MIT Primes Solutions

Adithya Prabha

November 2024

1 Problem 1

Problem One

Let $g : \mathbb{R} \to \mathbb{R}$ be a differentiable function satisfying the following conditions:

- $g(0) = 1$ and $g(t) \geq 0$ for all $t \in \mathbb{R}$.
- The derivative function $g' : \mathbb{R} \to \mathbb{R}$ is continuous.

Argue that the following inequality holds:

$$\left| \int_0^1 g(t)\,dt - \int_0^1 g(t)^3\,dt \right| \leq M \left(\int_0^1 g(t)\,dt \right)^2,$$

where M is the maximum value of $|g'(t)|$ in the closed interval $[0, 1]$.

We can start by dividing the inequality into two cases, based on the absolute value. The first case is when $\int_0^1 g(t)\,dt \geq \int_0^1 g(t)^3\,dt$ and the second case is when When $\int_0^1 g(t)\,dt \leq \int_0^1 g(t)^3\,dt$.

1.1 When $\int_0^1 g(t)\,dt \geq \int_0^1 g(t)^3\,dt$

When the above condition is true, we can remove the absolute value sign. We need to show the following

$$\int_0^1 g(t)\,dt - \int_0^1 g(t)^3\,dt \leq M \left(\int_0^1 g(t)\,dt \right)^2$$

We define a function $f(y)$. For this case, it suffices to show that $f(y) \geq 0$

$$f(y) = M \left(\int_0^y g(t)\,dt \right)^2 - \int_0^y g(t)\,dt + \int_0^y g(t)^3\,dt$$

the derivative of $f(y)$ with respect to y is:

$$f'(y) = M \cdot 2 \left(\int_0^y g(t)\,dt \right) g(y) - g(y) + g(y)^3.$$

From the problem statement, we know that $g(y) \geq 0$ on the interval so we can essentially remove it from $f'(y)$ as it won't affect the greater than 0 condition. Making it,

$$M \cdot 2 \left(\int_0^y g(t)\,dt \right) - 1 + g(y)^2.$$

Taking the derivative again which finally gets rid of integral leads us to,

$$M \cdot 2g(y) + 2g(y)g'(y) = 2g(y) \cdot (M + g'(y))$$

Because $g(y) \geq 0$ and by the definition of M, we know that "parts" of the product are positive so the product must be positive as well. Thus, we know

$$2g(y) \cdot (M + g'(y)) \geq 0$$

1.2 When $\int_0^1 g(t)\,dt \leq \int_0^1 g(t)^3\,dt$

In this case, we can remove the absolute value sign but multiply the LHS by -1.

$$f(y) = M \left(\int_0^y g(t)\,dt \right)^2 + \int_0^y g(t)\,dt - \int_0^y g(t)^3\,dt$$

$$f'(y) = M \cdot 2\left(\int_0^y g(t)\,dt\right) \cdot g(y) + g(y) - g(y)^3$$

We can do the same and remove the $g(y)$

$$2M\left(\int_0^y g(t)\,dt\right) + 1 - g(y)^2$$

Taking the derivative,

$$2M \cdot g(y) - 2g(y)g'(y) = 2g(y) \cdot (M - g'(y))$$

Once again, we know that $M \geq g'(y)$ by definition and that $g(y) > 0$ from the problem statement. Thus, we can conclude that $2g(y) \cdot (M - g'(y)) \geq 0$.

Remarks: I enjoyed this question. My inital attempt tried to make use of M by some form of substitution such as using MVT. This didn't work well, so I tried keeping M and eventually thought of "operating" on the inequality as a whole rather than dividing it into individual parts. That idea motivated the solution above.

2 Problem 2

> Problem Two
>
> Consider a chessboard of length 12 (with 144 unit squares). Two distinct unit squares of the chessboard are called *adjacent* if they share an edge. Find the largest $m \in \mathbb{N}$ such that whenever we mark the $2m$ unit squares covered by any m disjoint pairs of adjacent units, there are still two adjacent unit squares that remain unmarked.

We first aim to prove the following general claim: if there are no two adjacent unmarked unit squares, then the chessboard has at least $\frac{1}{3}s^2$ marked adjacent squares. I would like to think of two adjacent adjacent marked squares as a single unit rather than two separate units that are adjacent.

We will start by talking about the idea ray pointing. For every unmarked square in the grid, we release 4 rays starting from the center of the empty square such that it eventually meets a the perimeter of a "two adjacent marked unit squares". It's only function is to hit a "two adjacent marked unit squares", so if it doesn't, it won't exist (which is obviously when it is on the border of the chessboard).

Throughout this proof, I will use the variable M for the number of "two adjacent unmarked unit squares" in the chessboard. First, we can start by defining some variables that I will use in this proof:

1. Let A be the number of unmarked squares in the grid that are not on the border.
2. Let B be the number of unmarked squares that are on the border but aren't the corner.
3. Let C be the number of unmarked squares that are on the corner of the grid.
4. Let E be the number of adjacent marked unit squares that are not on the border.
5. Let F be the number of adjacent marked unit squares that are on the border.
6. Let G be the number of adjacent marked unit squares that contain a corner cell.

From these definitions, it follows that

$$M = E + F + G.$$

The total number of unmarked squares in total number of squares minus the total number of adjacent marked squares. So we get,

$$s^2 - 2M = A + B + C.$$

4rAdditionally, each of the 4 corners is occupied by either a marked or unmarked square.

$$C + G = 4.$$

Furthermore, we can establish that:

$$B + C \leq F + G.$$

We can get this inequality by noting that in order to ensure that there are no adjacent unmarked cells on the chessboard, there must be an alternating pattern around the border. This means that for every unmarked cell, there must be a marked square essentially paired with it. Since all marked squares come in pairs, the above inequality is true.

