

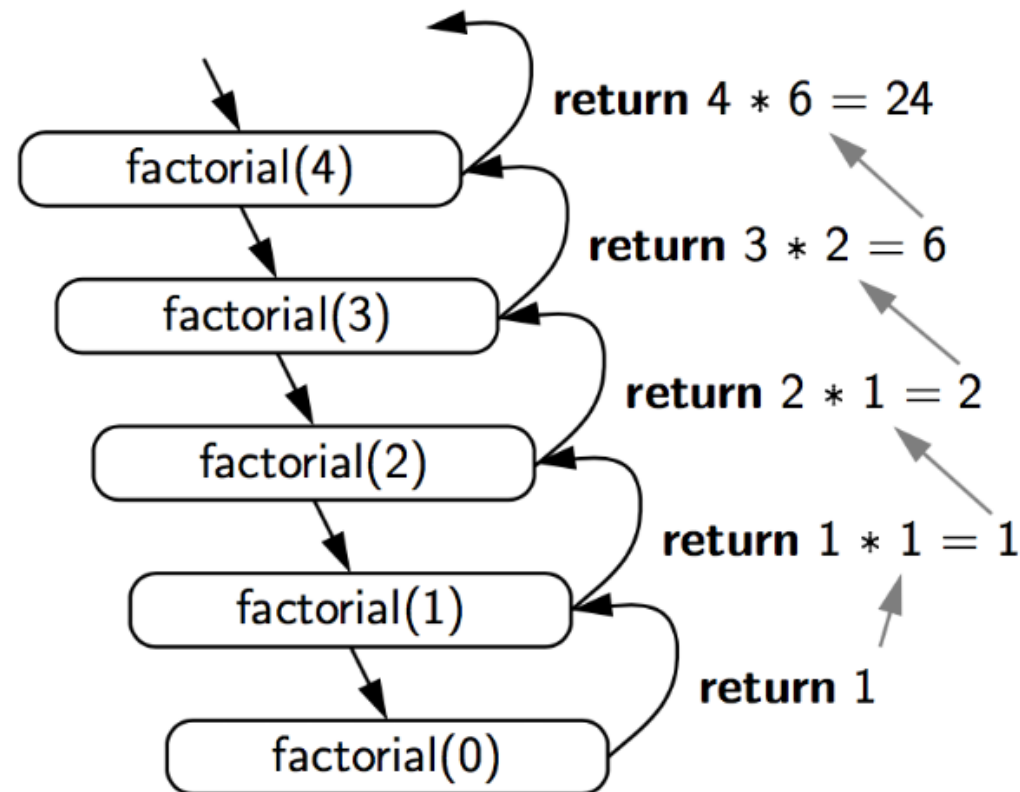


SW 역량 테스트 대비반 - 4회차

2020년 07월 29일

지난 수업 정리

1. 재귀함수



재귀적 설계는 추후
Dynamic
Programming의
기본이 됩니다.

2. 탐색 알고리즘

- 순차탐색
- 이진탐색
- DFS (깊이우선탐색) : 스택이나 재귀함수로 구현
- BFS (너비우선탐색) : 큐로 구현

오늘 할 내용

구분			
1주차 (07/08)		현재까지 배운 거 기반으로 예제 문제 풀어보기	
2주차 (07/15)			
3주차 (07/22)			
4주차 (07/29)			
5주차 (08/05)			
6주차 (08/12)		공간 점진 문제 테스트	
7주차 (08/19)		분할과 정복 강의 및 실습	
8주차 (08/26)		동적 계획법 강의 및 실습	
(옵션) 9주차 ~		그래프 알고리즘 강의 및 실습	
		실전 감각 기르기 (실전 기출 기반 문제풀이 집중반)	

문제 1

<https://www.acmicpc.net/problem/2493>

문제

KOI 통신연구소는 레이저를 이용한 새로운 비밀 통신 시스템 개발을 위한 실험을 하고 있다. 실험을 위하여 일직선 위에 N개의 높이가 서로 다른 탑을 수평 직선의 왼쪽부터 오른쪽 방향으로 차례로 세우고, 각 탑의 꼭대기에 레이저 송신기를 설치하였다. 모든 탑의 레이저 송신기는 레이저 신호를 지표면과 평행하게 수평 직선의 왼쪽 방향으로 발사하고, 탑의 기둥 모두에는 레이저 신호를 수신하는 장치가 설치되어 있다. 하나의 탑에서 발사된 레이저 신호는 가장 먼저 만나는 단 하나의 탑에서만 수신이 가능하다.

예를 들어 높이가 6, 9, 5, 7, 4인 다섯 개의 탑이 수평 직선에 일렬로 서 있고, 모든 탑에서는 주어진 탑 순서의 반대 방향(왼쪽 방향)으로 동시에 레이저 신호를 발사한다고 하자. 그러면, 높이가 4인 다섯 번째 탑에서 발사한 레이저 신호는 높이가 7인 네 번째 탑이 수신을 하고, 높이가 7인 네 번째 탑의 신호는 높이가 9인 두 번째 탑이, 높이가 5인 세 번째 탑의 신호도 높이가 9인 두 번째 탑이 수신을 한다. 높이가 9인 두 번째 탑과 높이가 6인 첫 번째 탑이 보낸 레이저 신호는 어떤 탑에서도 수신을 하지 못한다.

탑들의 개수 N과 탑들의 높이가 주어질 때, 각각의 탑에서 발사한 레이저 신호를 어느 탑에서 수신하는지를 알아내는 프로그램을 작성하라.

입력

첫째 줄에 탑의 수를 나타내는 정수 N이 주어진다. N은 1 이상 500,000 이하이다. 둘째 줄에는 N개의 탑들의 높이가 직선상에 놓인 순서대로 하나의 빈칸을 사이에 두고 주어진다. 탑들의 높이는 1 이상 100,000,000 이하의 정수이다.

출력

첫째 줄에 주어진 탑들의 순서대로 각각의 탑들에서 발사한 레이저 신호를 수신한 탑들의 번호를 하나의 빈칸을 사이에 두고 출력한다. 만약 레이저 신호를 수신하는 탑이 존재하지 않으면 0을 출력한다.

예제 입력 1 [복사](#)

