***Part 1***

Something is jamming your communications with Santa. Fortunately, your signal is only partially jammed, and protocol in situations like this is to switch to a simple repletion code to get the message through.

In this model, the same message is sent repeatedly. You've recorded the repeating message signal (your puzzle input), but the data seems quite corrupted - almost too badly to recover. Almost.

All you need to do is figure out which character is most frequent for each position. For example, suppose you had recorded the following messages:

eedadn

drvtee

eandsr

raavrd

atevrs

tsrnev

sdttsa

rasrtv

nssdts

ntnada

svetve

tesnvt

vntsnd

vrdear

dvrsen

enarar

The most common character in the first column is **e**; in the second, a; in the third, **s**, and so on. Combining these characters returns the error-corrected message, easter.

Given the recording in your puzzle input, what is the error-corrected version of the message being sent?

*Part 2*

Of course, that would be the message - if you hadn't agreed to use a modified repetition code instead.

In this modified code, the sender instead transmits what looks like random data, but for each character, the character they actually want to send is slightly less likely than the others. Even after signal-jamming noise, you can look at the letter distributions in each column and choose the least common letter to reconstruct the original message.

In the above example, the least common character in the first column is **a**; in the second, **d**, and so on. Repeating this process for the remaining characters produces the original message, **advent.**

Given the recording in your puzzle input and this new decoding methodology, what is the original message that Santa is trying to send?