## Lab work #4

**Topics**

* Text files
* Containers

**Exercises**

1. Implement a program to copy a text file to another one. Both files should be located in the folder where you execute the program and their names should be asked to the user.
2. Implement a program to calculate the sum, average, maximum and minimum of all the real numbers stored in a text file. Consider that the file is located in the folder where you execute the program. The name of the file should be asked to the user. The file should be in the following format (one number per line):

| nums.txt |
| --- |
| 12.39  1.93  7.85  ... |

1. The file /etc/dictionaries-common/words contains a list of words (in english), one per line. Some of those words have equal consecutive letters, such as “accent", "access", "pool", etc. Develop a program to list and count all those words. Suggestion: start by creating the function int checkDoubles(const string &w), which returns the number of letter pairs found in string w and test it conveniently.
2. File school1.csv contains the grades of students in a class. Each row has the record of one student and each column has an information field. The columns are separated by TAB characters. The first row contains a header with the titles of the fields. The files school2.csv and school3.csv have the grades of students from other classes in the same format.
   1. Write a program that asks for the filenames to be processed and save them into an array of strings. Then the program should load the files and print (on the screen) the global grade sheet (all classes), with the 3 columns: the student number, the student name and its final grade, formatted and aligned as in the example below.



* 1. Change your program in order to be able to write the grade sheet into an output file.

1. Implement a program that allows the user to input integer numbers and stores them into a vector. The input should finish with the introduction of zero. Display the stored numbers in reverse order and calculate their average.
2. Implement a program that simplifies management of several teams of indoor soccer (5 players). For each team, the user has to input the numbers of 5 players (all different - explore the use of the set container). After a team has been formed, the program should ask whether the user wants to add another team (the response can be 'y' or 'n'). Suggestion: consider the use of a vector of sets.
3. Implement a program to count the total number of words stored in a text file as well as the number of occurrences of a specific keyword. You should only consider exact matches, i.e. if the keyword is "contained" inside another word it should not be counted. The name of the text file and the keyword are provided by the user.

Test your program with the file "Os Lusíadas" (document 3333 from the Gutenberg project - <http://www.gutenberg.org/ebooks/3333>).