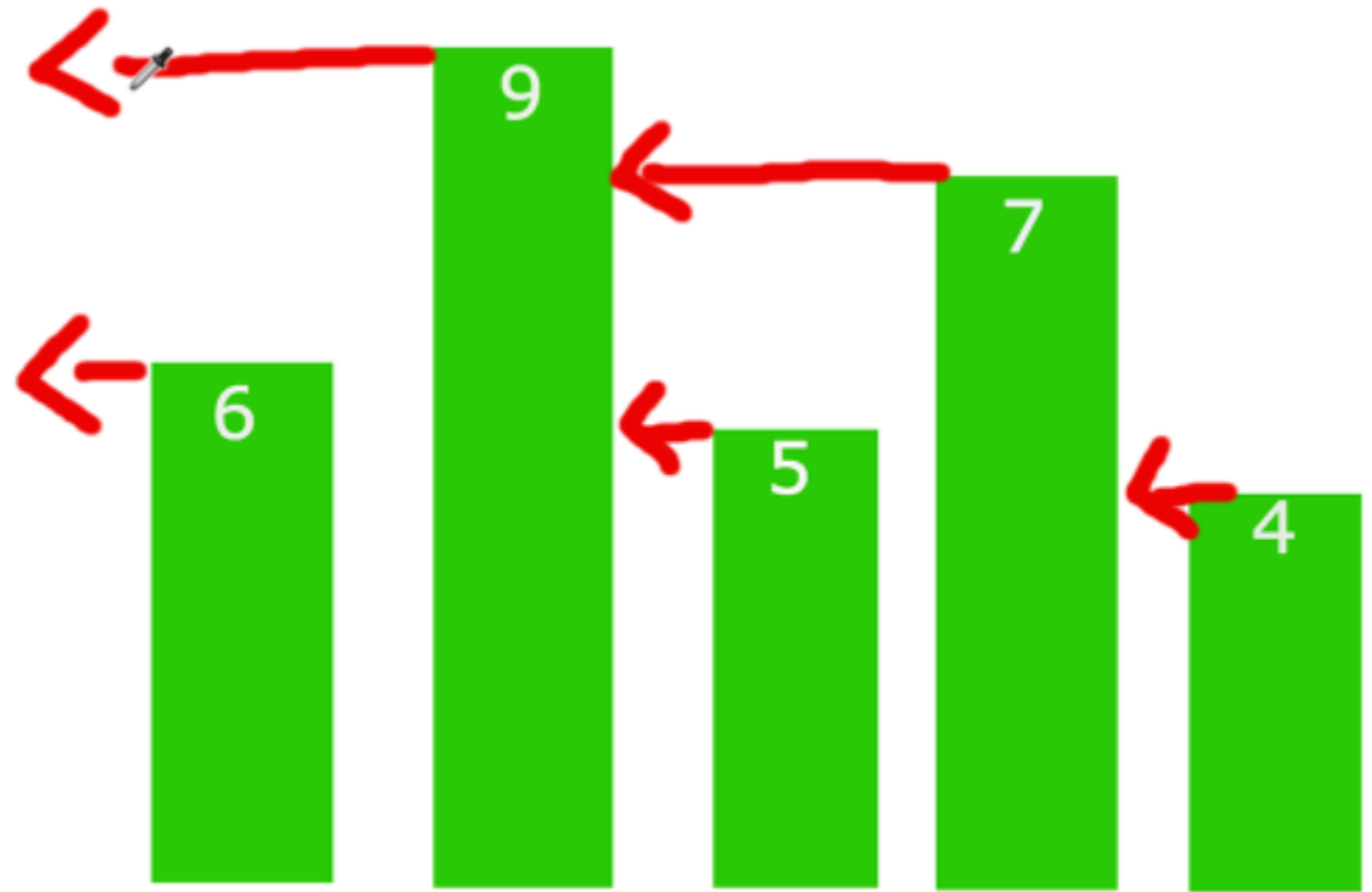
```
5
6 9 5 7 4
```

예제 출력 1 [복사](#)

```
0 0 2 2 4
```

문제 1

아이
디어



1. 좌측 기준으로 자신보다 큰 탑을 찾을 것이므로 스택과 페어(위치, 높이 저장용)를 사용
2. height가 st.top보다 크면 stack에서 큰 값을 찾은 후, 해당 위치를 저장
3. 이때, 중간에 있는 스택 값(height보다 작은 값)은 사용되지 않을 것이므로 st.pop으로 제거
4. height가 st.top보다 작으면 해당 위치를 저장 (바로 옆의 좌측 큰 값)
5. height를 stack에 넣는다. (나중에 사용될 수 있으므로)

답안

<https://onlinegdb.com/HJexaAXW6E>

문제 2

프린터

문제 설명

일반적인 프린터는 인쇄 요청이 들어온 순서대로 인쇄합니다. 그렇기 때문에 중요한 문서가 나중에 인쇄될 수 있습니다. 이런 문제를 보완하기 위해 중요도가 높은 문서를 먼저 인쇄하는 프린터를 개발했습니다. 이 새롭게 개발한 프린터는 아래와 같은 방식으로 인쇄 작업을 수행합니다.

1. 인쇄 대기목록의 가장 앞에 있는 문서(j)를 대기목록에서 꺼냅니다.
2. 나머지 인쇄 대기목록에서 j보다 중요도가 높은 문서가 한 개라도 존재하면 j를 대기목록의 가장 마지막에 넣습니다.
3. 그렇지 않으면 j를 인쇄합니다.

예를 들어, 4개의 문서(A, B, C, D)가 순서대로 인쇄 대기목록에 있고 중요도가 2 1 3 2 라면 C D A B 순으로 인쇄하게 됩니다.

내가 인쇄를 요청한 문서가 몇 번째로 인쇄되는지 알고 싶습니다. 위의 예에서 C는 1번째로, A는 3번째로 인쇄됩니다.

현재 대기목록에 있는 문서의 중요도가 순서대로 담긴 배열 priorities와 내가 인쇄를 요청한 문서가 현재 대기목록의 어떤 위치에 있는지를 알려주는 location이 매개변수로 주어질 때, 내가 인쇄를 요청한 문서가 몇 번째로 인쇄되는지 return 하도록 solution 함수를 작성해주세요.

