1. (5 Points) Explain why the (repaired1) ForwardElimination algorithm on page  
210 of Levitin fails to provide a solution for:

x1 + x2 + x3 = 6  
x1 + x2 + 2x3 = 9  
x1 + 2x2 + 3x3 = 14  
despite the fact that x = (1, 2, 3) or x1 = 1, x2 = 2, x3 = 3 can be easily verified  
as a solution to the system.

How does the BetterForwardElimination algorithm on page 211 of Levitin rem-  
edy this?

The forward elimination algorithm fails the system of equations above because one of the diagonal elements will be zero. When one of the diagonal elements is zero, it will have a divide to 0 error. For the system of equation above, subtracting the first and second row will cause x2 to have a 0 in its coefficient.

The better forward elimination algorithm remedies this issue by allowing partial pivoting. This allows a row with a larger pivot value to be swapped with the current row being evaluated. This means that rows 2 and 3 will swap, creating valid solutions for better forward elimination.

2. (10 Points) Explain in some detail why the BetterForwardElimination algorithm  
on page 211 of Levitin fails to provide a solution for:  
x1 + x2 + x3 = 6  
x1 + x2 + 2x3 = 9  
2x1 + 2x2 + 3x3 = 15  
despite the fact that x = (1, 2, 3) or x1 = 1, x2 = 2, x3 = 3 can be easily verified  
as a solution to the system.

What can be done to remedy this shortcoming in the algorithm?

Better forward elimination fails because it doesn’t consider inconstant or redundant solutions. For the system of equations above, the last row is swapped for the first row. Now, doing forward elimination to cancel out x1 also inadvertently cancels x2 out. Continuing with the algorithm, it’ll be divided by 0 when trying to cancel out x2.

One way to remedy this issue is to recognize that the matrix needs to be more consistent or varied. If no row swaps can swap a pivot of zero, the program will skip that pivot (avoid dividing by zero).