# Homework 2 (due Tuesday, October 8, 2024)

**Remark.** Please also turn in your code and worksheet for any program that you use for this homework.

1. **Archimedes method of $n$-gon**

   a) `(1 point)` Find the values of $p_3$ and $P_3$.

   b) `(1 point)` Use the iteration method of Archimedes to compute $P_6$, $p_6$, $P_{12}$, $p_{12}$, $P_{24}$, $p_{24}$, $P_{48}$, $p_{48}$, $P_{96}$, and $p_{96}$. How many digits of $\pi$ does this method compute correctly?

2. **Approximate $\sqrt{2}$**

   a) `(1 point)` Use what you learned from calculus to find the infinite series that can approximate $\sqrt{2}$.

   b) `(2 points)` Design your own simulation that can approximate $\sqrt{2}$. Find the 95% confidence interval from simulating your model.

3. **Generalized Monty Hall**

   a) `(1 point)` Modify the program to guess the probability of winning when there are 5 doors with 2 cars and 3 goats. In this variant, after the contestant chooses a door, the host opens one door that has a goat behind it. (If the host has a choice to make, they do so uniformly at random among the doors with goats behind.)

   b) `(1 point)` Prove your answer in part a).

4. **Truncated arctan series from class**

   Recall that we considered the following *truncated arctan series* in Lecture 3:

$$A_n := 4 \sum_{i=1}^{n} \frac{(-1)^{i+1}}{2i-1}.$$

   Example outputs include

$$A_{100} = 3.1315929...$$

   and

$$A_{10000} = 3.141492653590....$$

   Let us compare these results to the known approximation for $\pi$:

$$\pi \approx 3.14159265358979....$$

a) `(1 point)` Notice something interesting about the outputs. It seems that we have some number of decimal digits of $\pi$ correct from the left, and then there is one digit incorrect, and after that digit we continue to have more correct digits for $\pi$. Is this a coincidence? Could you provide some intuitive, or heuristic, or informal explanation for this? *Note:* In order to receive full credit for part a), you only need to show that you have spent some time thinking about this phenomenon. You do *not* need to give mathematically rigorous explanation for this part. (But if you want mathematical rigor, read part c) below.)

b) `(1 point)` Based on the observation above, it appears that the formula

$$\pi \approx 4\left(\frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots + \frac{(-1)^{n+1}}{2n-1}\right)$$

could be slightly improved. Give an improved formula for approximating the constant $\pi$.

c) **(extra credit)** `(+1 extra point)` (This part is extra credit. We recommend that you finish all the other problems—and possibly all other homework from other classes—first. Depending on how you approach it, it could be tedious. You have been warned.) Using the Euler–Maclaurin summation formula, determine, with rigorous proofs, the absolute constants $C_1, C_2 \in \mathbb{R}$ such that the asymptotic formula

$$A_{2n} = \pi + C_1 \cdot \frac{1}{n} + C_2 \cdot \frac{1}{n^2} + O\left(\frac{1}{n^3}\right)$$

holds for $n \in \mathbb{Z}_{\geq 1}$, and show that the remainder term $r_n$ defined by

$$A_{2n} = \pi + C_1 \cdot \frac{1}{n} + C_2 \cdot \frac{1}{n^2} + r_n$$

satisfies

$$|r_n| < \frac{3}{4n^3},$$

for every positive integer $n \in \mathbb{Z}_{\geq 1}$.

5. **(extra credit)** `(+1 extra point)` **Yet another way to approximate $\pi$.**

In this problem, we discuss another way to approximate $\pi$ "from scratch" using binary search and approximate polynomials. Let us begin with a binary search.

Suppose that $a < b$ are real numbers, and suppose that a function $f : [a, b] \to \mathbb{R}$ is strictly increasing and continuous on $[a, b]$, with $f(a) < 0 < f(b)$. By the intermediate value theorem from calculus,[1] we know there exists a unique real number $u \in (a, b)$ for which $f(u) = 0$.