제한사항

- 현재 대기목록에는 1개 이상 100개 이하의 문서가 있습니다.
- 인쇄 작업의 중요도는 1~9로 표현하며 숫자가 클수록 중요하다는 뜻입니다.
- location은 0 이상 (현재 대기목록에 있는 작업 수 - 1) 이하의 값을 가지며 대기목록의 가장 앞에 있으면 0, 두 번째에 있으면 1로 표현합니다.

입출력 예

4	6
2 1 3 2	1 1 9 1 1 1
C	a
1	5

입출력 예 설명

예제 #1 : 입력

- 첫번째 줄 : 인쇄 문서 개수 (4)
- 두번째 줄 : 인쇄 문서의 중요도 (2, 1, 3, 2)
- 세번째 줄 : 내가 요청한 문서 (알파벳 소문자 : c)

예제 #1 : 출력

- 내가 요청한 문서의 인쇄 순서 (1)

문서도 인쇄됩니다.

문제 2

아이
디어

1. 우선순위에 따라 대기 목록의 뒤로 보내기 위해 Queue를 사용
2. 현재 값이 max 우선순위가 아니면 queue의 뒤에 보내고..
3. max 우선순위라면 해당 위치를 별도 출력 배열(sorted)에 저장
4. 찾고자 하는 알파벳과 맞는 위치를 출력 배열에서 찾아서 해당 위치를 출력

다른 방법은?

1. 2차원 벡터를 입력 (알파벳, 우선순위)
2. C++ STL에 있는 priority_queue를 이용하여 자동 소팅 후,
3. priority_queue를 하나씩 pop 하면서 찾고자 하는 알파벳이 나오면 해당 위치의 인덱스를 출력

답안

<https://onlinegdb.com/ry6K2qL5U>

문제 3

계단 오르기

길동이는 n 개의 단으로 구성된 계단을 오르려고 한다.

길동이는 계단을 오를 때 기분에 따라서 한 번에 1단 또는 2단을 올라갈 수 있다.

계단의 크기 n 이 주어질 때, 길동이가 이 계단을 올라갈 수 있는 모든 경우의 수를 구하는 프로그램을 작성하시오.

만약 계단이 3개라면 길동이는 1, 1, 1로 올라가는 법과 1, 2로 올라가는 법, 2, 1로 올라가는 법의 3가지 서로 다른 방법이 있다.

입력

계단의 수 n 이 입력된다(단 n 은 20보다 작은 자연수).

출력

길동이가 계단을 오르는 모든 방법의 수를 출력한다.

