Background:

RLE compression relies on the fact that in many files there are many repeated characters (example: AAAAAA) .. The algorithm replaces this with a character+count resulting in a much smaller size.

For example: a string in a file contains: "AAAAADDDDDDDDEEEQQQQQQQQQQAAEE" is replaced with: "A5D8E3Q10A2E2"

There are two operations:

A) COMPRESS: read and compress the file into a 2nd file

B) DECOMPRESS: read a compressed file and reconstruct the original file.

Part1: compression

1) Write a Compress function that receives a string as parameter and return compressed string. The function will :
    a. Read 1st character from string parameter
    b. Count how many times it is repeated
    c. Add this info to a new string
    d. Read next character from source string,.
    e. Repeat from step b above, till end of string
    f. When done the function will return the new string

2) Test your function using sample string above.

Part2: file processing

1) Ask user for input file name and open it (this is the text file containing data/art to be compressed)
2) Open an output file to be the compressed file (add extension .comp to the name)
3) Read a line from input file (getline into a string)
4) Send the string to the compress function.
5) When the compress function returns the compressed string, write it to the output file
6) Read next line from input file, goto #4
7) Repeat till end of file

Part2: decompress

1) Write a function decompress that takes a string and return a string
2) Stringstream 1 char and 1 number
3) Create a string that is filled with x-number of this char
4) Add this string to the decompress string
5) Repeat till end of source string
6) Return the decompressed string

7) Use the above function and open the .comp file, read line by line from it, decompress each line, and write it to a .decomp file

RESULT:

You will have a new .comp file that is a compressed version of the original file AND a .decomp file that is the decompression of the decomp file.

Your .decomp file should be the same as the original file.