Next, we'll count the number of rays in two different ways:

$$4A + 3B + 2C \leq 4E + 3F + 2G.$$

Substituting $A = s^2 - 2M - B - C$,

$$4(s^2 - 2M) - B - 2C \leq 4M - F - 2G.$$

We said earlier that $B + C \leq F + G$, so we can make the following substitution on the RHS:

$$4(s^2 - 2M) - (B + C) - C \leq 4M - (B + C) - G$$

This leads to:

$$4(s^2 - 2M) \leq 4M + (C - G).$$

Since $C + G = 4$ and both C and G are non-negative, $C - G \leq 4$. Substituting this in:

$$4(s^2 - 2M) \leq 4M + 4.$$

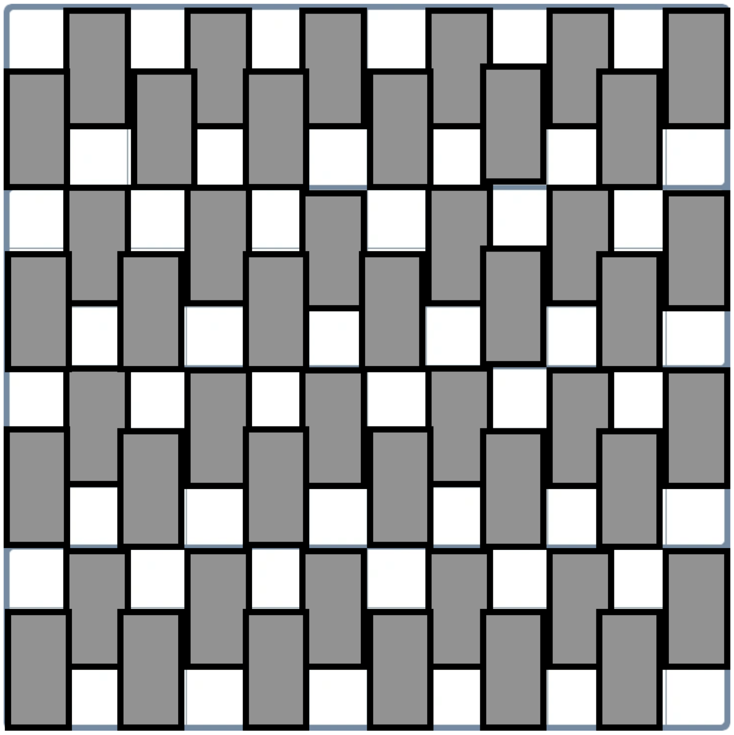
Dividing both sides by 4:

$$s^2 - 2M \leq M + 1.$$

$$\frac{s^2 - 1}{3} \leq M.$$

This proves our original claim that if there are no adjacent unmarked cells, then the chessboard has at least $\frac{1}{3}s^2$ adjacent marked cells. For our specific case where $s = 12$, we have the following: if there are no adjacent unmarked squares, then the chessboard has at least 48 pairs of unmarked cells. Thus we have the contrapositive of the statement: if there are less than 48 pairs of unmarked cells on the grid, then are adjacent unmarked squares.

Now what remains is to show that if there are 48 pairs of marked squares, then there aren't two adjacent unmarked cells. Below, I have a construction on the 12×12 grid. Thus, we have completed the proof.



Remarks: I had fun doing this problem. The idea of the ray pointing is an idea I have seen before on an Olympiad combinatorics question. If I am remembering correctly, I have also seen a similar question to this in the past, which motivated me to format my solution the way it is.

3 Problem 3

Problem Three

For $n \in \mathbb{N}$, let $b = b_1 b_2 \ldots b_n$ be a binary string with $b_1 + b_2 + \cdots + b_n \geq 1$ (that is, $b_i = 1$ for at least one index i), and let $v = (c_1, c_2, \ldots, c_n)$ be a value vector with entries in $\mathbb{N}$. An *improvement operation* on the value vector v with respect to the binary string b consists of the following two steps:

1. Choose an index i from the set $\{1, 2, \ldots, n\}$ with probability $\frac{c_i}{c_1+c_2+\cdots+c_n}$
2. For the chosen index i, replace c_i by: $c_i - 1$, if $b_i = 0$ and $c_i + 1$ if $b_i = 1$.

Design an efficient algorithm that computes the expected value of each entry of the vector v after m improvement operations, and explain the worst time complexity of your algorithm in terms of m and n.

When solving this question, I used ChatGPT to formulate some ideas on how to tackle parts of this question. I also used it to fix some of the bugs in my code when necessary. Originally, I had exponential time complexity, and I somewhat optimized it to a high polynomial degree complexity. I realized I could use Dynamic Programming and then optimized it to $O(m^3 + n)$ and ultimately brought it down to $O(m^2 + n)$.

My solution is a dynamic programming algorithm that uses $O(m^2 + n)$.

The first thing we want is to obviously find the initial sum of all c_i such that their corresponding $b_i = 1$ which we let be sum_{b_1}, and we find the initial sum of all c_i such that their corresponding $b_i = 0$ and we let this be sum_{b_0}.

Rather than working through each individual element in the c array, it suffices to only work on sum_{b_1} and sum_{b_0}. This is because for each individual element with corresponding $b[i] = 1$, the ratio between the expected value after m improvement operations and the original $c[i]$ is constant. It is also constant (a different constant) for when $b[i] = 0$. With the ratio in the former being > 1 and the ratio in the latter being < 1. The way I discovered this was using my exponential code, and dividing I saw the ratio being constant for each m increment. Using this approach greatly increases efficiency because we no longer need to worry about the state of each individual element. I will prove it mathematically, here

3.1 Ratio Claim Proof

We can prove the ratios of the expected value of two c_i, c_j after $m \geq 1$ improvement operations where $b_i = b_j$ is the same as the ratio of c_i, c_j. (WLOG), let $b_i = b_j = 1$; a similar proof holds for $b_i = b_j = 0$. We can do this by induction on m.

The base case is $m = 1$. Let $S = \sum_{t=1}^{n} c_t$, and we see that the expected value of c_i after one improvement step is simply $c_i + c_i/S$. Similarly, the expected value of c_j after one step is $c_j + c_j/S$. Since they are both the same multiple $(1 + 1/S)$ of their original values, their ratio remains the same.

For the inductive step, assume the hypothesis for m, and we prove it for $m + 1$. Let a_2 be the value of c_i after $m+1$ improvement rounds, and a_1 be the value of c_i after k improvement rounds. Similarly, define b_2 and b_1 for c_j. We wish to prove that

$$\frac{\mathbb{E}(a_2)}{\mathbb{E}(b_2)} = \frac{a}{b} = \frac{\mathbb{E}(a_1)}{\mathbb{E}(b_1)},$$

where the second equality is given by the induction hypothesis.

Then we write $\mathbb{E}(a_2)$ and $\mathbb{E}(b_2)$ as follows:

$$\mathbb{E}(a_2) = \sum_{x} P(a_1 = x)\mathbb{E}(a_2 \mid a_1 = x),$$

$$\mathbb{E}(b_2) = \sum_x P(b_1 = x)\mathbb{E}(b_2 \mid b_1 = x).$$

Here, $\mathbb{E}(P \mid Q)$ denotes the conditional expected value of P, given that Q is true. The sum ranges over all $x \in \mathbb{N}$, and similarly for y in the equations below.

