으로 남는 문자만으로 이루어진 문자열을 출력하는 프로그램을 작성하세요.

만약 "acbbcaa"라는 문자열이 주어진다면 최초 bb가 연속되어 있어 제거하고 나면 "acca"가 되고, 다시 cc가 연속되어 제거하면 "aa"가 되고 "a"연속되어 제거하면 "a"가 최종적으로 남습니다.

입출력 예:

s	result
"acbbcaa"	"a"
"bacccaba"	"bacaba"
"aabaababbaa"	"a"
"bcaacccbaabccabbaa"	"ba"

제한사항:

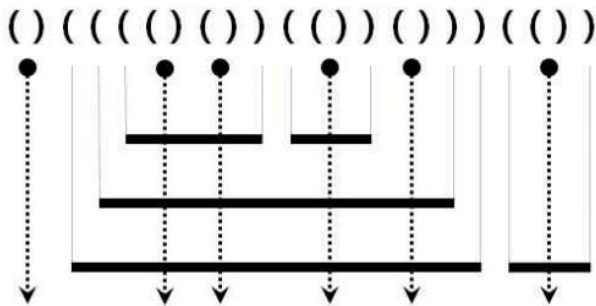
- 문자열 s 의 길이는 100,000을 넘지 않습니다.
- 문자열 s 는 소문자로만 이루어져 있습니다.

쇠막대기

여러 개의 쇠막대기를 레이저로 절단하려고 한다. 효율적인 작업을 위해서 쇠막대기를 아래에서 위로 겹쳐 놓고, 레이저를 위에서 수직으로 발사하여 쇠막대기들을 자른다. 쇠막대기와 레이저의 배치는 다음 조건을 만족한다.

- 쇠막대기는 자신보다 긴 쇠막대기 위에만 놓일 수 있다. - 쇠막대기를 다른 쇠막대기 위에 놓는 경우 완전히 포함되도록 놓되, 끝점은 겹치지 않도록 놓는다.
- 각 쇠막대기를 자르는 레이저는 적어도 하나 존재한다.
- 레이저는 어떤 쇠막대기의 양 끝점과도 겹치지 않는다.

아래 그림은 위 조건을 만족하는 예를 보여준다. 수평으로 그려진 굵은 실선은 쇠막대기이고, 점은 레이저의 위치, 수직으로 그려진 점선 화살표는 레이저의 발사 방향이다.



이러한 레이저와 쇠막대기의 배치는 다음과 같이 괄호를 이용하여 왼쪽부터 순서대로 표현할 수 있다.

1. 레이저는 여는 괄호와 닫는 괄호의 인접한 쌍 ‘()’으로 표현된다. 또한, 모든 ‘()’는 반드시 레이저를 표현한다.
2. 쇠막대기의 왼쪽 끝은 여는 괄호 ‘(’로, 오른쪽 끝은 닫힌 괄호 ‘)’로 표현된다.

위 예의 괄호 표현은 그림 위에 주어져 있다.

쇠막대기는 레이저에 의해 몇 개의 조각으로 잘려지는데, 위 예에서 가장 위에 있는 두 개의 쇠막대기는 각각 3개와 2개의 조각으로 잘려지고, 이와 같은 방식으로 주어진 쇠막대기들은 총 17개의 조각으로 잘려진다.

쇠막대기와 레이저의 배치를 나타내는 괄호 표현이 매개변수 s 에 주어졌을 때, 잘려진 쇠막대기 조각의 총 개수를 구하는 프로그램을 작성하시오.

입출력 예:

s	result
"()((()())(())())(())"	17
"(((()()())(())())(())())"	24
"((((())))"	8
"((((()()))(())(())())"	12

제한사항:

- 쇠막대기와 레이저의 배치를 나타내는 괄호 표현이 공백없이 문자열로 주어집니다.
- 괄호 문자의 개수는 최대 100,000이다.

Queue 자료구조

1. 큐(Queue)의 개념

한 쪽 끝에서 자료가 삽입되고, 반대쪽 끝에서 자료가 삭제되는 FIFO(First In First Out) 형식의 자료 구조이다.

즉, 먼저 추가한 항목이 가장 먼저 제거되는 자료구조이다.

2. 큐(Queue)의 연산(자바스크립트 기준)

1) push(item): item 하나를 큐의 뒷 부분에 추가한다.

2) shift(): 큐에 가장 앞에 있는 항목을 제거하고 반환한다.

3. 자바스크립트에서 큐사용하기

```
function solution(){
    queue=[]
    queue.push(1);
    queue.push(2);
    queue.push(3);
    console.log(queue.shift());
    console.log(queue.length===0) //큐가 비어 있는지 확인
    console.log(queue.shift());
    console.log(queue.shift());
    console.log(queue.length===0)
    return queue;
}
console.log(solution());
```

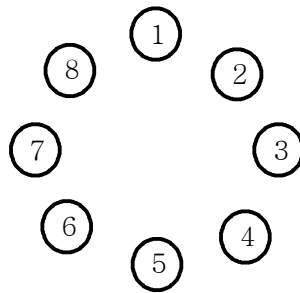
공주 구하기

정보 왕국의 이웃 나라 외동딸 공주가 숲속의 괴물에게 잡혀갔습니다.

정보 왕국에는 왕자가 N명이 있는데 서로 공주를 구하러 가겠다고 합니다. 정보왕국의 왕은 다음과 같은 방법으로 공주를 구하러 갈 왕자를 결정하기로 했습니다.

왕은 왕자들을 나이 순으로 1번부터 N번까지 차례로 번호를 매긴다. 그리고 1번 왕자부터 N번 왕자까지 순서대로 시계 방향으로 돌아가며 동그랗게 앉게 한다. 그리고 1번 왕자부터 시계방향으로 돌아가며 1부터 시작하여 번호를 외치게 한다. 한 왕자가 K(특정숫자)를 외치면 그 왕자는 공주를 구하러 가는데서 제외되고 원 밖으로 나오게 된다. 그리고 다음 왕자부터 다시 1부터 시작하여 번호를 외친다.

이렇게 해서 마지막까지 남은 왕자가 공주를 구하러 갈 수 있다.



예를 들어 총 8명의 왕자가 있고, 3을 외친 왕자가 제외된다고 하자. 처음에는 3번 왕자가 3을 외쳐 제외된다. 이어 6, 1, 5, 2, 8, 4번 왕자가 차례대로 제외되고 마지막까지 남게 된 7번 왕자에게 공주를 구하러갑니다.

N과 K가 주어질 때 공주를 구하러 갈 왕자의 번호를 출력하는 프로그램을 작성하시오.

입출력 예:

N	K	result
8	3	7

제한사항:

- $N(5 \leq N \leq 1,000)$
- $K(2 \leq K \leq 9)$