Applied Mathematics 2130
Lab 2006W–2B

The constant π has a long and interesting history. The Egyptians and Babylonians (circa 2000 B. C.) knew that the ratio of the circumference of a circle to its diameter was a constant. The Babylonians used 25/8 as an approximation to it while the Egyptians estimated it to be $4(8/9)^2$. The more accurate value 22/7 was often used and over the years that value was refined to a considerable extent, so that to this day algorithms for computing (millions of) digits of π remain a standard benchmarking tool for testing new computer architectures.

Some interesting algorithms for computing π were devised long before the age of modern computing. The purpose of this lab is to explore the algorithm developed by that famous scientist and philosopher, Archimedes of Syracuse (287–212 B.C.). He described the oldest known algorithm which enables the calculation of π to any desired accuracy: a novel idea for his period.

Imagine a circle with unit diameter. Archimedes suggested we inscribe regular n-sided polygons inside the circle and use the ratio of circumference to diameter of the polygon as a lower bound to π. At the same time, he suggested we also circumscribe about the circle a regular n-sided polygon and use the same calculation to obtain an upper bound to π. By choosing enough sides (make n larger and larger) we can generate more accurate approximations to π.

Methodology

Archimedes' estimates of π were based on 96-sided polygons. When you consider that he did this with no decimal number system, no trigonometry and no systematic means of computing square roots, then the feat becomes more than remarkable.

Find a regular hexagon ($n = 6$) inscribed in the unit circle centred at the origin of a standard two-dimensional coordinate system. Calculate the length of each side of the hexagon and thereby compute a lower bound for π.

To proceed to larger values of n, imagine that the edges of the hexagon are rubber bands. To construct a 12-sided figure from a 6-sided one, grab the midpoint of each of the hexagon's edges and pull those midpoints to the circle boundary in the centrifugal direction. We now have a dodecagon inscribed in the circle. What is the length of each side of this dodecagon? If you label one side of the hexagon ℓ_6 and a side of the dodecagon ℓ_{12}, then you can derive a formula for ℓ_{12} in terms of ℓ_6 using only the basic arithmetic operators $+, -, \times, \div, \sqrt{\ }$. In fact, all you need here is the theorem of Pythagoras. After you do this, it should be clear how to proceed in an iterative fashion, so that you can determine the length of the edges of a regular 24-sided polygon, then a 48-sided one, and so on. Upper bounds to π can be obtained in an analogous way, using circumscribed polygons.

Archimedes started with a hexagon ($n = 6$), because, although he had no trigonometry, he knew that in a $30 - 60 - 90$ triangle, the ratio of the short side to the hypotenuse is always 1/2 (that is, $\sin(\pi/6) = 0.5$). A means of computing square roots was required and none were available. Like any good mathematician, he made one up!

You may also start with an inscribed pentagon ($n = 5$) using the (less known) fact that $\cos(\frac{\pi}{5}) = \frac{1+\sqrt{5}}{4}$. In this approach, you will compute the perimeters of inscribed and circumscribed $10, 20, \ldots n$-gons obtained by the iteration, as explained before.

One of the tasks in this lab is to write a computer program that will implement Archimedes' algorithm. Your program should include the following ingredients.

- The user should be asked for an initial value of $n = n_0$. The program should then determine an estimate for π based on the perimeters of inscribed and circumscribed n-gons.
- Proceed to $n_1 = 2n_0$-sided polygons, then $n_2 = 4n_0$-sided polygons and so on up to the number of sides equal to $n_m = 2^m n$, where m is another user-inputted value. The program should only employ the basic arithmetic operators $+,\ -,\ \times,\ \div,\ \sqrt{\ }$. **Trigonometric functions must not be used in your program.**

After your program is working well, investigate how fast the difference between the upper and lower bounds decreases as n gets larger. Also, find out how fast these values approximate a high-precision value of π found elsewhere.

In the year 264 A. D., a Chinese scholar named Liu Hui estimated $\pi = 3.14159$ using a 3072-sided polygon (unaware that Archimedes had already invented the same algorithm 400 years earlier). Can you verify the accuracy of his calculation? How many sides would be required to obtain ten digit accuracy?

Episodes like this from the history of π will make great material for your report! Suitable graphics must be given. Describe the mathematical details of your implementation of Archimedes' algorithm along with a list of upper and lower bounds based on sample values of n.