# Hello World!

Jeevman Banggore

May 3, 2024

# 1 INTRODUCTION

This is a test of vimtex as a medium for describing the simple unoptimized prime number finder in haskell

At end of the document some concepts will be pushed forward on how to optimize the process. I feel like vimtex holds advantage over spacemacs latex fragments when writing more dense documents. My current understanding of utility is using this to write highly structured pdfs. While spacemacs will be used when programming for mathematical deconstruction / comprehension. I will eventually make spacemacs have the same capabilities as this, but the write-to-disk speed is a massive bonus, and with minimal buffers nvim uses low memory

# 2 ASSUMPTIONS

1) A prime number is a number that is divisible by 1 and itself.
2) All even numbers are divisible by 2, due to symmetry, hence no primes can be even, as they would be divisible by 2.
3) Hence all primes lie in the odd domain.
4) As the multiples of a factor are divisible by that factor, no prime can be a multiple of an odd factor.
5) $\therefore$ computing odd multiples within the bound of the domain, and simply removing them from the domain results in the domain containing only prime numbers.
6) As the bound increases the amount of multiples present increases as odd factors also increase, therefore the amount of checks required increases by an exponential degree.

# 3  DOMAIN OF ODD NUMBERS

Let the domain of odd numbers be denoted using set builder/constructor notation as such;
Let B denote the bound.
Let $\tau$ denote the set containing odd numbers within the bound.

$$[\tau] = \left[ i \in [1...B] \mid \begin{cases} i+1 \mid i \mod 2 = 0 \\ i+2 \mid otherwise \end{cases} \right] \tag{1}$$

# 4  DOMAIN OF MULTIPLES

Conditional logic will be applied inorder to optimize what multiples are yielded, i.e a multiple of 3 is 6, but 6 is even and hence not prime.
Let the set of multiples be denoted $\sigma$.

$$[\sigma] = \left[ [m, \tau_i \in [2... \mid \tau \mid], \tau] \to x, y \mid \begin{cases} x*y \mid x*y <= \tau_{|\tau|} \\ \wedge \\ x*y \mid x*y \mod 2 \neq 0 \end{cases} \right] \tag{2}$$

# 5  EXCLUSION LOGIC

If a member of $\tau$ not in $\sigma$ then it is prime. Hence, let $\lambda'([a], [b])$ denote the conditional function that operates on two sets and evaluates the existance of $a \in b$.

$$[\lambda'([a], [b])] = \left[ [i \in a] \to i \mid \begin{cases} 1 \mid i \in b \\ 0 \mid i \notin b \end{cases} \right] \tag{3}$$

# 6  APPLICATION OF (3) to (2) and (1)

3,2,1 : )

Let [a] = $[\sigma]$ and [b] = $[\tau]$, hence $\lambda'([\sigma], [\tau])$.

$$\lambda'([\sigma], [\tau]) = \left[ [\sigma_i \in \sigma] \to \sigma_i \mid \begin{cases} 1 \mid \sigma_i \in [\tau] \\ 0 \mid \sigma_i \notin [\tau] \end{cases} \right] \tag{4}$$