입력 예	출력 예
3	3

4

5

5

8

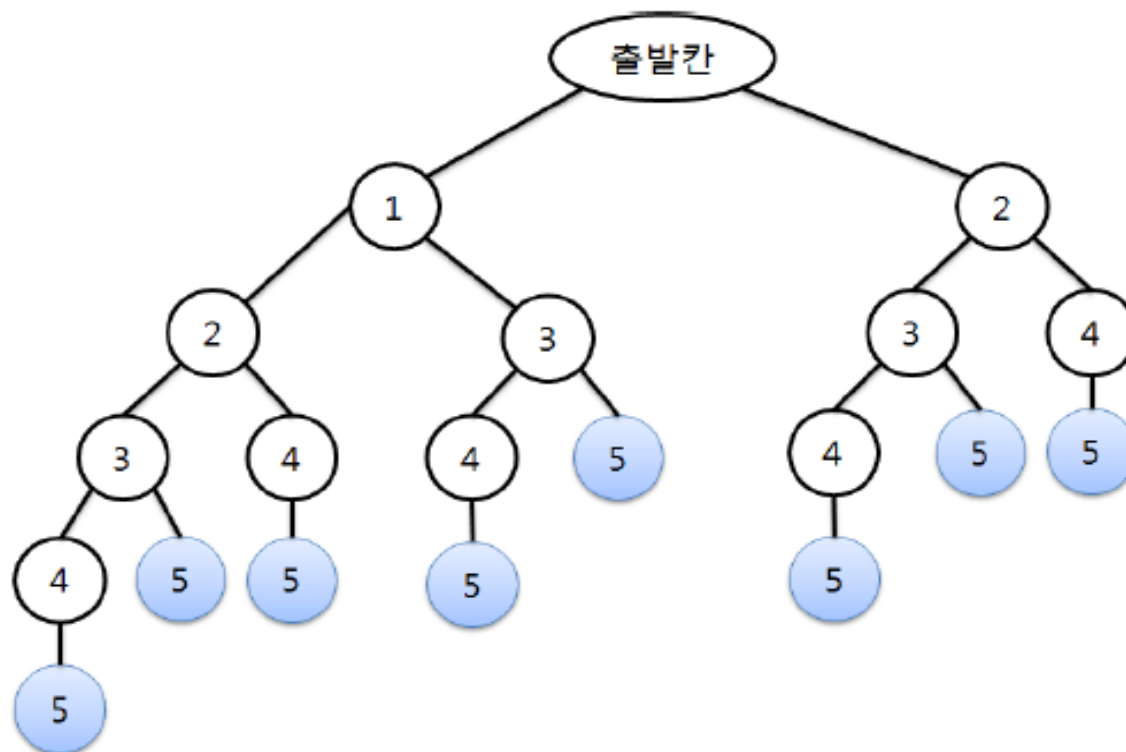
문제 3

아이
디어

이 문제도 비선형구조로 전체탐색을 하여 해를 구할 수 있다. 현재 상태에서 1칸 또는 2칸을 올라갈 수 있으므로, 탐색구조를 다음과 같이 설정할 수 있다. 단, 주의할 점은 정확하게 n 칸에 도착했을 때만 한 가지 경우로 처리해야한다는 점이다. 예를 들어 도착점까지 한 칸 남았을 경우에는 2칸을 올라갈 수 없다.



위의 탐색 과정으로 5칸의 계단을 오르는 과정을 보면 다음과 같다.

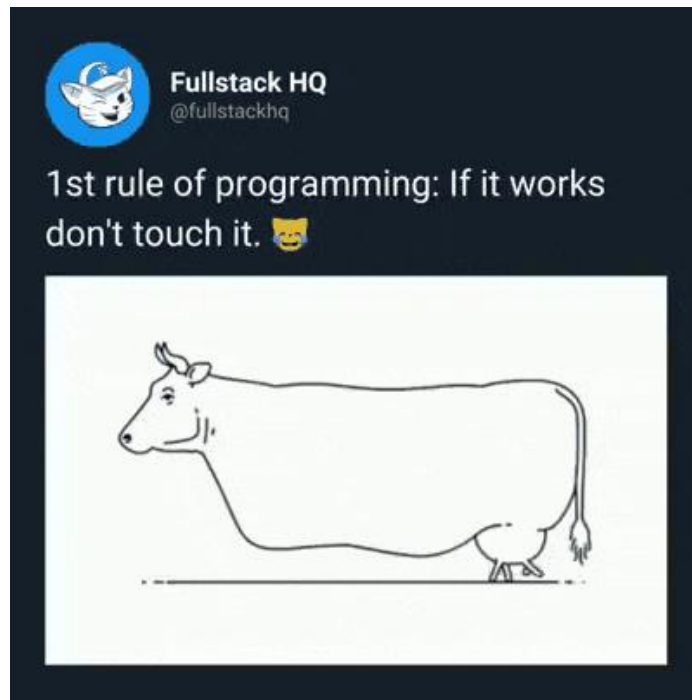


트리 구성이
가변적인 상황에서
Stack을 이용한
DFS 구현은?

답안

<https://onlinegdb.com/BkEySVZp4>

쉬어가는?



