Math 161 Final
Priya Malhotra

For my final I chose to prove the closed form equation of the number of irreducible monic polynomials of degree d, labelled as r_d, is $r_d = (1/d)\sum_{y|d}\mu(y)q^{(d/y)}$. There are three main steps to split this up into, as I outline below. This very closely mimics how I split up my main theorems in the Lean code, with some additional lemmas to reuse and make the work more readable. For this first draft, please find a write-up of question formulations and proofs for each of the three parts that directly build on previous parts to lead to a final answer.

Fix a prime number p. A polynomial $f(X) \in \mathbb{F}_p[X]$ is called *monic* if its leading coefficient is equal to one. Write a_d for the number of monic polynomials of degree d and write r_d for the number of irreducible monic polynomials of degree d. Our convention is that $a_0 = 1$.

1. Letting $A(x)$ denote the generating function for the sequence a_d, show that

$$A(x) = \frac{1}{1-px}.$$

. . . .

For a degree d monic polynomial

$$f(X) = X^d + c_{d-1}X^{d-1} + \cdots + c_0$$

in $\mathbb{F}_p[X]$, the vector of coefficients (except for the leading coefficient which is always the same) is $\langle c_{d-1}, \ldots, c_0\rangle \in \mathbb{F}_p^d$. Since each polynomial is uniquely determined by its vector of coefficients, the number of monic polynomials of degree d

$$\begin{aligned} a_d &= |\mathbb{F}_p^d| \\ &= |\mathbb{F}_p|^d \\ &= p^d, \end{aligned} \tag{1}$$

which is also consistent with the convention that $a_0 = 1$.

The power series expansion of

$$\frac{1}{1-z} = \sum_{i=0}^{\infty} z^i, \tag{2}$$

so

$$\begin{aligned} A(x) &= \frac{1}{1-px} \text{ as given} \\ &= \sum_{i=0}^{\infty}(px)^i \text{ by (2)} \\ &= \sum_{i=0}^{\infty} p^i x^i \\ &= \sum_{i=0}^{\infty} a_i x^i \text{ by (1).} \end{aligned}$$

$\square$

2. Using the fact that any polynomial can be uniquely factored into irreducible polynomials, prove that

$$A(x) = \prod_{d \geq 0} \left(\frac{1}{1 - x^d} \right)^{r_d}.$$

. . . .

$$\begin{aligned}
\prod_{d \geq 0} \left(\frac{1}{1 - x^d} \right)^{r_d} &= \prod_{d \geq 0} \left(\sum_{i=0}^{\infty} x^{id} \right)^{r_d} \text{ by (2)} \\
&= \prod_{d \geq 0} \prod_{j=1}^{r_d} \sum_{i=0}^{\infty} x^{id} \\
&= \prod_{\substack{f \in \mathbb{F}_p[X] \\ f \text{ monic} \\ f \text{ irreducible}}} \sum_{i=0}^{\infty} x^{i \deg f} \text{ by definition of } r_d \\
&= \prod_{\substack{f \in \mathbb{F}_p[X] \\ f \text{ monic} \\ f \text{ irreducible}}} \sum_{i=0}^{\infty} x^{\deg f^i} \\
&= \prod_{\substack{f \in \mathbb{F}_p[X] \\ f \text{ monic} \\ f \text{ irreducible}}} (1 + x^{\deg f} + x^{\deg f^2} + \cdots) \\
&= 1 \\
&\quad + \sum_{\substack{f \in \mathbb{F}_p[X] \\ f \text{ monic} \\ f \text{ irreducible}}} x^{\deg f} + \sum_{\substack{\{f,g\} \subset \mathbb{F}_p[X] \\ f,g \text{ monic} \\ f,g \text{ irreducible}}} x^{\deg f} x^{\deg g} + \cdots \\
&\quad + \sum_{\substack{f \in \mathbb{F}_p[X] \\ f \text{ monic} \\ f \text{ irreducible}}} x^{\deg f^2} \sum_{\substack{\{f,g\} \subset \mathbb{F}_p[X] \\ f,g \text{ monic} \\ f,g \text{ irreducible}}} x^{\deg f^2} x^{\deg g} + \cdots \\
&\quad + \cdots \\
&= 1 \\
&\quad + \sum_{\substack{f \in \mathbb{F}_p[X] \\ f \text{ monic} \\ f \text{ irreducible}}} x^{\deg f} + \sum_{\substack{\{f,g\} \subset \mathbb{F}_p[X] \\ f,g \text{ monic} \\ f,g \text{ irreducible}}} x^{\deg f + \deg g} + \cdots \\
&\quad + \sum_{\substack{f \in \mathbb{F}_p[X] \\ f \text{ monic} \\ f \text{ irreducible}}} x^{\deg f^2} \sum_{\substack{\{f,g\} \subset \mathbb{F}_p[X] \\ f,g \text{ monic} \\ f,g \text{ irreducible}}} x^{\deg f^2 + \deg g} + \cdots \\
&\quad + \cdots
\end{aligned}$$

