

괄호 나누기

괄호 문자열이란 (와)로만 이루어진 문자열을 말한다.

괄호 문자열 중, 다음 규칙만을 이용해서 만들 수 있는 문자열을 **좋은 괄호 문자열**이라고 한다.

- 빈 문자열은 좋은 괄호 문자열이다.
- X 가 좋은 괄호 문자열이면 (X) 도 좋은 괄호 문자열이다.
- X 와 Y 가 좋은 괄호 문자열이면 XY 도 좋은 괄호 문자열이다.

이 규칙들로 만들 수 없는 문자열은 좋은 괄호 문자열이 아니다.

키키에게 길이가 N 인 좋은 괄호 문자열 S 가 주어진다. 키키는 S 를 $K - 1$ 번 잘라서, 비어 있지 않은 연속한 조각 K 개 $P_1, P_2, \dots, P_K$ 로 나누어야 한다.

키키가 자르고 나면, 모그는 이 K 개의 조각을 원하는 순서로 늘어놓아 문자열 T 를 만든다. 그런 다음 모그는 T 에서 인접한 괄호 쌍 ()가 더 이상 남지 않을 때까지 이를 반복해서 지운다. **점수**는 다 지우고 나서 남은 문자열의 길이이다.

어떤 분할이 주어졌을 때, 모그가 조각을 재배열해서 얻을 수 있는 점수의 최댓값을 그 분할의 **값**이라고 한다. 키키는 값이 최대한 작아지는 분할을 찾고 싶어 한다.

이렇게 얻을 수 있는 가장 작은 값을 X 라고 하자. X 와, 값이 X 인 S 의 분할을 하나 출력하라.

입력

첫째 줄에 문자열의 길이 N 과 조각의 개수 K 가 주어진다.

둘째 줄에 문자열 S 가 주어진다.

출력

첫째 줄에 분할의 값 중 최솟값 X 를 출력한다.

다음 K 개의 줄에 걸쳐 $P_1, P_2, \dots, P_K$ 를 원래 순서대로 한 줄에 하나씩 출력한다.

각 조각은 비어 있지 않아야 하며, 조각들을 순서대로 이어 붙이면 S 가 되어야 한다. 또한 출력한 분할의 값은 정확히 X 여야 한다.

최적의 분할이 여러 가지라면 그중 아무것이나 출력해도 된다.

제약 조건

- $2 \leq N \leq 300$.
- N 은 짝수이다.
- $1 \leq K \leq N$.
- S 의 길이는 N 이며, (와)로만 이루어져 있다.
- S 는 좋은 괄호 문자열이다.

서브태스크와 채점

서브태스크	점수	추가 제약 조건
1	6	$N \leq 10$
2	2	$K = 1$
3	7	$K \leq 3$
4	15	S 의 앞 $N/2$ 개 문자는 모두 (이고, 뒤 $N/2$ 개 문자는 모두)이다.
5	33	$N \leq 70$
6	37	추가 제약 조건이 없다.

각 서브태스크에서는, 분할이 틀리더라도 값 X 만 올바르게 출력하면 해당 서브태스크 점수의 50%를 받을 수 있다.

더 정확히 말하면, 각 테스트 케이스에서 출력은 다음과 같이 채점된다.

- X 가 올바르고, 이어지는 K 개 줄이 값이 X 인 올바른 분할을 나타내면 만점을 받는다.
- X 는 올바르지만 이어지는 K 개 줄이 그러한 분할을 나타내지 않으면 점수의 50%를 받는다.
- 그 외의 경우에는 0점을 받는다.

각 서브태스크의 점수는 그 서브태스크에 속한 테스트 케이스들에서 받은 점수 중 최솟값이다.

예제

예제 입력 1

```
8 4
( ( ( ( ) ) ) )
```

예제 출력 1

```
2
(
()
(())
)
```

예제 출력의 마지막 네 줄은 S 를 $(, (), (()),)$ 네 조각으로 나눈 것이다.

모그는 이 조각들을 $), (), (()), ($ 순서로 늘어놓을 수 있다. 이때 가능한 괄호 쌍을 모두 지우고 나면 $)$ 가 남으므로 점수는 2이다. 이 조각들로는 모그가 2보다 큰 점수를 얻을 수 없고, 네 조각으로 나누는 어떤 분할도 값이 2보다 작을 수는 없다.

예제 입력 2

```
4 2
()()
```

예제 출력 2

```
0
()
()
```

예제 입력 3

```
12 6
((((((()))))))
```

예제 출력 3

```
6
(
(
(
((()))
)
))
```