Using this, let us prove our desired equality.

$$\mathbb{E}(a_2)\mathbb{E}(b_1) = \mathbb{E}(a_1)\mathbb{E}(b_2).$$

Substituting, we get,

$$\left(\sum_x P(a_1 = x)\mathbb{E}(a_2 \mid a_1 = x)\right) \cdot \mathbb{E}(b_1) = \left(\sum_x P(b_1 = x)\mathbb{E}(b_2 \mid b_1 = x)\right) \cdot \mathbb{E}(a_1).$$

Using the definition of expected value:

$$\mathbb{E}(b_1) = \sum_y P(b_1 = y) \cdot y, \quad \mathbb{E}(a_1) = \sum_y P(a_1 = y) \cdot y.$$

Substituting these into the equation gives,

$$\begin{aligned} &\left(\sum_x P(a_1 = x)\mathbb{E}(a_2 \mid a_1 = x)\right) \cdot \left(\sum_y P(b_1 = y) \cdot y\right) \\ &= \left(\sum_x P(b_1 = x)\mathbb{E}(b_2 \mid b_1 = x)\right) \cdot \left(\sum_y P(a_1 = y) \cdot y\right). \end{aligned}$$

Expanding both sides,

$$\sum_{x,y} P(a_1 = x)\mathbb{E}(a_2 \mid a_1 = x)P(b_1 = y) \cdot y,$$

$$\sum_{x,y} P(b_1 = x)\mathbb{E}(b_2 \mid b_1 = x)P(a_1 = y) \cdot y.$$

To prove equality, consider the (x, y) terms,

$$P(a_1 = x)\mathbb{E}(a_2 \mid a_1 = x)P(b_1 = y) \cdot y = P(b_1 = y)\mathbb{E}(b_2 \mid b_1 = y)P(a_1 = x) \cdot x.$$

Rearranging, all we need to do know is prove the following:

$$\frac{\mathbb{E}(a_2 \mid a_1 = x)}{x} = \frac{\mathbb{E}(b_2 \mid b_1 = y)}{y}.$$

This follows directly from the base case, completing the induction.

We now use the ratio idea to construct our DP Table as well as our code.

3.2 DP Table

In the code above, the DP Table models the probability distribution of states after each improvement operation. We let the nth element in the dp-table be the probability that sum_{b_1} was incremented n times. The array is initialized with size $m + 1$ because the number of times sum_{b_1} can be between 0 and m. We let the 0th index be 1 because the probability it is chosen 0 times in 0 iterations is 1, and the remaining indices are set at 0 and will be computed later on in the nested for loop.

```
dp_table = np.zeros(m + 1)
dp_table[0] = 1.0
```

We then create a temporary table in the first loop. The first loop essentially iterates through each improvement operation.

```
temp_dp_table = np.zeros(m + 1)
```

We now move onto the inner loop where we iterate through all possible counts of operations that have already played a role in sum_{b_1}. Thus, our inner loop only goes up to the m value we are on. The inner loop computes the probability that the new operation affects either sum_{b_1} or sum_{b_0}. After the given improvement operation, we must determine the probability that the next operation will sum_{b_1} and similar reasoning for sum_{b_0}. Now we transition into the the next iteration of the inner loop, using our already calculated probabilities:

```
temp_dp_table[selected_count + 1] += dp_table[selected_count] * prob_sum_b1
```

We also know that the expected sum of group G_1 after m operations is given by:

$$\text{expected\_sum\_b}_1 = \sum_{k=0}^{m} \text{dp\_table}[k] \cdot (\text{sum\_b1} + k)$$

$$\text{expected\_sum\_b}_0 = \sum_{k=0}^{m} \text{dp\_table}[k] \cdot (\text{sum\_b1} - (m - k))$$

It is reflected in the code,

```
expected_sum_b1 = np.sum(dp_table * (sum_b1 + k))
expected_sum_b0 = np.sum(dp_table * (sum_b0 - (m - k)))
```

These by definition make sense.

Now the last part of our code is the scaling idea. This was something I originally didn't use; however, I eventually put it in. The scaling is essentially the same ratio idea that I brought up earlier in this writeup. The rest of the code follows:

```
import numpy as np

def expected_val(c, b, m):
    c = np.array(c, dtype=float)
    b = np.array(b, dtype=int)
    num_values = len(c)

    sum_b1 = np.sum(c[b == 1])
    sum_b0 = np.sum(c[b == 0])
    total = sum_b1+sum_b0

    dp_table = np.zeros(m + 1)
    dp_table[0] = 1.0

    for m_increment in range(m):
        temp_dp_table = np.zeros(m + 1)
        for selected_count in range(m_increment + 1):
            current_sum_b1 = sum_b1 + selected_count
            current_sum_b0 = sum_b0 - (m_increment - selected_count)
            current_total = current_sum_b1 + current_sum_b0


            prob_sum_b1 = current_sum_b1 / current_total
            prob_sum_b0 = current_sum_b0 / current_total

            if selected_count + 1 <= m:
                temp_dp_table[selected_count + 1] += dp_table[selected_count]*prob_sum_b1
            temp_dp_table[selected_count] += dp_table[selected_count] * prob_sum_b0

        dp_table = temp_dp_table

    k = np.arange(m + 1)
    expected_sum_b1 = np.sum(dp_table * (sum_b1 + k))
    expected_sum_b0 = np.sum(dp_table * (sum_b0 - (m - k)))

    expected_values = np.zeros(num_values, dtype=float)

    ratio_1 = expected_sum_b1 / sum_b1 if sum_b1 != 0 else 1.0
    ratio_0 = expected_sum_b0 / sum_b0 if sum_b0 != 0 else 1.0

    expected_values[b == 1] = c[b == 1] * ratio_1
    expected_values[b == 0] = c[b == 0] * ratio_0

    return expected_values
```

Remarks: I did multiple test cases by hand to be confident that my exponential code was accurate. After I was confident it was accurate, I used it to check my solutions for my dp for $O(m^3 + n)$ and I eventually got it down to $O(m^2 + n)$. I found this to be an interesting coding question. I did the proof for the ”ratio” lemma quite last minute actually, and now I after writing it up, I think there may be a better algorithm to solve this question. This is a question I will attempt to answer over the next few days after the deadline.

4 Problem 4

Problem Four A and B

Consider the following sample properties:

Let $\mathbb{F}_2$ be the field of two elements, and let $\mathbb{F}_2[x]$ be the ring of polynomials over $\mathbb{F}_2$. For each $n \in \mathbb{N}$, consider the following polynomial in $\mathbb{F}_2[x]$:

$$f(x) = x^{6n} + x^{5n} + x^{4n} + x^{3n} + 1.$$

(a) For which values of n does $f(x)$ factor into exactly two irreducible polynomials in $\mathbb{F}_2[x]$?

(b) For which values of n does $f(x)$ factor into exactly three irreducible polynomials in $\mathbb{F}_2[x]$?

We start by factoring $f(x)$ in $\mathbb{F}_2[x]$. We can group terms and factor as we see below

$$f(x) = x^{6n} + x^{5n} + x^{4n} + x^{3n} + 1 = (x^{6n} + x^{5n} + x^{4n}) + (x^{3n} + 1)$$

$$= x^{4n} \cdot (x^{2n} + x^n + 1) + (x^n + 1)(x^{2n} - x^n + 1)$$

Because we are in $\mathbb{F}_2[x]$, we have $x^{2n} - x^n + 1 = x^{2n} + x^n + 1$.

$$= x^{4n} \cdot (x^{2n} + x^n + 1) + (x^n + 1)(x^{2n} + x^n + 1)$$

Grouping terms,

$$f(x) = (x^{2n} + x^n + 1)(x^{4n} + x^n + 1)$$

This is a factored form of $f(x)$. For part a, we need to find all n such that both $x^{2n} + x^n + 1$ and $x^{4n} + x^n + 1$ are irreducible in $\mathbb{F}_2$. I will first find all values of n that make only $x^{2n} + x^n + 1$ irreducible. Then I will find all values of n that make $x^{4n} + x^n + 1$ irreducible.

4.1 Finding all n s.t. $x^{2n} + x^n + 1$ is irreducible

I originally used Alpertron to try values and test conditions for irreducibility on the polynomial. This leads me to claim that for $x^{2n} + x^n + 1$ is irreducible if and only if n is of the form 3^k for $k \geq 0$.

4.1.1 When $3 \not| n$

$\Phi_3(x) = \frac{x^3-1}{x-1} = x^2 + x + 1 = (x - \omega)(x - \overline{\omega})$, where $\omega = e^{\frac{2\pi i}{3}}$. We can do casework on $n \pmod 3$ to see whether or not ω is a root of $x^{2n} + x^n + 1$. If

1. $n \equiv 1 \pmod 3$, $w^{2n} + w^n + 1 = 0$
2. $n \equiv 2 \pmod 3$, $w^{2n} + w^n + 1 = 0$
3. $n \equiv 0 \pmod 3$, $w^{2n} + w^n + 1 = 3$

From this we, know that for all n s.t. $3 \not| n$, $(x^2 + x + 1)|(x^{2n} + x^n + 1)$. However, we don't know anything yet about when $3|n$. We further divide this into 2 subcases: when n is a power of 3 and when is not a power of 3 but is divisible by 3.

4.1.2 When $n \equiv 0 \pmod 3$ but $3^{v_3(n)} \neq n$

For the latter case, let k be the greatest power of 3 factor of n, so $k = v_3(n)$. I claim that $(x^{2(3^k)} + x^{3^k} + 1)|(x^{2n} + x^n + 1)$. Proving this is essentially follows a similar case as the the one above. First we can represent $x^{2n} + x^n + 1$ as $x^{2m \cdot 3^k} + x^{m \cdot 3^k} + 1$. Also, the roots of the polynomial $x^{2(3^k)} + x^{3^k} + 1$ are the 3^{a+1}th primitive roots of unity so it will always be of the form $e^{\frac{2a\pi i}{3^{k+1}}}$. Plugging it into $x^{2m \cdot 3^k} + x^{m \cdot 3^k} + 1$, we get $e^{\frac{4am\pi i}{3}} + e^{\frac{2am\pi i}{3}} + 1$. This is where it kind of ties back to the previous case. First of all, by definition 3 does not divide m. Also, 3 does not divide a because if it would, it would no longer be a primitive root of unity. Thus, we can divide the case into two cases: $am \equiv 1 \pmod 3$ and $am \equiv 2 \pmod 3$. This takes us exactly back to the cases tested in the previous part, so we know that both cases work. Thus, we can conclude that $(x^{2(3^{v_3(n)})} + x^{3^{v_3(n)}} + 1)|(x^{2n} + x^n + 1)$.

Case 4.1.2 and 4.1.1 imply that when n is not a power of 3, then it is not irreducible. Thus, we can conclude that if $x^{2n} + x^n + 1$ is irreducible, then n is a power of 3.

4.1.3 When $n \equiv 0 \pmod 3$ and $3^{v_3(n)} = n$

I use this Wikipedia link for the theorem below.

Theorem

Over a finite field with a prime number p of elements, for any integer n that is not a multiple of p, the cyclotomic polynomial Φ_n is irreducible if and only if p is a primitive root modulo n.

When $n = 3$, we can see that 2 is a primitive root of 3 so, $x^2 + x + 1$ is irreducible in $F_2[x]$. However, we want to generalize this for 3^n. We can do so by proving that 2 is a primitive root for all powers of 3.

> To do so, we have to show that $\text{ord}_{3^k} 2 = \phi(3^k) = \frac{2}{3} \cdot 3^k = 2 \cdot 3^{k-1}$. We can show this statement is true by considering the factors of the order. If $2^{2 \cdot 3^{k-1}} \equiv 1 \pmod{3^k}$, factors of $2 \cdot 3^{k-1}$ could still be the order, so we must check if the order could be them. Because the only prime factors are 2 and 3, it suffices to show that $2^{\frac{2 \cdot 3^{k-1}}{2}} \neq 1$ and $2^{\frac{2 \cdot 3^{k-1}}{3}} \neq 1$ as all other factors are factors of these. Using the binomial theorem on these expressions, we get
>
> 1. $2^{\frac{2 \cdot 3^{k-1}}{2}} = 2^{3^{k-1}} \equiv (3-1)^{3^{k-1}} = -1 \pmod{3^k}$
> 2. $2^{\frac{2 \cdot 3^{k-1}}{3}} = 2^{2 \cdot 3^{k-2}} \equiv (3-1)^{2 \cdot 3^{k-2}} = 3^{k-1} + 1 \pmod{3^k}$

Lastly, it is easy to see that $2^{2\cdot 3^{k-1}} \equiv (3-1)^{2\cdot 3^{k-1}} = 1 \pmod{3^k}$ making $\text{ord}_{3^k}2 = 2\cdot 3^{k-1}$ which is equal to $\phi(3^k)$ as calculated earlier. This shows that 2 is a primitive root of 3^k.

Now we have the fact that 2 is a primitive root of 3^k for all $k \geq 1$. Further, we know that $x^{2\cdot 3^k} + x^{3^k} + 1 = \Phi_{3^{k+1}}$. Using these 2 facts and the theorem, we can conclude that above, we can conclude that $x^{2\cdot 3^k} + x^{3^k} + 1$ is irreducible in $\pmod 2$.

Thus, we have that:

$$\underline{x^{2n} + x^n + 1 \text{ is irreducible if and only if } n = 3^k \text{ for } k \geq 0.}$$

4.2 Finding all n s.t. $x^{4n} + x^n + 1$ is irreducible

After trying some values using Alpertron and Wolfram, I believe that $x^{4n} + x^n + 1$ is irreducible when n is in the form $3^{k_1} \cdot 5^{k_2}$ with $k_1 \geq 0$ and $k_2 \geq 0$.
The following theorem is from this textbook, specifically Theorem 3.2.5.

Theorem

Consider the following sample properties: Let t be a positive integer and $P(z) \in \mathbb{F}_q[z]$ be irreducible of degree n and exponent e (equal to the order of any root of $P(x)$). Then $P(z^t)$ is irreducible over $\mathbb{F}_q$ if and only if

(i) $\gcd(t, (q^n - 1)/e) = 1$,

(ii) each prime factor of t divides e, and

(iii) if $4 \mid t$, then $4 \mid (q^e - 1)$.

We can verify through bashing through polynomials with degree less than or equal to 2 that $x^4 + x + 1$ is irreducible. We can also verify that the smallest n such that $x^4 + x + 1$ divides $x^n - 1$ is 15. Thus, we know that $x^4 + x + 1$ is a primitive polynomial. Because it is primitive, we can use the fact that a primitive polynomial of degree m has m different roots in GF(p^m) which all have order $p^m - 1$. What this means is that any of the roots generate the multiplicative group of the field. We can get from here that the order of any root in $P(x)$ is equal to 15.

Simple computation for part i shows us that $(q^n - 1)/e) = 1$. Thus, the only case we need to worry about is the second case. For t to have the same prime factors as e, it must only have prime factors 3 and 5 and at any powers. Thus, we have that $x^{4n} + x^n + 1$ is irreducible only when $n = 3^a \cdot 5^b$. We also have both directions from the proof, so we have finished.

It is easy to notice that $x^4 + x + 1$ has order 15. Thus, $(x^{\cdot 3^{k_1}\cdot 5^{k_2}})^4 + (x^{\cdot 3^{k_1}\cdot 5^{k_2}}) + 1$ has order $15 \cdot 3^{k_1} \cdot 5^{k_2}$ which we can simplify to $3^{k_1+1} \cdot 5^{k_2+1}$. Based on the proofs we did earlier, we know the $\text{ord}_{3^{k_1+1}}2 = 2\cdot 3^{k_1}$ and similarly $\text{ord}_{5^{k_1+1}}2 = 4 \cdot 5^{k_2}$. To have the order for $\pmod{3^{k_1} \cdot 5^{k_2}}$, we take the lcm of our orders getting usm $4 \cdot 3^{k_1} \cdot 5^{k_2}$. This is also the degree of our original polynomial and thus it must be the minimal polynomial.

4.3 Part A

For part a, given the fact that $x^{2n} + x^n + 1$ is irreducible when $n = 3^a$ where $a \geq 0$, and that $x^{4n} + x^n + 1$ is irreducible when $n = 3^a \cdot 5^b$ with $a, b \geq 0$. Based on the claims and their proofs in 4.1 and 4.2, we can conclude that the polynomial $x^{6n} + x^{5n} + x^{4n} + x^{3n} + 1$ is factors into two irreducible polynomials only when $n = 3^k$ for $k \geq 0$.

4.4 Part B

I claim that it is not possible for $f(x) = x^{6n} + x^{5n} + x^{4n} + x^{3n} + 1$ cannot be factored into 3 irreducible factors. The only way for $x^{6n} + x^{5n} + x^{4n} + x^{3n} + 1$ to factor into 3 irreducible polynomials is when one of $x^{2n} + x^n + 1$ or $x^{4n} + x^n + 1$ is irreducible and the other reduces into 2 irreducible polynomials. It is important here to note that whenever $x^{2n}+x^n+1$ is irreducible, so is $x^{4n}+x^n+1$. Thus, we need it the other way: we need $x^{4n}+x^n+1$ to be irreducible and $x^{2n}+x^n+1$ to reduce into 2 irreducible polynomials. Thus would mean that n is of the form $3^a \cdot 5^b$ where $b \geq 1$.

The first thing I did was factored the following two polynomials in F_2 and found a general trend and use this for my construction

$$x^{10}+x^5+1=(x^2+x+1)(x^4+x+1)(x^4+x^3+1)$$
$$x^{30}+x^{15}+1=(x^6+x^3+1)(x^{12}+x^3+1)(x^{12}+x^9+1)$$

Using this, we can construct a general trend. Let $w=3^a \cdot 5^{b-1}$.

$$x^{10w}+x^{5w}+1=(x^{2w}+x^w+1)(x^{4w}+x^{3w}+1)(x^{4w}+x^w+1)$$

Expanding the RHS, we can see that it equals the LHS. Thus, we see that $f(x)$ can never be factored into 3 irreducible polynomials in F_2.

Remarks: The $x^{2n}+x^n+1$ case was rather straightforward, but the $x^{4n}+x^n+1$ case was quite complicated. I cited a theorem for it however, I was unable to prove it without than copy majority of the proof, so I didn't prove it. I liked the ending of this question.

5 Problem 5

Problem Five

The special linear group $\mathrm{SL}(2,\mathbb{Z})$ over $\mathbb{Z}$ is the multiplicative group consisting of all 2×2 matrices with entries in $\mathbb{Z}$ and determinant 1; that is,

$$\mathrm{SL}(2,\mathbb{Z})=\left\{\begin{pmatrix} a & b \\ c & d \end{pmatrix} : a,b,c,d \in \mathbb{Z} \text{ and } ad-bc=1\right\}.$$

Let G be the quotient group $\mathrm{SL}(2,\mathbb{Z})/\{\pm I\}$, where I is the 2×2 identity matrix. Find, with proof, the number of subgroups of G of index m for each $m \in \{2,3,4,5,6\}$.

For this question, I will be citing multiple papers for a general formula for the number of subgroups of specific index in $\mathrm{SL}(2,\mathbb{Z})\ \{\pm I\}$. I will also share progress that I have made relating to doing the first few indices by hand. After doing some research into what the quotient group really means, I came across this website about $\mathrm{PSL}(2,\mathbb{Z})$. It states that $\mathrm{SL}(2,\mathbb{Z})\{\pm I\} \cong \mathrm{PSL}(2,\mathbb{Z}) \cong C_2 * C_3$. With the last term being a free product of two cyclic groups: one of order 2 and one of order 3.

5.1 Proof of $\mathrm{SL}(2,\mathbb{Z})\{\pm I\} \cong C_2 * C_3$

By definition, we have $\mathrm{SL}(2,\mathbb{Z})\{\pm I\} \cong \mathrm{PSL}(2,\mathbb{Z})$. So to prove $\mathrm{SL}(2,\mathbb{Z})\{\pm I\} \cong C_2 * C_3$, it is sufficient to prove that $\mathrm{PSL}(2,\mathbb{Z}) \cong C_2 * C_3$

5.1.1 Proof that $\mathrm{PSL}(2,\mathbb{Z}) \cong C_2 * C_3$

I claim that the generators are:

$$S=\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \quad \text{and} \quad T=\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

It is easy to verify that $S,T \in \mathrm{SL}(2,\mathbb{Z})$, so we know that $\langle S,T \rangle$ is a subgroup of $\mathrm{SL}(2,\mathbb{Z})$. It is also important to note that $T^n=\begin{pmatrix} 1 & n \\ 0 & 1 \end{pmatrix}$. We will now consider the effect on a random matrix $M=\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ in $\mathrm{SL}(2,\mathbb{Z})$ of S and T. .

$$S\begin{pmatrix} a & b \\ c & d \end{pmatrix}=\begin{pmatrix} -c & -d \\ a & b \end{pmatrix}, \quad T^n\begin{pmatrix} a & b \\ c & d \end{pmatrix}=\begin{pmatrix} a+nc & b+nd \\ c & d \end{pmatrix}$$

The form for the top entries in the matrix lays down the path to use the division algorithm repeatedly to reduce one matrix entry all the way to 0.

We now consider the case where $|a| > |c|$. If this is not the case, you can do an "S-operation" on the matrix to get it into the correct form.

We can apply T^{-n} to our matrix M. Using the division algorithm, we have $a = p \cdot c + r$. So now, we can apply the T-operation on M getting us,

$$T^{-n}M = \begin{pmatrix} a - qc & b - qd \\ c & d \end{pmatrix}$$

By the division algorithm, $a - q \cdot c < |c|$, and because of this, we must perform an S-operation on our matrix in order to put the value with the greater absolute value on the top to re-perform the division algorithm. This cycle(of multiplying by S or multiplying by powers of T) needs to be continuously repeated until $r = 0$. By the division algorithm, we are guaranteed that repeatedly performing this cycle will eventually lead to the value being 0 and in this case the value would be the entry in the bottom left corner of the matrix.

Because we are multiplying M by S and T with all M, S, T being in $\mathrm{SL}(2, \mathbb{Z})$, our final matrix must be in $\mathrm{SL}(2, \mathbb{Z})$. After doing the operations above, we can get that the final matrix will be of the form:

$$\begin{pmatrix} \pm 1 & w \\ 0 & \pm 1 \end{pmatrix}$$

Where both ± 1s have the same sign. Observe that M, after the transformations are a power of T. This matrix is thus either T^n or $-T^{-n}$. Thus, we have that there is some matrix $m \in \mathrm{SL}(2, \mathbb{Z})$ such that $gM = \pm T^n$. Obviously, since $\mathrm{SL}(2, \mathbb{Z})$ is a group, we have that $T^n \in G$ so $M = \pm g^{-1}T^n$.

Now that we have the generators of $\mathrm{SL}(2, \mathbb{Z})$, it follows that the images of S and T generate $\mathrm{PSL}(2, \mathbb{Z})$. We let, R be the product of the generators,

$$R = ST = \begin{pmatrix} 1 & -1 \\ 1 & 0 \end{pmatrix}$$

Since, $T = RS^{-1}$, we know that $\mathrm{SL}(2, \mathbb{Z})$ can also be generated by S and R. This is true because earlier we had that any element in $\mathrm{SL}(2, \mathbb{Z})$ can be generated with S and T and now we have T in terms of R and S, so all elements generated by S and T can now be represented in terms of R and S (any their inverses obviously). Let the images of R and S be $\tilde{R}$ and $\tilde{S}$ respectively, we want to find the orders of $\tilde{R}$ and $\tilde{S}$ in $\mathrm{PSL}(2, \mathbb{Z})$. Note the following,

$$S = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \quad S^2 = -I = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$R = \begin{pmatrix} 1 & -1 \\ 1 & 0 \end{pmatrix}, \quad R^2 = \begin{pmatrix} 0 & -1 \\ 1 & -1 \end{pmatrix}, \quad R^3 = -I = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$$

Thus, the order of S is 2 and the order of R is 3 so consequently $\tilde{R}$ and $\tilde{S}$ have order 2 and 3 respectively in $\mathrm{PSL}(2, \mathbb{Z})$. It is enough to make $\mathrm{PSL}(2, \mathbb{Z})$ act on the set of irrational numbers:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot r = \frac{ar + b}{cr + d}$$

Using the set of irrational numbers idea, we have that

$$S : z \mapsto -\frac{1}{z}, R : z \mapsto \frac{z - 1}{z}, R^{-1} : z \mapsto \frac{1}{1 - z}.$$

Let P and N be the set of positive and negative irrational numbers in the same form. We can see that,

$$S(P) \subset N \text{ and } R^{\pm 1}(N) \subset P.$$

We now want to verify the alternating word idea which essentially translates to showing that:

For a word w made from $\langle S\rangle$ and $\langle T\rangle$, $w \neq 1$ for w in $\mathrm{PSL}(2,\mathbb{Z})$.

We can do this by considering two cases: whether w has odd or even length.

1. **If w has odd length**

   w both begins and ends with S

   w both begins and ends with $R^{\pm 1}$

2. **If w has even length**

   w begins with S ends in R

   w begins with S ends in R^{-1}

   w begins with R ends in S

   w begins with R^{-1} ends in S

Looking at the first case and first subcase, it is easy to see that $w(P) \subset N$ when it both begins and ends with B. Similarly, for the second subcase, we have $w(N) \subset P$. In both subcases, we see that $w \neq 1$.

Now we look at the second case for when w has even length. The first two subcases can be morphed into the second two cases by conjugating by S. Conjugating would necessarily make w start with one of $R^{\pm 1}$ and end with S. This is true because we have, for this case, w starts with S and ends with $R^{\pm 1}$. Conjugating adds to the beginning S^{-1} and adds to the end S. Thus, w would start with $S^{-1}S$ which equals 1. The next occurrence after S would necessarily have to be $C^{\pm 1}$. Also, W would end with S. Using this, we have morphed the first two cases into the second two cases. Thus, it's sufficient to look at just the bottom two cases. Looking at the 3rd and 4th case only, we have the following:

$$w(P) \subset R(N) \text{ and } w(P) \subset R^{-1}(N)$$

What this leaves us with is that for the former, $w(P)$ is a subset of irrationals greater than 1. For the latter, we have that $w(P)$ is a subset of all irrationals less than 1 looking back at the transformations earlier. We see once again that $w(1) \neq 1$.

