Motivation
We practise our knowledge of floating point arithmetic.

Doubles
What is the largest number representable as a double, the smallest positive number, the smallest normalized positive number?

Some Computations are Exact
Let $a, b \in F$ with $\frac{1}{2} \leq \frac{a}{b} \leq 2$. Show that $a \ominus b = a - b$. This was first observed by Sterbenz.

Doubles and Orientation
Assume for this exercise that point coordinates are doubles in $[1/2, 1]$. Show

- $\text{orient}(p, q, r) = 0$ implies $\text{floatorient}(p, q, r) = 0$.
- $\text{floatorient}(p, q, r) \neq 0$ implies $\text{orient}(p, q, r) = \text{floatorient}(p, q, r)$.
- What does this mean for the geometry of float-orient?
- Can you find examples that make the floating point implementation of the convex hull algorithm crash when point coordinates are restricted to doubles in $[1/2, 1]$?

Error Analysis
Assume that a point $p$ is given by its homogeneous coordinates $(px, py, pw)$. Assuming $\text{sign}(aw \cdot bw \cdot cw) = 1$, we have

$$\text{orient}(a, b, c) = \text{sign}(aw \cdot (bx \cdot cy - by \cdot cx) - bw \cdot (ax \cdot cy - ay \cdot cx) + cw \cdot (ax \cdot by - ay \cdot bx)).$$

Compute the $d$-value and $m$-value of this expression.

A High Precision Computation of $\pi$
Show how to compute $\pi$ with an error less than $2^{-200}$. 
Linear Kernel

In class we discussed the concept of a linear kernel and several models of it. The notes contain a C++ implementation. Give an implementation in a programming language of your choice (preferably not C++).

Rational Points on a Circle

In class we showed that for any rational point \( p = (p_x, p_y) \) on the unit circle there is a rational \( a \) such that

\[
(p_x, p_y) = \left( \frac{2a}{a^2 + 1}, \frac{a^2 - 1}{a^2 + 1} \right).
\]

In order to find a rational point in direction \( \alpha \), we therefore need to find a rational \( a \) such that

\[
a \approx \frac{1}{\cos \alpha} + \sqrt{\frac{1}{\cos^2 \alpha} - 1}.
\]

Show how to find such an approximation with error less than \( 2^{-t} \) by binary search.

Have fun with the solution!