채용담당자가 뽑은 자기소개서 및 면접용 답안 최악 vs 최고

◇Worst(복수응답)

자료: 취업포털 커리어(www.career.co.kr)

1	"엄격하신 아버지와 자상한 어머니 사이에서 태어나~"	71.1%
2	"~(뽑아/시켜)만 주신다면 무슨 일이든 하겠습니다."	48.0%
3	"귀사가~ 귀사에 항상 관심이 있었습니다."('귀사' 반복 문장)	35.9%
4	"솔직히 말씀드리면~"	31.7%
5	"나는~저는 약속드립니다."('저는/나는' 으로 시작하는 문장 반복)	27.4%
6	"학창시절 결석 한 번 없이 성실하게 생활했으며~"	22.1%
7	"귀사를 통해서 발전하도록 하겠습니다."	14.1%

◇Best(복수응답)

1	"~(이 회사/직무)에 지원하기 위해 ~준비를 했습니다."	48.5%
2	"책임감을 갖고 있기 때문에"	27.1%
3	"~했지만 ~을 통해 극복했습니다."	23.9%
4	"항상 웃음을 잃지 않고 긍정적으로"	22.1%
5	"~년 후 ~분야에서 전문가가 되고 싶습니다."	20.1%
6	"친구가 많고 대인관계가 좋은 편입니다."	18.8%
7	"~한 경험을 통해 ~을 키웠습니다."	17.8%



f /techindustan t /techindustan i /techindustan

문제 4

타겟 넘버

문제 설명

n개의 음이 아닌 정수가 있습니다. 이 수를 적절히 더하거나 빼서 타겟 넘버를 만들려고 합니다. 예를 들어 [1, 1, 1, 1, 1]로 숫자 3을 만들려면 다음 다섯 방법을 쓸 수 있습니다.

```
-1+1+1+1+1 = 3
+1-1+1+1+1 = 3
+1+1-1+1+1 = 3
+1+1+1-1+1 = 3
+1+1+1+1-1 = 3
```

사용할 수 있는 숫자가 담긴 배열 numbers, 타겟 넘버 target이 매개변수로 주어질 때 숫자를 적절히 더하고 빼서 타겟 넘버를 만드는 방법의 수를 return 하도록 solution 함수를 작성해주세요.

제한사항

- 주어지는 숫자의 개수는 2개 이상 20개 이하입니다.
- 각 숫자는 1 이상 50 이하인 자연수입니다.
- 타겟 넘버는 1 이상 1000 이하인 자연수입니다.

입출력 예

```
5
1 1 1 1 1
3
5
```

문제 4

아이
디어

1. 모든 경우의 수를 생각해야 하므로 전체탐색 문제로 보임
2. DFS 형태로 문제를 풀되 중복된 값과 트리의 크기가 확정되지 않았으므로 재귀함수로 구현
3. 재귀함수 구현 시, 음수로 시작하는 경우와 양수로 시작하는 경우를 모두 고려해야 함
4. n 을 증가시키면서 해당 위치(n)의 수를 $+/-$ 시키면서 재귀함수를 호출
5. Target 값과 같으면 total count를 증가

답안

<https://onlinegdb.com/SkEH94-pV>

문제 5

단어 변환

문제 설명

두 개의 단어 `begin`, `target`과 단어의 집합 `words`가 있습니다. 아래와 같은 규칙을 이용하여 `begin`에서 `target`으로 변환하는 가장 짧은 변환 과정을 찾으려고 합니다.

1. 한 번에 한 개의 알파벳만 바꿀 수 있습니다.
2. `words`에 있는 단어로만 변환할 수 있습니다.

예를 들어 `begin`이 "hit", `target`가 "cog", `words`가 ["hot","dot","dog","lot","log","cog"] 라면 "hit" -> "hot" -> "dot" -> "dog" -> "cog"와 같이 4단계를 거쳐 변환할 수 있습니다.

두 개의 단어 `begin`, `target`과 단어의 집합 `words`가 매개변수로 주어질 때, 최소 몇 단계의 과정을 거쳐 `begin`을 `target`으로 변환할 수 있는지 return 하도록 `solution` 함수를 작성해주세요.

제한사항

- 각 단어는 알파벳 소문자로만 이루어져 있습니다.
- 각 단어의 길이는 3 이상 10 이하이며 모든 단어의 길이는 같습니다.
- `words`에는 3개 이상 50개 이하의 단어가 있으며 중복되는 단어는 없습니다.
- `begin`과 `target`은 같지 않습니다.
- 변환할 수 없는 경우에는 0를 return 합니다.

