(Adv.) Competitive Programming

Submit until end of contest, via the judge interface

Problem: gifts2 (1 second timelimit)

Yet again, little Tommy's mom Tammy got him a nice and long string for his birthday. She even bribed her schoolyard snitch, Stan, to tell her about these cool *ternary strings* every child is obsessed with. Apparently they are the new palindromes! Using only "0", "1", and "2" as characters, they are slick and enjoyable compared to their old-school alphanumeric predecessors. The day before the party, however, she hears Tommy talk about *balanced* ternary strings. Stan must have known that the Carmichael number he got for the intel was forged. After all, a real connoisseur recognizes a violated Korselt's criterion a mile off. So Stan left out the most important detail about the ternary string trend: each character should occur equally often! Hastily Tammy proceeds to balance the string with her professional sewing kit but is quickly overwhelmed by the sheer number of possibilities. Because altering a character to a different one is a time-consuming operation, she is faced with a difficult optimization problem. Reflecting on the empty bottle of wine on the counter, she decides she needs some emergency IT support. That is where you come into play. Since Tammy somewhat underspecified your task, I guess you couldn't go wrong with lexicographically smallest. Right?

Input The input contains a ternary string of length at most $3 \cdot 10^5$. Tammy already cut it to shape, meaning it is divisible by 3.

Output Print the lexicographically smallest balanced ternary string that can be obtained from the given one with the minimum number of replacements.

Sample input	**Sample output**
`121`	`021`
`000000`	`001122`
`211200`	`211200`

```
120110
```

```
120120
```

```
222222200000
```

```
111222200001
```