Given a target String, you are to determine whether target String can be formed by combining some or all of the letters from two other Strings.

The Zipper class has a single constructor with a single String parameter representing the target String. In this problem you will be completing the canBuild and the canZipper methods in the Zipper class.

The canBuild method has the two String parameters to be used to construct the target String. The order of the letters in the two String parameters is NOT important. That is, the order of letters may be rearranged in attempting to construct the target String.

You may assume all letters (in all three Strings) will be lower case letters with NO special characters.

You should also assume that the length of the target String is greater than zero and the length of the two String parameters is great than or equal to 0.

The canBuild(String a, String b) method returns true if the target String can be constructed from the letters (in any order) String a and String b and returns false otherwise.

The following code shows the results of the canBuild method.

|  |  |
| --- | --- |
| The following code | Returns |
| Zipper zip = new Zipper("tcraete"); |  |
| zip.canBuild("cat", "tree") | true |
| zip.canBuild("tcae", "etre") | true |
| zip.canBuild("cat", "ret") | false |
| zip.canBuild("teak", "crochet") | true |

The canZipper method is described on the next page.

The canZipper method has the two String parameters to be used to construct the target String. The order of the letters in the two String parameters is important. That is, the order of letters may **NOT** be rearranged in attempting to construct the target String.

You may assume all letters will be either a lower case letters **or** a space.

You should also assume that the length of the Target String is greater than zero and the length of the two String parameters is great than or equal to 0.

The canZipper(String a, String b) method returns true if the target String can be constructed from the letters contained in String a and String b **without** changing the order of the letters in both String a and String b and returns false otherwise.

The following code shows the results of the canZipper method.

|  |  |
| --- | --- |
| The following code | Returns |
| Zipper zip1 = new Zipper("tcraete");  zip1.canZipper("cat", "tree") | true |
| Zipper zip2 = new Zipper("catrtee");  zip2.canZipper("cat", "tree") | true |
| Zipper zip3 = new Zipper("cttaree");  zip3.canZipper("cat", "ret") | false |
| Zipper zip4 = new Zipper("jejunator");  Zip4.canZipper("dejeuner", "jubilation") | true |
| Zipper zip5 = new Zipper("to or too");  zip5.canZipper("tor ", "ot oo")  zip5.canZipper("to r", "o too") | false  true |