Greek Letter Club

This program is an application of random numbers.

You and your friends want to form a social club on campus. And to give your organization a collegiate cachet, you have decided to create a name for your group that contains Greek letters. A good sounding name would have exactly three Greek letters, and none of the letters should repeat. In other words, three different letters. Sometimes, the name of a Greek letter organization is the abbreviation of some Greek phrase. But you don't know Greek. So, the easiest thing to do is …. Pick three random Greek letters!

Write a program that accomplishes the following:

1. Have your program generate a single Greek letter. For example, you could write a function `newLetter` that will return a string containing the name of some random Greek letter. Inside this function, you should call the built-in random number generator to select a random integer in the range 0..23. Then, use this number to index into the Greek alphabet, which you could have stored as an array of 24 strings. ["Alpha", "Beta" …, "Omega"].

2. In your main program, write code that will call `newLetter` three times. However, you need to ensure that on the second and third calls, the new Greek letter being returned does not duplicate an existing letter. There is more than one way to tackle this problem! For example, you could delete a letter from the Greek alphabet once it has been selected, or you could enclose your calls to `newLetter` inside do-while loops that terminate once a unique letter is selected.

3. At this point, your program should be able to suggest a name for the group that contains 3 Greek letters. However, you could be unlucky. The name might already exist for some other fraternity or sorority, for example. Use an input file called existing.txt that contains a list of some Greek letter organizations whose names already exist and that you don't want to duplicate. Write a function that takes a candidate name for a Greek letter organization, and returns true if the name already exists in the existing.txt list, and false if it is not on the list. Don't assume that the entries in existing.txt are in alphabetical order already.

    Before you continue, let's do some checking of your code. In your main program, be sure to test your duplicate checker to make sure it works. Test it on names of Greek organizations you already know to see that it returns true. Give it a fictitious name as well to see that it can return false.

4. In your main program, print out 5 possible names for the group.

5. Write a function that reads existing.txt and counts how many times each Greek letter is used. Report these results to the user. Which letter is least common?