The code is written and is used to solve the **anagram string problem**. It starts by reading the input and then processes each test case. If the string has an odd length, it immediately returns -1 because it cannot be evenly divided into two halves. If the length is even, the string is split into two parts. The program counts the characters in the second half and then compares them with the first half. For every character in the first half that doesn’t match, it counts one change. In the end, it returns the minimum number of changes needed to make the two halves anagrams of each other.