# COCI '21 Contest 6 #5 Superozicija

**Time limit:** 1.0s **Memory limit:** 512M

World-renowned physicist Juraj has recently discovered a new kind of elementary particle – a parenthesision. A parenthesision can have either an open `(` or `)` closed configuration. Using his homemade particle accelerator, Juraj has created $t$ sequences of superpositions of $n$ parenthesisions. In each of the $t$ sequences, there are $n$ parenthesisions in a superposition between two different positions and (not necessarily different) configurations. If the sequence is observed, the wave function of parenthesisions collapses and each of them ends up in one of its possible positions and configurations. Juraj wants to know if it is possible that the parenthesisions collapse into a valid sequence of parenthesis?

Juraj M. PhD knows that the quantum physics of these revolutionary and completely scientifically founded particles are way over the head of an average COCI contestant, so he provided a formal task statement:

You are given $t$ sequences of $2n$ open and closed parenthesis. Each parenthesis is a member of exactly one pair of parentheses. Two parentheses within a pair can be different, both open, or both closed. Juraj wants to know if it is possible to choose a single parenthesis from each of the pairs such that the chosen parentheses form a valid sequence of parentheses. Furthermore, if this is possible he asks you to print which parentheses he should choose to get a valid sequence. A sequence of parentheses is valid if it is empty or it can be written as $(A)$ or $AB$ where $A$ and $B$ are arbitrary valid sequences of parentheses.

## Input Specification

The first line contains an integer $t$ $(1 \leq t \leq 100\,000)$, number of parentheses sequences. $t$ sequence descriptions follow.

The first line of sequence description contains an integer $n$ $(1 \leq n \leq 100\,000)$, number of pairs of parentheses in the sequence.

The second line contains $z$, a string of length $2n$. $z$ contains exclusively characters `(` and `)`.

The following $n$ lines of sequence description contain two integers $a_i$ and $b_i$ $(1 \leq a_i < b_i \leq 2n)$. Each of the numbers $1, 2, \ldots, 2n$ appears exactly once.

Sum of all $n$ will be less than or equal to $100\,000$.

## Output Specification

In the $i^{\text{th}}$ of $t$ lines, print a sequence of zeros and ones representing a possible choice of parentheses. If parenthesis at index $a_j$ of the $j^{\text{th}}$ pair of $i^{\text{th}}$ sequence is chosen, print `0`; otherwise, if parenthesis at index $b_j$ is chosen, print `1`. If there is no valid sequence of parentheses, print `-1`.

## Constraints

| Subtask | Points | Constraints |
|---|---|---|
| 1 | 10 | $1 \le n \le 10$ |
| 2 | 10 | $z[a_i] = z[b_i]$ for all $i = 1, 2, \ldots, n$. |
| 3 | 20 | $b_i = a_i + 1$ for all $i = 1, 2, \ldots, n$. |
| 4 | 70 | No additional constraints. |

## Sample Input 1

```
1
4
()))((()
1 2
3 5
4 6
7 8
```

## Sample Output 1

```
0 1 0 1
```

## Explanation for Sample Output 1

From the original sequence `()))((()`, only the bolded parentheses will remain **()))(**(**()**. That is, `()()`, which is a valid sequence of parentheses.

## Sample Input 2

```
1
4
)()()()(
1 2
3 4
5 6
7 8
```

## Sample Output 2

```
1 1 0 0
```

## Explanation for Sample Output 2

From the original sequence `)()()()(`, only the bolded parentheses will remain )**()()()**(. That is, `(())`, which is a valid sequence of parentheses.

## Sample Input 3

```
1
3
(()())
1 6
2 4
3 5
```

## Sample Output 3

```
-1
```