Textbook problems to be graded:

- 1.1.20, 1.1.25
- 1.2.4 (you can’t simply cite Theorem 1.2.3), 1.2.16(a), 1.2.20
- 1.3.4, 1.3.14, 15 (write proper MATLAB code)

Written problems to be graded:

- Explain why an algorithm of $O(n^2)$ may not be faster than an algorithm of $O(n^3)$, even though they are calculating the same thing.

- Using basic programming (for loops, while loops, and if statements), write two functions in Matlab, both taking as input:
  - dimension n
  - $n \times n$ matrix A
  - $n \times n$ matrix B
  - $n \times 1$ matrix x

Have the first function compute $ABx$ through $(AB)x$ and the second compute $A(Bx)$. Have both output:

- the number of flops used

1. Print out and turn in the first function
2. Print out and turn in the second function
3. Apply each algorithm for $n = 100, 200, 400, 800$. Which approach is better?

Additional Problems to try but don’t have to be turned in:

- 1.1.9 (notice there’s just one short question at the end),