입출력 예

```
7
hit hot dot dog lot log cog
4
```

```
6
hit hot dot dog lot kkk
0
```

입출력 예 설명

예제 #1

문제에 나온 예와 같습니다.

예제 #2

`target`인 "cog"는 `words` 안에 없기 때문에 변환할 수 없습니다.

문제 5

아이
디어

1. 숫자 한자리 바꿔서 Target 숫자로 가는 과정에 대한 문자열 버전
2. 중간값들을 활용하여 Target으로 갈 수 있는 모든 경우를 체크해야 하므로 Queue를 사용
3. 주어진 단어를 사용해야 하므로 char 하나씩 위치를 확인하여 1 char 변환으로 이동이 가능한지 체크하는 함수를 만듦
4. 3을 통해 만들어진 단어는 queue에 다시 넣어 주고 단계를 1씩 증가
5. queue에서 하나씩 빼서 위의 과정을 수행하다 target과 일치하는 단어가 나오면 해당 단계를 출력

답안

<https://onlinegdb.com/ryZhfbWaN>

문제 6

<https://www.acmicpc.net/problem/2667>

문제

<그림 1>과 같이 정사각형 모양의 지도가 있다. 1은 집이 있는 곳을, 0은 집이 없는 곳을 나타낸다. 철수는 이 지도를 가지고 연결된 집들의 모임인 단지를 정의하고, 단지에 번호를 붙이려 한다. 여기서 연결되었다는 것은 어떤 집이 좌우, 혹은 아래위로 다른 집이 있는 경우를 말한다. 대각선상에 집이 있는 경우는 연결된 것이 아니다. <그림 2>는 <그림 1>을 단지별로 번호를 붙인 것이다. 지도를 입력하여 단지수를 출력하고, 각 단지에 속하는 집의 수를 오름차순으로 정렬하여 출력하는 프로그램을 작성하시오.

0	1	1	0	1	0	0
0	1	1	0	1	0	1
1	1	1	0	1	0	1
0	0	0	0	1	1	1
0	1	0	0	0	0	0
0	1	1	1	1	1	0
0	1	1	1	0	0	0

<그림 1>

0	1	1	0	2	0	0
0	1	1	0	2	0	2
1	1	1	0	2	0	2
0	0	0	0	2	2	2
0	3	0	0	0	0	0
0	3	3	3	3	3	0
0	3	3	3	0	0	0

<그림 2>

입력

첫 번째 줄에는 지도의 크기 N (정사각형이므로 가로와 세로의 크기는 같으며 $5 \leq N \leq 25$)이 입력되고, 그 다음 N 줄에는 각각 N 개의 자료(0혹은 1)가 입력된다.

문제 6

출력

첫 번째 줄에는 총 단지수를 출력하시오. 그리고 각 단지내 집의 수를 오름차순으로 정렬하여 한 줄에 하나씩 출력하시오.

예제 입력 1 [복사](#)

```
7
0110100
0110101
1110101
0000111
0100000
0111110
0111000
```

예제 출력 1 [복사](#)

```
3
7
8
9
```

문제 6

접근법

아이디어

1. 지도에서 하나하나 조사하면서 집을 발견하면 단지로 정의하고 집의 수를 세는 방법으로 풀이 (완전탐색)
2. 단지 안에서 집의 수를 세는 건 주변으로 뻗어 나가는 방법을 사용하기 때문에 BFS나 DFS를 통해서 4방향으로 탐색을 진행
3. DFS(재귀함수, 스택), BFS(큐) 모두 풀어보자

풀이법

1. 전체의 지도를 돌면서 집이 있는지 확인하다.
2. 집(배열의 값 == 1)을 발견하면 단지로 정의하고 ID를 부여한다.
3. BFS 또는 DFS 탐색을 통해서 해당 단지의 집의 수를 구한다.
4. 해당 단지의 집의 수를 다 구하게 되면 해당 집의 ID를 인덱스로 하는 배열에 추가해준다.
5. 전체 지도의 탐색이 다 끝나면 배열을 오름차순으로 정렬하고 출력한다

답안

<https://onlinegdb.com/SJWtsVY98>