Exercise: Files and Directories

This document defines the **exercises** for the "Java Advanced" course @ Software University. Please submit your solutions (source code) of all below described problems in Judge.

Problem 1. Odd Lines

Write a program that reads a text file and writes its every **odd** line in another file. Line numbers starts from 0.

Examples

Input.txt	Output.txt
Two households, both alike in dignity, In fair Verona, where we lay our scene, From ancient grudge break to new mutiny, Where civil blood makes civil hands unclean. From forth the fatal loins of these two foes A pair of star-cross'd lovers take their life; Whose misadventured piteous overthrows Do with their death bury their parents' strife.	In fair Verona, where we lay our scene, Where civil blood makes civil hands unclean. A pair of star-cross'd lovers take their life; Do with their death bury their parents' strife

Read **01_OddLinesIn.txt** and test your output against **01_OddLinesOut.txt**, then read **02_OddLinesIn.txt** and test against its respective output file and so on.

Problem 2. Line Numbers

Write a program that reads a text file and inserts line numbers in front of each of its lines. The result should be written to another text file.

Examples

Input.txt	Output.txt
Two households, both alike in dignity, In fair Verona, where we lay our scene, From ancient grudge break to new mutiny, Where civil blood makes civil hands unclean. From forth the fatal loins of these two foes A pair of star-cross'd lovers take their life; Whose misadventured piteous overthrows Do with their death bury their parents' strife.	1. Two households, both alike in dignity, 2. In fair Verona, where we lay our scene, 3. From ancient grudge break to new mutiny, 4. Where civil blood makes civil hands unclean. 5. From forth the fatal loins of these two foes 6. A pair of star-cross'd lovers take their life; 7. Whose misadventured piteous overthrows 8. Do with their death bury their parents' strife.

Read from **01_LinesIn.txt** and compare your output against **01_LinesOut.txt**, then read **02_LinesIn.txt** and compare to its respctive output file and so on.

Problem 3. Word Count

Write a program that reads a list of words from the file `words.txt` and finds how many times each of the words is contained in another file **text.txt**. Matching should be **case-insensitive**.

Write the results in file **results.txt**. Sort the words by frequency in descending order.

Examples

words.txt	Input.txt	Output.txt
quick is fault	-I was quick to judge him, but it wasn't his fault. -Is this some kind of joke?! Is it? -Quick, hide here…It is safer.	is - 3 quick - 2 fault - 1

First read the **words1.txt** in order to retrieve the words you will be looking for, then read **text1.txt** to retrieve the text in which you will be searching for the words. Finally test your output against **01_WordCount.txt**. Then read **words2.txt** and so on

Problem 4. Merging two files into third one

Write a program that reads the contents of two text files and merges them together into a third one.

Examples

Input1.txt	Input2.txt	Output.txt
1 3 5	2 4 6	1 2 3 4 5 6

First read **01_FileOne.txt** and **01_FileTwo.txt**, merge them and compare your output against **01_Merged**. Then read **02_FileOne.txt** and **02_FileTwo.txt** and compare it to **02_Merged**.

Problem 5. Get folder size

You are given a folder named TestFolder. Get the size of all files in the folder, which are **NOT directories.** Print the result on the console in **Megabytes**.

Examples

Output
5.161738395690918

Problem 6. Timer

Write a program that determines how long does it take to read a file. You are given two files – "File1" and "File2". Your task is to create a program, which determines the duration of reading them. On the output – print **true if the first** file is taking **less** time or **print false if the second** one does.

Hint: Use - System.nanoTime() to measure time duration

Problem 7. Read from specified line

Write a program that reads from the console an integer, which specifies the starting line from which you should start reading from. Print on the console the remaining text.