Exercises: (Chapter 6 §12 and §13, and Chapter 7 §1)

1. Implement a recursive FFT in the programming language of your choice. Test it against the direct method[1] on the function $f(x) = \cos(1234 * x)$, with $N = 32768$, to verify the FFT is faster.
   NOTE: You can use "time" at a unix prompt to see how long a program takes. For example, typing "time a.out" will run a.out and then tell you how long it took. Any other method of reporting runtime that you prefer also works though!
   Turn in both your code and the runtimes you recorded.

2. Let $\omega \in \mathbb{Z}$. Show that
$$\frac{1}{2\pi}\int_0^{2\pi} \mathbb{e}^{\mathbb{i}\omega x}\,dx \;=\; \begin{cases} 1 \text{ if } \omega = 0 \\ 0 \text{ if } \omega \neq 0 \end{cases} \quad .$$

3. Let $p : \mathbb{R} \to \mathbb{C}$ be a degree n trigonometric polynomial given by $p(x) = \sum_{\omega=-n}^{n} \hat{p}_\omega \mathbb{e}^{\mathbb{i}\omega x}$. Show that
$$\frac{1}{2\pi}\int_0^{2\pi} |p(x)|^2\,dx \;=\; \sum_{\omega=-n}^{n} |\hat{p}_\omega|^2.$$

4. Let a numerical process be described by
$$L = \phi(h) + \sum_{j=1}^{\infty} a_j h^j.$$
   Explain how Richardson extrapolation will work in this case. Prove an analogue of Theorem 1 in Section 7.1 of the book for this case.

5. Derive the following two formulas for approximating the third derivative of a function. Find their error terms. Which formula is more accurate?
$$f'''(x) \approx \frac{1}{h^3}\left[f(x+3h) - 3f(x+2h) + 3f(x+h) - f(x)\right]$$
$$f'''(x) \approx \frac{1}{2h^3}\left[f(x+2h) - 2f(x+h) + 2f(x-h) - f(x-2h)\right]$$

[1] You can find example direct method C code at https://users.math.msu.edu/users/iwenmark/Teaching/MTH451/HWandCode/DirectDFT.c.