Looking through the cases, we see that $w(1) \neq 1$ for all of them. Thus, we verified the alternating word property. What this means is that words with alternating elements do not reduce to the identity unless trivial.

This leaves us with $\mathrm{PSL}(2,\mathbb{Z}) = \langle S\rangle * \langle T\rangle$. Thus, $\mathrm{PSL}(2,\mathbb{Z})$ is the product of two cyclic groups since they are both generated by 1 element. Based on earlier computation, we can conclude

$$\mathrm{PSL}(2,\mathbb{Z}) = \langle S\rangle * \langle T\rangle$$

This proof was mapped after this paper's first section.

5.2 General Formula for Index N Subgroup

Originally, I found this paper with theorem 5.7 and corollary 5.8 exactly what we need. When I used it to compute values, it ultimately produced answers that were incorrect based on this oeis sequence I found. I went to the citations of this paper and found the original paper(Thm 6.10), and noticed there was a typo in the formula in the first paper.

Theorem 5.1

Let $G = *_{j\in J} A_j$, where J is a finite set $\{1, 2, \ldots, k\}$, and let $d_j^n (n > 0)$ be the number of homomorphisms of C_j into the symmetric group on n symbols, with $d_j^1 = 1$. Let N_n be the number of subgroups of G of index n. Then:

$$N_n = \frac{1}{(n-1)!}\prod_{j=1}^{k} d_j^n - \sum_{i=1}^{n-1} \frac{1}{(n-i)!}\prod_{j=1}^{k} d_j^i N_i.$$

I will sketch this proof out. Obviously, we know that $a_1(G) = 1$ as the only index one subgroup of a group G is itself.

By definition, there are d_j^n homomorphisms from A_j to the symmetric group S_n on n symbols. A permutation representation of G is formed by taking permutation representations of all A_j, denoted as:

$$\Pi(G) = \text{gp}\{\Pi(A_j) : j \in J\}.$$

The total number of permutation representations of G on n symbols is then:

$$\prod_{j \in J} d_j^n.$$

To find the number of subgroups of G of index n, we need the count of transitive permutation representations. A subgroup H of index n corresponds to such a transitive representation.

Let $\Pi(G)$ be a particular permutation representation of G on n symbols. The transitive part of the representation is determined by the transitivity class that includes 1, say $\{1, b_2, b_3, \ldots, b_i\}$. Let H be the subgroup of G of index i corresponding to this transitive representation. After having the index i, we have $n - i$ symbols remaining. What happens to the $n - i$ remaining symbols is irrelevant. After fixing the i symbols, there are still $n - i$ remaining and they can also form a representation of G. For each subgroup H of index i, the number of permutations is:

$$(n-1) \cdot (n-2) \ldots (n-i+1) \cdot \prod_{j=1}^{k} d_j^{n-i} = \frac{(n-1)!}{(n-i)!} \cdot \prod_{j=1}^{k} d_j^{n-i}$$

This, however, only accounts for 1 subgroup of 1 index. We need to go through all subgroups of all indices up to n. For a fixed index i, we have

$$\frac{(n-1)!}{(n-i)!} \cdot \prod_{j=1}^{k} d_j^{n-i} \cdot N_i$$

Summing over all indices, we get the total number of permutation representations of G on n symbols as

$$\sum_{i=1}^{n} \frac{(n-1)!}{(n-i)!} \prod_{j=1}^{k} d_j^{n-i} N_i.$$

Equating this to our original interpretation of the number of permutation representations, we have

$$\prod_{j=1}^{k} d_j^n = \sum_{i=1}^{n} \frac{(n-1)!}{(n-i)!} \prod_{j=1}^{k} d_j^{n-i} N_i.$$

In the summation on the right-hand side, the term corresponding to $i = n$ is: $(n-1)! \cdot N_n$. Rewriting the equation, we get,

$$\prod_{j=1}^{k} d_j^n = (n-1)! \cdot N_n + \sum_{i=1}^{n-1} \frac{(n-1)!}{(n-i)!} \prod_{j=1}^{k} d_j^{n-i} N_i.$$

Rearranging again,

$$(n-1)! \cdot N_n = \prod_{j=1}^{k} d_j^n - \sum_{i=1}^{n-1} \frac{(n-1)!}{(n-i)!} \prod_{j=1}^{k} d_j^{n-i} N_i.$$

Divide by $(n-1)!$:

$$N_n = \frac{1}{(n-1)!} \prod_{j=1}^{k} d_j^n - \sum_{i=1}^{n-1} \frac{1}{(n-i)!} \prod_{j=1}^{k} d_j^{n-i} N_i.$$

This is the original value in the theorem above. This proof was mapped after theorem 6.10 in this paper.

Theorem 5.2

Let $d_p^n = \#\mathrm{Hom}(C_p, \mathrm{Sym}(n))$ and let C_p be cyclic group of prime order p.

$$d_p^n = \sum_{0 \leq r \leq n/p} \frac{n!}{r!(n-pr)!p^r}$$

Proof: Permutations in S_n with order dividing p are those that have one of the two following: fixed points and p-cycles. We know there cannot be any other possibilities other than the two listed because if the cycle length was anything else, there would be an element whose order is not divisible by p. This uses the fact that p is prime because the cycles can obviously only be of size that is a factor of the size of the cyclic group. If it wasn't prime, the formula should be more complicated.

In order to count the number of permutations, consider how many p cycles there will be. We know the maximum number of p-cycles there can be is $\lfloor \frac{n}{p} \rfloor$ (all the p-cycles are disjoint). This is due to the fact that $\mathrm{Sym}(n)$ For a given number r number of p-cycles, the total number of fixed points is equal to $n - p \cdot r$. Thus, out of the n, we have $\binom{n}{pr}$ ways this can be done. Now we look at the pr elements chosen that need to be partitioned into r p-cycles. The number of ways this can be done is $\frac{(pr)!}{p^r \cdot r!}$. There are $(pr)!$ ways to arrange the $p \cdot r$ remaining elements. However, we have over-counted. Because each group is cyclic, we need to divide by p for each of the individual cycles. Further, the r p-groups are indistinguishable so that means we over counted by $r!$ as well. Thus we get

$$\binom{n}{pr} \cdot \frac{(pr)!}{r! \cdot p^r} = \frac{n!}{r! \cdot (n-pr)! \cdot p^r}$$

