Write a program which:

- Receives, as command line arguments, the name of **two files** containing:
  - **First file**: a list of strings, only composed by alfanumeric characters [a-zA-Z0-9], one per line, no blanks, max length of line: 80 chars.
  - **Second file**: a list of paths (filenames), one per line, max length of line: 80 chars.
- Reads and parse the two files.
- Looks for each string, found in the first file, in all the pathnames (files) indicated in the second file.
- Outputs the statistics of matches that have been found, by printing:
- For each string in the first file, the name of each file containing the match and the number of occurrences.
- *Optional: for each file, the strings found and the number of matches.*

Steps (just an idea)

- A main which receives on the command line the name of two files and checks for possible errors.
- Function to read from a file (first arg), the next alfanumeric string, returning EOF at the end of file, and managing possible error condizion.
  - *Optional: remove the 80 chars limitation in the length of input string.*
- Function to look up for a string (first arg) into a file (second arg), returning the number of matches.
  - *Optional: write a function which receives an array of strings (instead of a single string) and returns all matches.*
- A set of functions to manage a list of *struct*, containing the pointers to strings and the occurrences. E.g:
  - Create and initialize the list
  - Insert (and update) a new item
  - Visit the list
  - Dispose the list (and the other malloc()'ed data structures)

Example:

$ myProg File1 File2

- File1:

a
bbc
cc

- File2:

pippo.txt
pluto.txt

- pippo.txt

dddd
e
aa
bbc

- pluto.txt

a
aa
a

- Output

a: pluto.txt(2)
Bbc: pippo.txt(1)
cc: