Problem 1　反例 (簡化版)

考慮以下的問題： 給定長度為 `n` 的整數序列 $A = \{a_1, a_2, \ldots, a_n\}$，計算

$$\max_{1 \le \ell \le r \le n} \sum_{\ell \le i \le r} (r - \ell + 1) \cdot a_i,$$

其中 $1 \le n \le 2000$，並且 $|a_i| \le 10^6$ 。

在嘗試解這個問題的過程裡，小美回想起課堂上學過的 Maximum Sum Segment 問題，並想出了一個 $O(n)$ 線性時間複雜度的方法，其虛擬碼如下：

```
procedure FIND_ANSWER(A, n)
    # Assumes n is an integer between 1 and 2000, inclusive
    # Assumes A is an array with n integers:  a[0], a[1], ..., a[n-1]
    res = 0
    cur = 0
    k = −1
    for i = 0 to n-1  do
        cur = cur + A[i]
        if  cur < 0  then
            cur = 0
            k = i
        end if
        res = max( res, (i-k)*cur )
    end for
    Return res
end procedure
```

如你所見，小美的方法並不完全正確。舉例來說，若 `n=4`，`A={ 6, -8, 7, -42 }`，那麼 `find_answer(A,n)` 會傳回 `7`，但實際上最佳解答為 $3 \cdot (6 - 8 + 7) = 15$。

你試著告訴小美，她的方法並不正確，但她不相信。

『你說的是特例吧？ 我只要在這邊加一個 `if` 判斷式就會對了。』 小美說。

『噢噢，這邊也要加一個 `if` 。』

眼見小美開始寫下一個又一個 `if` 判斷式，你決定寫一個程式來幫助小美回歸正途。

Problem 1 continued on next page. . .

給定一個正整數 k, 其中 $1 \leq k \leq 10^9$, 你的任務是產生一個長度為 `n` 的整數序例 `A` (作為小美的方法的反例), 使得小美的 `find_answer(A,n)` 函式計算的結果 與 最佳解之間的差洽好等於 k。

舉例來說，若給定 $k = 8$，那麼上頁的序列 `A={ 6, -8, 7, -42 }` 是一個所求的反例，因為小美的程式會傳回 7，而實際上的最佳解為 15，兩者相差 8。

注意：你可以自由選擇序列的長度以及序列的內容，但序列的長度 `n` 必須符合原始題目的要求，亦即， `n` 的值必須符合 $1 \leq n \leq 2000$。 **小美堅信不存在反例，她說，即使把元素絕對值的限制放寬到 10^{13} 也沒關係**。

輸入格式

輸入包含一個正整數 k, $1 \leq k \leq 10^9$。

輸出格式

若不存在題目要求的反例序列，就印出 "Yes, you are right."。

否則的話，在第一行印出你設計的序列的長度 n；
接著，在第二個印出你設計的序列的內容，n 個整數，以空白格開。

注意：輸出的 n 必須符合 $1 \leq n \leq 2000$、且每個整數的絕對值都不超過 10^{13}。

範例一

Input

```
8
```

Output

```
4
6 -8 7 -42
```

範例二

Input

```
612
```

Output

```
7
30 -12 -99 123 -2 245 -300
```