MATHEMATICAL ALGORITHMS I

1. Set 4: Fourier transform and even more multiplication

1.1. Problem: Repeated squaring and powers. Implement the repeated squaring algorithm that computes an integer power of a template variable. Here is an example of a template:

```
template <typename generic_float>
generic_float sq(generic_float x)
{
    return x*x;
}
```

This function computes the square of the input variable. Unlike the usual syntax, any variable for which the product `*` is defined is allowed. If `int`, `double`, etc all work.

1.2. Problem: Fourier series and SFT. Consider polynomials of degree at most $n-1$.

(1) Suppose that f is a real polynomial, i.e. $f_0, \ldots, f_{n-1} \in \mathbb{R}$. We interpret f as a vector in $\mathbb{C}^n$. Show that the k-th component of the vector $DFT_\omega(f)$
$$DFT_\omega(f)_k = \overline{DFT_\omega(f)_{n-k}},$$
where the $\bar{\ }$ denotes complex conjugation. We use cyclic notation, so $f_n = f_0$.

(2) Design an algorithm that computes the discrete Fourier transform via matrix multiplication with a Vandermonde matrix: let us call this SFT for slow Fourier transform. Analyze the speed of the algorithm: how many ring operations are needed? You don't need to think of a clever way to multiply matrices: you can in fact do better than the obvious way you have learned.

(3) Implement the SFT on a computer: you are allowed to use floats, doubles or long doubles: if you want, you can use `math.h` and `complex.h` to work with complex numbers.

(4) Implement the algorithm for convolution using SFT.

1.3. Problem: FFT.

(1) Implement the algorithm for fast Fourier transform, and adapt convolution to use FFT rather than SFT.

(2) Design an algorithm to compute the product of two polynomials using FFT. Input: two polynomials f, g whose coefficients are `int`'s, output: the product fg.

In other words, we have $f, g, fg \in R[X]$ with $R = \mathbb{Z}/2^w\mathbb{Z}$.

Remark 1.1. Working with `double`s involves rounding errors, which should be dealt with. However, ignore this issue for now. If time permits, we describe how this can be controlled. Also keep in mind that FFT involves an imaginary part.