To get the total number of homomorphisms, we must sum over all possible values for the number of p-cycles. We know from earlier that it maxes out at $\lfloor \frac{n}{p} \rfloor$ because then we would exceed the number of elements. Using this reasoning, we reach

$$d_p^n = \sum_{0 \leq r \leq n/p} \frac{n!}{r!(n-pr)!p^r}$$

5.3 Extracting Values

We can use Theorem 5.2 on C_2 and C_3, getting us the below table:

k	$h_k(C_2)$	$h_k(C_3)$
1	1	1
2	2	1
3	4	3
4	10	9
5	26	21
6	76	81

For the formula below, I will be using the corrected version of the formula in the original paper I found. It is essentially the same as the one in Theorem 5.1, but I will be using it because it is easy to "parse through".

$$a_n(C_2 * C_3) = \frac{1}{(n-1)!} h_n(C_2) h_n(C_3) - \sum_{k=1}^{n-1} \frac{1}{(n-i)!} h_{n-k}(C_2) h_{n-k}(C_3) a_k(\mathrm{PSL}(2, \mathbb{Z}).$$

Using the values in the table as well as Theorem 5.1, we can recursively find $a_2(G), a_3(G), a_4(G), a_5(G), a_6(G)$. It is easy to know that $a_1(\mathrm{PSL}(2, \mathbb{Z}) = 1$ as it is the group itself, so we can continue the recursion until index 6.

n	a_n
2	1
3	4
4	8
5	5
6	22

5.4 Proof for index 2 Without Paper

We know that the kernel of the homomorphism from $C_2 * C_3$ to C_2 is a normal subgroup of index 2. Let a be the generator of C_2 and b be the generator of C_3, because they are both cyclic there is only 1 generator for both.

Since H is a normal subgroup of index 2, the quotient group G/H has order 2, which is isomorphic to C_2. Therefore, there exists a single surjective homomorphism:

$$\varphi : C_2 * C_3 \to C_2$$

The kernel of φ consists of all elements in $C_2 * C_3$ that map to the identity in C_2. Since φ is surjective, $\ker(\varphi)$ is the unique subgroup H of index 2.

$$\varphi(a) = 1 \quad \text{(the generator of } C_2\text{)}$$

$$\varphi(b) = 0 \quad \text{(the identity in } C_2\text{, since } b \text{ has order 3 and cannot map to a non-identity element in } C_2\text{)}$$

We know that in group homomorphisms, the order of an element must divide the order of its image. b has order 3 so $\varphi(b) = 0$. This would mean that all powers of b must be in the subgroup. This would mean that b is one of the generators. Similarly, we can get the second generator by conjugating b by a so our second generator would be aba^{-1}. I don't really have a good way to put this, but it would make sense that the conjugate works. Thus, the generators of the kernel or the subgroup will be $\langle b, aba^{-1} \rangle$. The way I originally thought about it was using the alternating word property was more or less that word are alternating a and a power of b, and the words generated by $\langle b, aba^{-1} \rangle$ allow that to happen.

Remarks: This was one of my favorite questions on the problem set because of the extensive learning and research I had to do. I think this problem ultimately took the longest for me. It took a while to research and understand the proofs of the theorems that I cited in this solution, including learning some basic representation theory. Also in general, these theorems originated around the 1960s and the proofs were either quite short and lacking detail or not there at all. However, for the proof that $\mathrm{PSL}(2, \mathbb{Z}) \cong C_2 * C_3$, I was happy to see a proof that used action on the set of irrationals rather than on the hyperbolic plane by Roger Alperin. For Theorem 5.2, there was no proof in any of the papers I looked into. There was only a citing to another paper that was restricted on the AMS website (if I remember correctly). This was my best attempt at recreating a potential proof after doing some research into the idea of symmetric and permutation groups.

6 Problem 6

Problem Six

Let $\mathbb{N}_0\langle x\rangle$ be the set of all nonzero generating functions

$$\sum_{n=0}^{\infty} a_n x^n$$

with coefficients in $\mathbb{N}_0$ (having at least one nonzero coefficient), and observe that $\mathbb{N}_0\langle x\rangle$ is closed under the standard multiplication of generating functions: for any $f(x) = \sum_{n=0}^{\infty} a_n x^n$ and $g(x) = \sum_{n=0}^{\infty} b_n x^n$ in $\mathbb{N}_0\langle x\rangle$,

$$f(x)g(x) = \left(\sum_{n=0}^{\infty} a_n x^n\right)\left(\sum_{n=0}^{\infty} b_n x^n\right) := \sum_{n=0}^{\infty}\left(\sum_{k=0}^{n} a_k b_{n-k}\right) x^n \in \mathbb{N}_0\langle x\rangle.$$

- A generating function $f(x) \in \mathbb{N}_0\langle x\rangle \setminus \{1\}$ is called *indecomposable* if whenever the equality

  $$f(x) = g(x)h(x)$$

  holds for some $g(x), h(x) \in \mathbb{N}_0\langle x\rangle$, either $g(x) = 1$ or $h(x) = 1$.

- A generating function $s(x) \in \mathbb{N}_0\langle x\rangle$ is called *supported* if either $s(x) = 1$ or $s(x)$ can be written as a product of finitely many indecomposable generating functions in $\mathbb{N}_0\langle x\rangle$ (repetitions of factors are allowed).

(a) Find a generating function in $\mathbb{N}_0\langle x\rangle$ that is not supported.

(b) Is it possible to find, for each $f(x) \in \mathbb{N}_0\langle x\rangle$, a supported generating function $s(x) \in \mathbb{N}_0\langle x\rangle$ such that $s(x)f(x)$ is also supported?

(c) Is it possible to find a supported generating function $s(x) \in \mathbb{N}_0\langle x\rangle$ such that $s(x)f(x)$ is a supported generating function for all $f(x) \in \mathbb{N}_0\langle x\rangle$?

6.1 Part A

In the following paper, I found a reference to the following generating function as not supported. We know that this generating function is in $\mathbb{N}_0\,[x]$ based on the conditions laid out in the question.

$$f(x) = \sum_{i=0}^{\infty} x^i$$

The paper uses the definition of a atomic semidomain, which the paper defines as a semidomain in which every nonzero element that is not a multiplicative unit factors into irreducible. Specifically, in example 3.3 we see that the semidomain of this power series in $\mathbb{N}_0\langle x\rangle$ isn't atomic.

Thinking about it, I believe without the use of the paper there would be a way to prove this using contradiction.

6.2 Part B

I was unable to come up with something useful to answer this question. I looked into properties of atomic and nearly atomic semidomains, but ultimately I couldn't make meaningful progress.

6.3 Part C

I see the similarity between this question and question 3.5 posed in the paper. It is described as an unsolved question.

Thank you for letting me have the opportunity to solve these problems! I learned a great deal solving some of these questions.