File I/O

1-)   The text files boynames.txt and girlnames.txt, which are included in the Sakai, contain a list of the 1,000 most popular boy and girl names in the United States for the year 2003 as compiled by the Social Security Administration.
        These are blank-delimited files, where the most popular name is listed first, the second most popular name is listed second, and so on, to the 1,000th most popular name, which is listed last. Each line consists of the first name followed by a blank space and then the number of registered births using that name in the year. For example, the girlnames.txt file begins with
        Emily 25494
        Emma 22532
        Madison 19986
        This indicates that Emily was the most popular name with 25,494 registered namings, Emma was the second most popular with 22,532, and Madison was the third most popular with 19,986.
        Write a program that reads both the girl and boy files into memory using arrays. Then, allow the user to input a name. The program should search through both arrays. If there is a match, then it should output the popularity ranking and the number of namings. The program should also indicate if there is no match.
        For example, if the user enters the name "Justice," then the program should output
                Justice is ranked 456 in popularity among girls with 655 namings.
                Justice is ranked 401 in popularity among boys with 653 namings.
         If the user enters the name "Walter," then the program should output
                Walter is not ranked among the top 1000 girl names.
                Walter is ranked 356 in popularity among boys with 775 namings.

2-)   Write a program that will count the total occurrences of the number '10' in a text file of strings representing numbers of type int and will show the value of the count on the screen once the whole file is read. The file contains the following numbers separated by space.
                10 4 7 8 10 34 11 10 15 6 10

3-)   Write a program that takes its input from a text file of strings representing numbers of type double and outputs the average of the numbers in the file to the screen. The file contains nothing but strings representing numbers of type double, one per line.

4-)   Write a program that gives and takes advice on program writing. The program starts by writing a piece of advice to the screen and asking the user to type in a different piece of advice. The program then ends. The next person to run the program receives the advice given by the person who last ran the program. The advice is kept in a text file and the content of the file changes after each run of the program. You can use your editor to enter the initial piece of advice in the file so that the first person who runs the program receives some advice. Allow the user to type in advice of any length so that it can be any number of lines long. The user is told to end his or her advice by pressing the Return key two times. Your program can then test to see that it has reached the end of the input by checking to see when it reads two consecutive occurrences of the character '\n'.

5-)   Create a file WordBuff.txt that contains the following list of words: MADAM, DAD, RISK, JAVA, MALAYALAM, RACECAR, RADAR, ROTOR, REFER, SEDES, SOLOS, COURSE, STATS, TOROT, TENET, MACHINE, VIRTUAL, STUDENT, PULLUP, PROGRAMME, and CORE. Write a program that reads each word from the file and outputs the number of palindromes in the file.

TO PRACTICE MORE, YOU CAN TRY TO SOLVE PREVIOUS ASSIGNMENTS WITH FILE I/O.