$$
\begin{aligned}
&= 1 \\
&\quad + \sum_{\substack{f \in \mathbb{F}_p[X] \\ f \text{ monic} \\ f \text{ irreducible}}} x^{\deg f} + \sum_{\substack{\{f,g\} \subset \mathbb{F}_p[X] \\ f,g \text{ monic} \\ f,g \text{ irreducible}}} x^{\deg fg} + \cdots \\
&\quad + \sum_{\substack{f \in \mathbb{F}_p[X] \\ f \text{ monic} \\ f \text{ irreducible}}} x^{\deg f^2} \sum_{\substack{\{f,g\} \subset \mathbb{F}_p[X] \\ f,g \text{ monic} \\ f,g \text{ irreducible}}} x^{\deg f^2 g} + \cdots \\
&\quad + \cdots
\end{aligned}
\tag{3}
$$

Because any polynomial can be uniquely factored into irreducible polynomials, and (3) contains all possible combinations of all powers of the irreducible monic polynomials of $\mathbb{F}_p$, (3) is equal to

$$\sum_{\substack{f \in \mathbb{F}_p[X] \\ f \text{ monic}}} x^{\deg f}. \tag{4}$$

When indexed by degree, (4) is equal to

$$\sum_{d=0}^{\infty} a_d x^d, \tag{5}$$

which is $A(x)$ by definition. □

3. By taking logarithmic derivative (i.e. computing $\frac{\mathrm{d}}{\mathrm{d}x} \log A(x) = \frac{A'(x)}{A'(x)}$ in two ways using above formulas for $A(x)$ and comparing the results) deduce that

$$p^n = \sum_{d|n} d r_d.$$

Explain why this relation determines the numbers r_d uniquely.

. . . .

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}x} \log \frac{1}{1-px} &= \frac{\mathrm{d}}{\mathrm{d}x} \log \prod_{d \geq 0} \left(\frac{1}{1-x^d} \right)^{r_d} \\
\frac{\frac{\mathrm{d}}{\mathrm{d}x} \frac{1}{1-px}}{\frac{1}{1-px}} &= \frac{\mathrm{d}}{\mathrm{d}x} \log \prod_{d \geq 0} \left(\frac{1}{1-x^d} \right)^{r_d} \\
\frac{\frac{p}{(1-px)^2}}{\frac{1}{1-px}} &= \frac{\mathrm{d}}{\mathrm{d}x} \log \prod_{d \geq 0} \left(\frac{1}{1-x^d} \right)^{r_d} \\
\frac{p}{\frac{1}{1-px}} &= \frac{\mathrm{d}}{\mathrm{d}x} \log \prod_{d \geq 0} \left(\frac{1}{1-x^d} \right)^{r_d} \\
\frac{p}{\frac{1}{1-px}} &= \frac{\mathrm{d}}{\mathrm{d}x} \sum_{d \geq 0} \log \left(\frac{1}{1-x^d} \right)^{r_d}
\end{aligned}
$$

