Floating Point Arithmetic

Consider the following two formulas for the solution x to the equation $ax^2+bx+c=0$:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}, \quad x = \frac{-2c}{b \pm \sqrt{b^2 - 4ac}}$$

(a) The first expression for x is the familiar quadratic formula. Show that the second expression is equivlent

(b) In exact arithmetic, which (if either) of these formulas is better?

(c) Consider the equation $x^2 + 50.01x + 2 = 0$. The roots of this equation are $x_1 \approx -0.040024033...$ and $x_2 \approx -49.969975....$ Compute the + root using each formula above and 5-digit rounding. Which provides the better approximation?

(d) Now do the same for the $-$ root. Explain each of these results.

Gaussian Elimination - Row Interchanges

Consider the following matrix and vector

$$A = \begin{bmatrix} \delta & 2 \\ 1 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

where $\delta << 1$. Let's consider asking a computer to solve the system $Ax = b$, using Gaussian elimination. Natrually, the computer must use some form of floating point arithmetic to solve this problem. We will investigate what may (or may not) go wrong.

(a) If we were designing an algorithm to solve linear systems, would we expect(or at least hope) the accuracy of our result to get better or worse as $\delta \to 0$? (Hint: Set $\delta = 0$. Is the system 'easy' or 'hard' to solve at this point?)

(b) Set $\delta = 10^{-k}$ for some integer $k > 0$. Determine the approximate solution of the system, $\tilde{x}$, using GE and floating point arithmetic. What happens if our floating point representation is k-digit rounding? (Hint: set $k = 4$ and work out the exact answer if you are stuck)

(c) Repeat the above, but now use GE with partial pivoting. If we use $(k-1)$-digit rounding now, has the accuracy improved?

(d) Qualitatively speaking, why have we noticed the results above?