

1. Define a structure to represent complex numbers. Using this definition, write a function that takes as input 3 real numbers, a , b , and c , and returns the **two** roots of the quadratic equation $ax^2 + bx + c = 0$.
2. Write a program to count the number of occurrences of each keyword in a C program.
THIS IS WORKED OUT IN SECTION 6.3 OF K&R. PLEASE TRY TO SOLVE THIS ON YOUR OWN AND COMPARE AGAINST THE SOLUTION IN THE TEXTBOOK.
3. Write a program that reads, stores and pretty-prints the roll number, name, and UID of the students in your class. This information is available from the `/etc/passwd` file on the machine where you are working. You may assume that the lines in `/etc/passwd` are at most 256 characters long.

4. You are given n integers, $0, 1, 2, \dots, n - 1$. Each integer i ($0 \leq i \leq n - 1$) is associated with a *label*. Initially, $label(i) = i$ for $0 \leq i \leq n - 1$. We now define an operation **union**(j, k) as follows. Let the labels of j and k be L_j and L_k respectively. For each integer $i \in \{0, \dots, n - 1\}$ that has label L_k , **union**(j, k) changes the label of i to L_j .

Write a program that takes as input a sequence of M **union** operations, and prints $label(i)$ for $i \in \{0, \dots, n - 1\}$.

Design your program in two different ways to handle the following two cases efficiently: (a) $M \gg n$; (b) $n \gg M$.

5. Let **s** and **t** be strings containing at most 100 characters. Implement the following functions in C:
- (a) `strlen(s)`: returns the length of **s**, i.e., the number of characters present in **s**;
 - (b) `strcmp(s, t)`: returns 1 if **s** and **t** are identical, 0 otherwise;
 - (c) `diffByOne(s, t)`: returns 1 if **s** and **t** are of the same length, and differ in exactly one position, 0 otherwise.

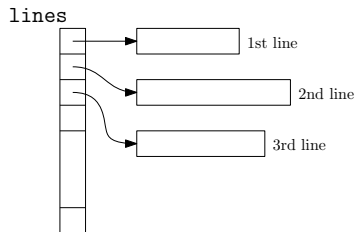
6. Write a function `uniquify` that

- takes two arguments: an array `a` containing at most 100 non-negative integers, and `n`, the number of integers contained in the array, and
- stores in `a` a list of the *distinct* integers contained in `a` if the integers in `a` are sorted in increasing order, and returns the number of distinct integers in `a`;
- returns -1 otherwise.

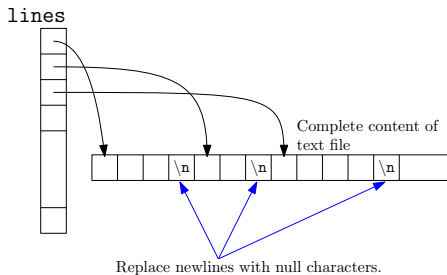
7. Suppose that we choose to represent sets by arrays of non-negative integers, containing at most 100 elements each. Let A and B be 2 such sets of integers. Implement the following set operations:
- (a) $A \cup B$;
 - (b) $A \cap B$;
 - (c) $A \subseteq B$ (returns 1 if A is a subset of B , 0 otherwise);
 - (d) prints the power set of A on the screen.

8. Write a program to print the lines of a text file in alphabetically sorted order.

Naive Method



Better Method



HINT: `$ man 2 stat`