

이진 변환 반복하기

나의 풀이

```
using System;
using System.Linq;

public class Solution {

    public int[] solution(string s) {
        int numberOfCycles = 0;
        int toRemove = 0;

        while ( !s.Equals("1") ) {
            int numberOfZeros = s.Count(x => x == '0');
            toRemove += numberOfZeros;
            s = Convert.ToString(s.Length - numberOfZeros, 2);
            numberOfCycles++;
        }

        return new int[] { numberOfCycles, toRemove };
    }
}
```

문제 설명

0과 1로 이루어진 어떤 문자열 x 에 대한 이진 변환을 다음과 같이 정의합니다.

1. x 의 모든 0을 제거합니다.
2. x 의 길이를 c 라고 하면, x 를 " c 를 2진법으로 표현한 문자열"로 바꿉니다.

예를 들어, $x = "01111010"$ 이라면, x 에 이진 변환을 가하면 $x = "01111010" \rightarrow "1111" \rightarrow "100"$ 이 됩니다.

0과 1로 이루어진 문자열 s 가 매개변수로 주어집니다. s 가 "1"이 될 때까지 계속해서 s 에 이진 변환을 가했을 때, 이진 변환의 횟수와 변환 과정에서 제거된 모든 0의 개수를 각각 배열에 담아 return 하도록 solution 함수를 완성해주세요.

제한사항

- s 의 길이는 1 이상 150,000 이하입니다.
- s 에는 '1'이 최소 하나 이상 포함되어 있습니다.

입출력 예

s	result
"110010101001"	[3, 8]
"01110"	[3, 3]
"1111111"	[4, 1]

입출력 예 설명

입출력 예 #1

- "110010101001"이 "1"이 될 때까지 이진 변환을 가하는 과정은 다음과 같습니다.

회차	이진 변환 이전	제거할 0의 개수	0 제거 후 길이	이진 변환 결과
1	"110010101001"	6	6	"110"
2	"110"	1	2	"10"
3	"10"	1	1	"1"

- 3번의 이진 변환을 하는 동안 8개의 0을 제거했으므로, [3, 8] 을 return 해야 합니다.

입출력 예 #2

- "01110"이 "1"이 될 때까지 이진 변환을 가하는 과정은 다음과 같습니다.

회차	이진 변환 이전	제거할 0의 개수	0 제거 후 길이	이진 변환 결과
1	"01110"	2	3	"11"
2	"11"	0	2	"10"
3	"10"	1	1	"1"

- 3번의 이진 변환을 하는 동안 3개의 0을 제거했으므로, [3, 3] 을 return 해야 합니다.

입출력 예 #3

- "1111111"이 "1"이 될 때까지 이진 변환을 가하는 과정은 다음과 같습니다.

회차	이진 변환 이전	제거할 0의 개수	0 제거 후 길이	이진 변환 결과
1	"1111111"	0	7	"111"
2	"111"	0	3	"11"
3	"11"	0	2	"10"
4	"10"	1	1	"1"

- 4번의 이진 변환을 하는 동안 1개의 0을 제거했으므로, `[4, 1]` 을 return 해야 합니다.