$$
\begin{aligned}
\frac{p}{\frac{1}{1-px}} &= \frac{\mathrm{d}}{\mathrm{d}x}\sum_{d\geq 0} r_d \log\frac{1}{1-x^d} \\
\frac{p}{\frac{1}{1-px}} &= \frac{\mathrm{d}}{\mathrm{d}x}\sum_{d\geq 0} r_d \log\frac{1}{1-x^d} \\
\frac{p}{\frac{1}{1-px}} &= \sum_{d\geq 0} r_d \frac{\mathrm{d}}{\mathrm{d}x}\log\frac{1}{1-x^d} \\
\frac{p}{\frac{1}{1-px}} &= \sum_{d\geq 0} r_d \frac{\frac{\mathrm{d}}{\mathrm{d}x}\frac{1}{1-x^d}}{\frac{1}{1-x^d}} \\
\frac{p}{\frac{1}{1-px}} &= \sum_{d\geq 0} r_d \frac{\frac{dx^{d-1}}{(1-x^d)^2}}{\frac{1}{1-x^d}} \\
\frac{p}{\frac{1}{1-px}} &= \sum_{d\geq 0} r_d \frac{dx^{d-1}}{1-x^d} \\
p\frac{1}{\frac{1}{1-px}} &= \sum_{d\geq 0} dr_d x^{d-1}\frac{1}{1-x^d} \\
p\sum_{n=0}^{\infty}(px)^n &= \sum_{d\geq 0} dr_d x^{d-1}\sum_{n=0}^{\infty}x^{nd} \text{ by (2)} \\
p\sum_{n=0}^{\infty}p^n x^n &= \sum_{d\geq 0} dr_d x^{d-1}\sum_{n=0}^{\infty}x^{nd} \\
\sum_{n=0}^{\infty}p^{n+1} x^n &= \sum_{d\geq 0} dr_d \sum_{n=0}^{\infty}x^{(n+1)d-1} \\
\sum_{n=1}^{\infty}p^{n} x^{n-1} &= \sum_{d\geq 0} dr_d \sum_{n=1}^{\infty}x^{nd-1} \\
x\sum_{n=1}^{\infty}p^{n} x^{n-1} &= x\sum_{d\geq 0} dr_d \sum_{n=1}^{\infty}x^{nd-1} \\
\sum_{n=1}^{\infty}p^{n} x^{n} &= \sum_{d\geq 0} dr_d x\sum_{n=1}^{\infty}x^{nd-1} \\
\sum_{n=1}^{\infty}p^{n} x^{n} &= \sum_{d\geq 0} dr_d \sum_{n=1}^{\infty}x^{nd} \\
px+p^2x^2+\cdots &= \sum_{d\geq 0} dr_d(x^d+x^{2d}+\cdots) \\
px+p^2x^2+\cdots &= \sum_{d\geq 0} (dr_dx^d+dr_dx^{2d}+\cdots) \qquad (6)
\end{aligned}
$$

For each term in the right-hand side of (6), dr_d is added to its coefficient if and only if the degree of the term is a multiple of d, because for each $d \geq 0$, dr_d is added to $x^d, x^{2d}, \ldots$ on the right-hand side. Therefore, we have:

$$\begin{aligned}
px + p^2x^2 + \cdots &= \sum_{d|1} dr_d x + \sum_{d|2} dr_d x^2 + \cdots \\
\sum_{n=1}^{\infty} p^n x^n &= \sum_{n=1}^{\infty} \sum_{d|n} dr_d x^n \\
\sum_{n=1}^{\infty} p^n x^n &= \sum_{n=1}^{\infty} x^n \sum_{d|n} dr_d \\
p^n &= \sum_{d|n} dr_d \qquad (7)
\end{aligned}$$

which allows r_d to be calculated recursively using only r_b with $b < d$. (The numbers r_d can also be calculated in closed form using Möbius inversion.) □

In Lean, I have defined the sets of polynomials with degree d, monic polynomials of degree d, and irreducible monic polynomials of degree d respectively over any semiring as

```
variables (R : Type) [semiring R]

def degree_eq (d : ℕ) : set R[X] := { f | f.nat_degree = d }
def monic_degree_eq (d : ℕ) := degree_eq R d ∩ { f | f.monic }
def irreducible_degree_eq (d : ℕ) :=
  monic_degree_eq R d ∩ { f | irreducible f }
```

From these definitions, I refine to specifically finite fields $\mathbb{F}_{p^n}$

```
def monics (d : ℕ) : ℤ :=
    (#(monic_degree_eq (galois_field p n) d)).to_nat
def irreducibles (d : ℕ) :=
    (#(irreducible_degree_eq (galois_field p n) d)).to_nat
```

using `field_theory.finite.galois_field` from `mathlib`.

In order to prove Part 1 of my paper proof, I define an equivalence between `mathlib`'s existing submodule of polynomials over a semiring R with degree strictly less than d `degree_lt` and my `monic_degree_eq` set by bijectively erasing the leading coefficient of a monic polynomial to obtain an element in `degree_lt`, and conversely adding a monic term with degree exactly d to obtain an element of `monic_degree_eq`, and encapsulate the equivalence in

```
def monic_degree_lt_equiv {d : ℕ} : monic_degree_eq R d ≃ degree_lt R d
```

I proceed to prove the consequential lemma

```
lemma card_degree_lt (d : ℕ) : #(degree_lt F d) = #F ^ d
```

which, in turn, depends on very recently proved theorems in `mathlib`, like `cardinal_mk_eq_cardinal_mk_field_pow_rank` and `rank_span_set`: these results were not in `mathlib` when the current semester started! This shows that my current final project is on the active edge of `mathlib`'s reach.

