Data and File Structures Laboratory

Programming in C – Warmup

Malay Bhattacharyya

Assistant Professor

Machine Intelligence Unit
Indian Statistical Institute, Kolkata

July, 2019

Problems – Day 1

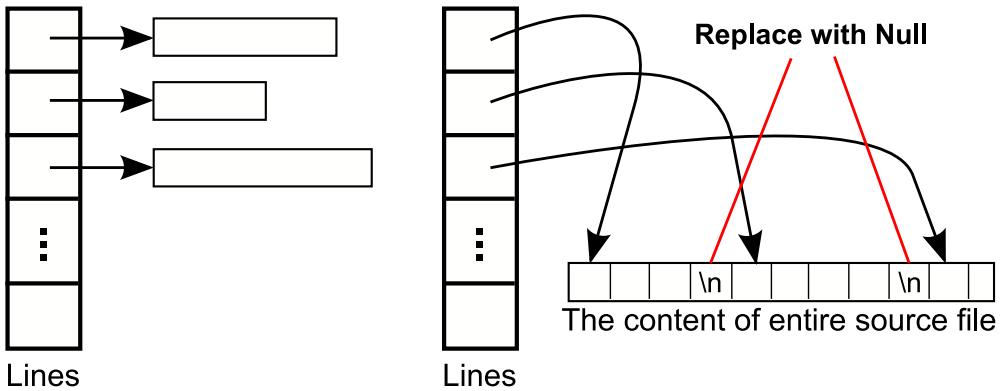
1. A number is *pseudoperfect* if the sum of all or some of its proper divisors is equal to the number itself. Write a program to verify whether a given number is pseudoperfect or not.
   Note: An efficient implementation will adopt a dynamic programming approach.
2. Write a program to verify whether an input matrix is square or not. If it is not a square matrix, print `NOT SQUARE`. Otherwise, further check whether it is singular (determinant is 0) or unimodular (determinant is 1). Accordingly, print `SQUARE - SINGULAR` or `SQUARE - UNIMODULAR`, otherwise print `SQUARE - OTHER`.

Problems – Day 1

3. Let us define a string, comprising English alphabets, as `NICE` if each vowel within it are equidistant from its successor and predecessor vowel, if applicable. E.g., "rhythm", "cool", "malayalam" are `NICE` strings. Write a program to verify whether a given string is `NICE` or not. You are required to take the string as a direct input without asking for its length.
4. Define a structure for representing complex numbers. Using this definition, write a function that takes three real numbers, a, b and c as input, and returns the two roots of the quadratic equation $ax^2 + bx + c = 0$. Finally, compute the ratio of the two roots obtained. Recall that, for dividing two complex numbers, you need to multiply the numerator and denominator by their complex conjugates and then simplify.

Problems – Day 1

5. You have to write a program that reads its own source file (i.e., mtc19xx-day1-prog5.c), and prints the lines in that file in lexicographically sorted order. The output of your program should be identical to the output of the command "sort mtc19xx-day1-prog6.c". Recall that, given any two strings s and t, the function `strcmp()` may be used to determine the lexicographic ordering of s and t.
   Note: An efficient implementation is highlighted below.



Problems – Day 1

6. Suppose there are two separate files containing sufficiently large integer values. Write a program that will take those two filenames as command line arguments and return the multiplication result.
   Note: An efficient implementation will not depend on the primary memory of the system.