When $f$ is such a nice function as we described above, we can approximate $u$ by a binary search. We provide a simple pseudocode in Algorithm 1.

[1] Yes, we have calculus again!

**Algorithm 1** `BinSearch`$(a, b, \varepsilon, f)$

```
L ← a                                        ▷ Lower bound
U ← b                                        ▷ Upper bound
while U − L ≥ ε do
    m ← (L + U)/2                            ▷ Midpoint of [L, U]
    if f(m) ≥ 0 then
        U ← m                                ▷ Update the upper bound
    else
        L ← m                                ▷ Update the lower bound
    end if
end while
return the interval [L, U].
```

a) Write the `BinSearch`$(a, b, \varepsilon, f)$ program. Test your program by plugging in

$$a = 1, \quad b = 2, \quad \varepsilon = 10^{-10},$$

and $f : x \mapsto x^3 - 2$, to approximate $\sqrt[3]{2}$ up to 7 significant digits.

Now that we have a binary search program, let us try to approximate $\pi$. The idea is to note that $\pi$ is the unique solution to the equation

$$2\sin\left(\frac{x}{6}\right) - 1 = 0$$

in the interval $x \in [0, 4]$. But because we want to do this "from scratch", let us *not* use the sine function. Instead, we use polynomial approximations to it.

Recall that we have the following Taylor expansion:

$$\sin\left(\frac{x}{6}\right) = \sum_{k=1}^{\infty} \frac{(-1)^{k+1}}{(2k-1)! \cdot 6^{2k-1}} \cdot x^{2k-1}.$$

Let us truncate the series. For each positive integer $n$, define

$$P_n(x) := \sum_{k=1}^{n} \frac{(-1)^{k+1}}{(2k-1)! \cdot 6^{2k-1}} \cdot x^{2k-1}.$$

For instance, $P_1(x) = \frac{x}{6}$, $P_2(x) = \frac{x}{6} - \frac{x^3}{1296}$, and so on. Now each $P_n(x)$ is simply a polynomial in $x$.

b) For each $n = 1, 2, \ldots, 8$, let $f_n$ denote the function $x \mapsto 2P_n(x) - 1$. Run `BinSearch`$(0, 4, 10^{-10}, f_n)$. How good are these approximations of $\pi$?

6. **(extra credit)** `(+1 extra point)` **Newton's Method**

In this problem, we discuss Newton's method, which is a powerful way to approximate solutions to equations.

Consider the following equation

$$x^2 - \sin(x+1) - 1 = 0. \tag{1}$$

We know there are two distinct real solutions to (1): one of them is $-1$, which can be seen easily, and the other is some positive real number between 1 and 2, it seems. Our goal is to approximate this mysterious positive solution to many decimal digits.

Here's how we do Newton's method. First, write

$$f(x) := x^2 - \sin(x+1) - 1.$$

Now we would like to solve for the real number $x > 0$ for which $f(x) = 0$. We begin with an initial *guess*. Say we consider

$$a := 1.$$

From the initial guess, create the following sequence $\{x_n\}_{n=1}^{\infty}$, by defining $x_1 := a$, and for each $n \geq 1$, let

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)},$$

as long as $f'(x_n) \neq 0$. (If $f'(x_n)$ is zero at some point, then quit the process.)

Recall that

$$f'(x) = 2x - \cos(x+1),$$

from calculus.

a) Write a code to perform Newton's method. Begin with the initial guess $a = 1$. Output the sequence

$$x_1, x_2, x_3, \ldots$$

for a certain number of terms. You should find that the sequence seems to converge to a certain number. Call it $A$.

Indeed, $A = \lim_{n\to\infty} x_n$ turns out to be the desired solution to (1) in the region $x > 0$.

b) Comment on the speed of convergence of the sequence from part a).

c) Now try part a) again with different initial guesses. Do you always find that the limit of the output sequence seems to converge to the desired solution to (1) in the region $x > 0$?

d) Now use the binary search program `BinSearch` you wrote in the previous problem to find an alternative approximation to the solution. Compare the two results.