The central result of Part 1 is fully proven transitively using `card_degree_lt`, from which the cardinality of the monic polynomials of degree d can be determined via `monic_degree_lt_equiv`, and a readable `calc` construction, as

```
  -- A(x) = 1/(1-QC)
  lemma monic_generating_function : mk (monics p n) = rescale ↑(p ^ (n : ℕ))
(inv_units_sub 1) :=
  calc mk (monics p n) = mk (pow ↑(p ^ (n : ℕ))) : -- A(x) = Σ_{i=0}^∞ q^i
x^i
    begin
      apply power_series.ext,
      simp only [coeff_mk],
      intro d,
      unfold monics,
      norm_cast,
      rw [cardinal.mk_congr (monic_degree_lt_equiv (galois_field p n)),
          card_degree_lt d],
      swap,
      { apply_instance },
      rw cardinal.mk_fintype,
      rw galois_field.card,
      swap,
      { simp only [ne.def, pnat.ne_zero, not_false_iff] },
      norm_cast,
      rw cardinal.to_nat_cast
    end
  ...                              = _ : -- Σ_{i=0}^∞ q^i x^i = 1/(1-qx)
    begin
      apply power_series.ext,
      simp only [coeff_mk, coeff_rescale, coeff_inv_units_sub, one_pow, one_divp,
int.units_inv_eq_self, units.coe_one, mul_one, eq_self_iff_true, forall_const]
    end
```

For Part 2 of my proof, an infinite product of power series is used; however, mathlib contains no such notion. Therefore, I rigorously defined such an infinite product in my `section infinite_product` based on power series convolution, or equivalently, an infinite Cauchy product of power series. The general, reusable, and rigorous proof of the infinite product and its related lemmas was a very intensive effort, which paved the way to a formalization of Part 2 as `monic_generating_function'`.

I formalized the statement of Part 3 as

```
lemma irreducibles_arithmetic_function :
  ∀ m > (0 : ℕ),
    (p ^ (n : ℕ)) ^ m = Σd in m.divisors, d * irreducibles p n d
```

Since Part 3 uses logarithmic differentiation over infinite series, which `mathlib` also has no support for, and very little time was left after defining all of `section infinite_product`, the logarithmic differentiation is left as `sorry`, though the equivalence which needs proving has been set up from `monic_generating_function'` and `monic_generating_function` previously.

Finally, the Möbius inversion of `irreducibles_arithmetic_function` is rigorously completed using `mathlib`'s existing canonical definition of the Möbius μ and divisor sums

```
-- r_d = (1/d)Σ_{y|d}μ(y)q^(d/y)
theorem irreducibles_closed_form
  (d : ℕ) (h : 0 < d)
  : ↑(irreducibles p n d) = (Σx in d.divisors_antidiagonal, μ x.fst * (p ^
(n : ℕ)) ^ x.snd) / d :=
begin
  let f : ℕ → ℤ := (λ d, d * irreducibles p n d),
  let g : ℕ → ℤ := pow (p ^ (n : ℕ)),
  suffices : Σx in d.divisors_antidiagonal, ↑(μ x.fst) * g x.snd = f d,
  { dsimp [f, g] at this,
    norm_cast at this,
    rw_mod_cast this,
    ring_nf,
    rw [nat.mul_div_assoc _ (dvd_refl d), nat.div_self h, mul_one] },
  apply nat.arithmetic_function.sum_eq_iff_sum_mul_moebius_eq.mp,
  swap,
  { exact h },
  intros d hd,
  dsimp [f, g],
  rw_mod_cast (irreducibles_arithmetic_function p n d hd)
end
```

My final project pushes the boundaries of what has been formalized in Lean, utilizes extremely recent additions to `mathlib`, and contributes reusable code for infinite Cauchy products of infinite series—an advanced topic which has nontrivial formalization. Though I have had to leave some parts as `sorry` because of the time the needed to formalize the infinite product and cardinality of the set of monic polynomials of given degree, I have made novel progress towards the formalization of this important result of abstract algebra and number theory.