CPS815/CP8201 - Assignment 3

Recall the Newton-Raphson method from Calculus. It was used to compute an approximate root of a smooth real-valued function $f(x) = 0$. The idea was to start from an initial value $x_0 \in \mathbb{R}$ and compute the sequence $x_1, x_2, \ldots$ of numbers using the formula

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

If x_0 is an appropriate initial value, i.e., it is not too far from the root, then it is easy to prove that the above sequence converges very quickly to the actual root.

In this assignment, we want to adapt the Newton-Raphson method to compute specific sequences of polynomials over the rational numbers $\mathbb{Q}$. A polynomial of degree m over $\mathbb{Q}$ is written as

$$g(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_m x^m$$

where the coefficient a_i is rational for all i. We denote by $g \bmod x^k$ the polynomial $g' = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}$. This is the usual division and remainder operation, g' is the remainder of dividing g by x^k. Let $R = \mathbb{Q}[x]/x^N$ be the set of polynomials mod x^N, that is, the set of polynomials of degree less than N. Multiplication and division operations in R is done modulo x^N.

Example. For $N = 4$, the set R consists of polynomials of degree at most 3. Let $f = 2 + x^3$ and $g = 1 + 3x + x^2$. Then

$$\begin{aligned} fg \bmod x^N &= 2 + 6x + 2x^2 + x^3 + 3x^4 + x^5 \bmod x^4 \\ &= 2 + 6x + 2x^2 + x^3. \end{aligned}$$

The inverse of a polynomial $f \in R$ is a polynomial $g \in R$ such that $fg = 1 \bmod x^N$. In the above example for $N = 4$, the inverse of $f = 1 - x$ is $g = 1 + x + x^2 + x^3$ since

$$fg \bmod x^4 = 1 - x^4 \bmod x^4 = 1.$$

A polynomial f is a $\frac{1}{k}$-approximation of a polynomial g if $f - g = 0 \bmod x^k$. In other words, f is a $\frac{1}{k}$-approximation of g if f and g have the first k coefficients in common. For example $f = 3 + x + 9x^2$ is a $\frac{1}{2}$-approximation of $g = 3 + x + 2x^8 + x^{12}$.

1. (60 marks) Given a polynomial $f \in R$ and an integer $t > 0$, write an algorithm to find a $1/2^t$-approximation of the inverse of f. This means that if g is the inverse of f modulo

x^N, then your algorithm should find a polynomial h such that $g - h = 0 \bmod x^{2^t}$. Assume that $f(0) = 1$ so that f always has an inverse mod x^N.

Hint: use the Newton-Raphson method. You need to formulate finding an inverse in the form of finding a root. Start with the constant polynomial $g_0 = 1$ and compute the sequence of polynomials $g_1 \bmod x^2, g_2 \bmod x^{2^2}, \ldots$.

2. (40 marks) Assume two polynomials of degree at most n can be multiplied in $\mathsf{M}(n)$ operations, where $\mathsf{M}(n)$ is a superlinear function: $\mathsf{M}(n_1) + \mathsf{M}(n_2) \leq \mathsf{M}(n_1 + n_2)$. What is the complexity of your algorithm?