1. Answer the following questions about stability and backward stability. Remember, both stability and backward stability require errors to be on the order of $O(\epsilon^\alpha)\|\text{input}\|$ for some $\alpha \geq 1$.

   (a) Let $f_1(x) = x - x$ for any $x \in \mathbb{C}$, and $\tilde{f}_1(x) = \text{fl}\left(\text{fl}(x) - \text{fl}(x)\right)$. Is $\tilde{f}_1$ stable? Is it backward stable?

   (b) Let $f_2(x) = \frac{x}{x}$ for any $x \in \mathbb{C} \setminus \{0\}$, and $\tilde{f}_2(x) = \text{fl}\left(\frac{\text{fl}(x)}{\text{fl}(x)}\right)$. Is $\tilde{f}_2$ stable? Is it backward stable?

   (c) Let $f_3(x) = \sqrt{x}$ for any $x \in \mathbb{C}$, and let $\tilde{f}_3(x)$ be the value $y \in F$ which minimizes

   $$|\text{fl}(\text{fl}(y * y) - \text{fl}(x))|.$$

   Is $\tilde{f}_3$ stable? Is it backward stable?

   (d) Let $f(a, b, c)$ output the smallest magnitude root of the quadratic polynomial $p(x) = ax^2 + bx + c$, so that $f : \mathbb{C}^3 \to \mathbb{C}$. Let $\tilde{f}$ be defined by

   $$\tilde{f}(a, b, c) = \text{fl}\left(\frac{\text{fl}\left(-b \pm \tilde{f}_3\left(\text{fl}\left(\text{fl}\left(b * b\right) - \text{fl}\left(4 * a * c\right)\right)\right)\right)}{\text{fl}(2 * a)}\right) \text{ for } a, b, c \in F,$$

   where $\tilde{f}_3$ is defined as above. Is $\tilde{f}$ stable? Is it backward stable?
   Hint: You might want to consider $b \approx 1$, $c \approx 0$, and $a \in F \cap \mathbb{R}$ of size $\approx \sqrt{\epsilon} > \epsilon$.

2. **MATLAB EXERCISE – Turn in printouts of your programs and plots:** Compute the sum

   $$\sum_{k=1}^{5000} \frac{1}{k^{1.1}} \approx 6.3177$$

   in two ways. Describe what you see. Do you get the precision you expect?

   (a) Round all intermediate sums to 4 floating point significant digits of accuracy (using, e.g., Matlab's "round" command appropriately) to simulate lower precision arithmetic, and sum from the largest to the smallest term.

   (b) Round all intermediate sums to 4 floating point significant digits of accuracy to simulate lower precision arithmetic, and sum from the smallest to the largest term. You might also like to use Matlab's "flip" command.

3. Do 16.1 on page 119 of Trefethen and Bau. You may cite theorems from class to save a little time. However, make sure to read the problem carefully, and to show backward stability via the definition given in class.

4. Do 17.2 on page 127 of Trefethen and Bau.

5. Use Weyl's bounds from class to answer the following questions:

   (a) You enter a unitary matrix $Q \in \mathbb{C}^{n \times n}$ into a digital computer. In the process you obtain a new matrix $\tilde{Q}$ with $\tilde{Q}_{i,j} = \text{fl}(Q_{i,j})$ for all i, j. Give the best upper bound you can for the condition number of the matrix $\tilde{Q}$. Be explicit in your bound about the dependence on both n and the machine $O(\epsilon)$-precision. Do not consider n to be a constant here!

   (b) You now multiply $\tilde{Q}$ by it's adjoint. Give the best upper bound you can for the condition number of the matrix $\tilde{Q}^*\tilde{Q}$. Be explicit in your bound about the dependence on both n and the machine $O(\epsilon)$-precision. Do not consider n to be a constant here!

6. Do 18.1 on page 136 of Trefethen and Bau.

7. Do 18.4 on page 136 of Trefethen and Bau.

8. Do 18.2 on page 136 of Trefethen and Bau.
9. Do 19.1 on page 143 of Trefethen and Bau.