Math 242 Challenge Problems

Spring 2024

The explorations and exercises below refer to the *Computational Mathematics* text. See the text for full details and guidance about each problem.

1. **Exploration 1.16: π as area.** Implement the algorithm outlined in this exercise (pages 24–25 in the text) to approximate π as the area of a quarter circle. Analyze and discuss the accuracy and efficiency of this method.

2. **Exploration 1.23: Create your own Machin-like formulas**. Use the methods described in Section 1.4 of the text to find your own Machin-like formulas for π. This exploration requires you to find at least three Machin-like formulas, convert each formula to a power series, implement them in Mathematica, and assess their accuracy for approximating π. See the text for details.

3. **Exploration 1.24: Accuracy of Machin-like formulas.** Investigate the accuracy of Machin-like formulas, such as those in Section 1.4 of the text. Formulate a precise conjecture, supported by your own computational evidence, about how the accuracy of the formula depends on the values of the a_i and b_i. (See page 29 of the text.) For this problem it is important to consider lots of Machin-like formulas and look for patterns.

4. **Exercise 1.43: Continued Fractions for π.** Read Section 1.7 of the text. Implement one of the methods for computing convergents of continued fractions. Use your code to complete Exercise 1.43.

5. **Exploration 2.25: Generalize a Fibonacci identity.** Start with the identity in the text, introduce a new index (that is, a new subscript such as k), and conjecture a new identity.

6. **Exploration 2.26: Fibonacci identities.** Search for Fibonacci identities based on the four expressions given in the text.

7. **Exploration 2.38: Generalized Fibonacci polynomial identity.** Search for a general formula that gives the coefficients of the Fibonacci $(2q+1)n$ identity for odd integers $2q+1$.

8. **Exploration 2.41: Fibonacci identities involving sums.** Explore a class of Fibonacci identities involving $\sum_{a+b=n} F_a F_b$.

9. **Exploration 3.8: Collatz trajectory sets.** Explore the sets of integers that arise in Collatz trajectories for certain sets of starting values. Be careful! This exploration is about *sets*, not *sequences*.

10. **Exploration 3.31: Collatz stopping times.** Explore how the maximum stopping time for Collatz trajectories grows with n. See the text for details.

11. **Exploration 3.32: "Horizontal segments" in Collatz stopping time plot.** Investigate the horizontal line segments that appear in the Collatz stopping time plot. How does the length of these segments increase with n?

12. **Exploration 3.37: Accelerated Collatz trajectories.** Compute accelerated Collatz trajectories for big integers, as stated in the text. Do your computations support Estimate 3.4 in the text?

13. **Exploration 3.70: Ergodicity of logistic map trajectories.** How many iterations of the logistic map are required, on average, until the trajectory contains a point in each interval of size $\frac{1}{M}$? How does this depend on M? See the text for details.

14. **Exploration 4.22: a prime polynomial.** How often do polynomials produce primes? Find prime numbers produced by the polynomial $f(n)$ in the text and compare with other polynomials. For this exploration, you should consider *lots* (hundreds, at least) of quadratic polynomials and discuss your observations.

15. **Exploration 4.37: counting prime pairs.** Implement a function that computes $\pi_k(x)$ as described in the text, then compute and plot $\pi_k(x)$ for various even integers $k > 6$. What do you observe? What conjectures can you formulate?

16. **Exploration 4.45: Pseudoprime bases.** Investigate the function $\sigma(n)$ as described in the text.

17. **Exploration 4.48: Carmichael numbers.** Carmichael numbers are composite integers that are likely to be labeled as prime by primality tests based on Fermat's little theorem. Implement a function to find Carmichael numbers, as described in the text.

18. **Exploration 4.55: Strong primality test.** Implement a strong primality test with a low failure probability that you can quantify. See the text for details.

19. **Exploration 4.79: Riemann's explicit formula.** Implement a function to compute $\pi_0(x)$ as described in the text. Demonstrate that you can compute partial sums that appear to converge to the prime-counting function $\pi(x)$.

20. **Exercise 5.60: Seat assignment problem.** Create a simulation of the seat assignment problem, as stated in the text. Estimate the probability that the last person sits in their assigned seat, and discuss how this answer depends on the number of people N. See the text for details.

21. **Exercises 5.62–5.63: Boxes and prizes.** Implement both the naïve approach and the sophisticated strategy, as described in the text. What is the probability of winning with each strategy? Discuss.