

E. 题目名以 OJ 为准 / Conditionop

时间限制：1.0 秒

空间限制：512 MiB

【题目背景】

三目运算符表达式 $a?b:c$ 的含义是，如果 a 为真，那么返回 b ，否则返回 c 。

三目运算符是右结合的： $a?b:c?d:e$ 和 $a?b:(c?d:e)$ 等价。如果你不记得运算顺序，可以总是使用括号。

0 为假，1 为真。

【题目描述】

给定一个长为 $2n+1$ 的 01 串，你需要使用 n 次三目运算符，即在中间插入恰好 n 个 $?$ 和 n 个 $:$ 以及若干括号，使得表达式的结果为 1，或判断无解。

【输入格式】

从标准输入读入数据。

第一行一个正整数 n ($1 \leq n \leq 1.5 \times 10^5$)。

第二行一个长为 $2n+1$ 的 01 串，表示给定的字符串。

【输出格式】

输出到标准输出。

如果无解，输出一行 No。

如果有解，第一行输出 Yes，第二行输出一个值为 1 的表达式。你可以使用括号，但是需要保证你的表达式中数字的顺序和原串相同。你需要保证你输出的表达式长度不超过 $10n+1000$ 。可以证明如果存在解，则一定存在满足条件的构造方案。

【样例 1 输入】

```
1 2
2 10101
```

【样例 1 输出】

```
1 Yes
2 (1?0:1)?0:1
```

【样例 1 解释】

你如果输出 $((1?0:(((1))))?0:1)$ 等表达式也算正确。

【样例 2 输入】

```
1 2
2 00000
```

【样例 2 输出】

```
1 No
```

【提示】

你可以直接使用 `g++` 编译你的表达式来检查表达式的值，但是这种方法并不能检测数字的顺序是否一致，也不能检测你使用三目运算符的次数是否恰好为 n ，即是否每两个相邻的数字之间都有一个 `?` 或 `::`。

```
1 #include <cassert>
2 #define YOUR_EXPRESSION <your_expression>
3 int main(){
4     assert(YOUR_EXPRESSION);
5     return 0;
6 }
```