2650BF4A2CCC2153	**Title**	**Conspectus 12023 Q1**
5EF8719D941FE2CC	**Subtitle**	**Chronical - V**
B355EADC9C77BFB7	**Author**	**Recursion Ninja**
54F0B88A5BD8E086	**Date**	**12023+079 Monday, March 20**
3019BF559A3D704A	**Word Count**	**1457 (ERT 6 min)**
3424453EA8281775	**Code Lines**	**0**
A1C7E69FCBC24878		
03E24368ACCA63F7	**Formats**	**.adoc .epub .html .markdown .txt**

Quarter 1: Winter Solstice to Vernal Equinox

Doctoral Program

Coursework has again formed the preponderance of my doctoral program this semester. For the spring 2023 semester I have enrolled in:

CSci 75100 – Logic in CS
CSci 80010 – Research Seminar
CSci 85011 – Distributed Computing
PHIL 76600 – First-Order Modal Logic

Enrollment in 13 credits is a great undertaking, especially while adamantly pushing for small progress of research objectives. Despite my heavy workload, I have achieved modest progress on developing supporting infrastructure for program synthesis research. This infrastructure will facilitate my anticipated direction of research during the summer.

TreeKEM Formal Verification

Multiple soundness preserving abstractions have been composed together. Additionally model encoding utilizing “bit-packing” and “bit-twiddling” techniques have been finalized. Subsequently, I dispatched multiple series of benchmarking to determine the optimal (with respect to scalability) compile-time directives and runtime flag provided by Spin to verify the newly encoded model.

Scaling of the new model encoding, under the metric of state vector $\vec{s}$ size in bytes, are astronomically superior to the results of my masters thesis. Recall that within the model exists a $(T,\ C,\ N)$ adversary $\mathcal{A}$ who attempts to glean some information from the TreeKEM cryptographic protocol shared by a group of up to N unique participants over at most T communication epochs, and utilizing at most C challenge

queries. The old model encoding $\mathcal{M}_{old}(T,\, N,\, \mathbb{P})$ from my masters thesis, verifies that $\mathbb{P}$ holds $\forall C \in [1,\, T] \subset \mathbb{N}$ the property $\mathbb{P}$ holds against a $(T,\, C,\, N)$ adversary $\mathcal{A}$. The new model encoding $\mathcal{M}_{new}(N,\, \mathbb{P})$ verifies that $\forall T \in \mathbb{N}$ and $\forall C \in [1,\, T]$ the property $\mathbb{P}$ holds against a $(T,\, C,\, N)$ adversary $\mathcal{A}$.

	$\mathcal{M}_{new}$		$\mathcal{M}_{old}$			$\mathcal{M}_{new}$		$\mathcal{M}_{old}$			$\mathcal{M}_{new}$		$\mathcal{M}_{old}$	
N	T	$\vec{s}$	T	$\vec{s}$	N	T	$\vec{s}$	T	$\vec{s}$	N	T	$\vec{s}$	T	$\vec{s}$
			4	$192\,B$				4	$224\,B$				4	$260\,B$
			5	$240\,B$				5	$296\,B$				5	$352\,B$
4	∞	$56\,B$	6	$248\,B$	7	∞	$68\,B$	6	$304\,B$	10	∞	$68\,B$	6	$352\,B$
			7	$248\,B$				7	$304\,B$				7	$360\,B$
			8	$268\,B$				8	$316\,B$				8	$372\,B$
			4	$200\,B$				4	$236\,B$				4	$268\,B$
			5	$264\,B$				5	$312\,B$				5	$368\,B$
5	∞	$64\,B$	6	$264\,B$	8	∞	$88\,B$	6	$320\,B$	11	∞	$88\,B$	6	$376\,B$
			7	$264\,B$				7	$320\,B$				7	$376\,B$
			8	$284\,B$				8	$340\,B$				8	$388\,B$
			4	$216\,B$				4	$244\,B$				4	$276\,B$
			5	$280\,B$				5	$336\,B$				5	$384\,B$
6	∞	$64\,B$	6	$280\,B$	9	∞	$88\,B$	6	$336\,B$	12	∞	$88\,B$	6	$392\,B$
			7	$288\,B$				7	$336\,B$				7	$392\,B$
			8	$300\,B$				8	$356\,B$				8	$412\,B$

These results will be presented in a manuscript I am preparing for publication.

Project Euler

Some of my peers have been utilizing Project Euler as a set of sample problems to solve while learning new languages. I have used this opportunity to dive back in and continue my progression by participating with them. Interestingly, they have not chosen the easier problems from the beginning of the archive. Rather, they have select problems from the 200 range. The combinatorics required to solve Problem 205 was a refreshing experience in recurrence relations and dynamic programming. Similarly, when solving Problem 206 I appreciated applying number theory to prune the search space rather than employing brute force search.

AMNH Research Associate

My collaboration with Wheeler Lab at AMNH has, at long last, culminated in a manuscript submitted for publication. We define a representation of phylogenetic

networks with minor restrictions on character coding and describe an accompanying algorithm for computing an efficient scoring measure of said networks utilizing memoization of shared substructures. The network restrictions specify the concept of character blocks which are atomically optimized and the scoring algorithm introduces a network edge penalty. The conjunction of these results in a optimization procedure which searches over the space of (constrained) phylogenetic networks, yet does not yield a trivial result with many "reticulated" edges. The manuscript has been submitted for publication in Cladistics.

A second manuscript is in preparation announcing the lab's next-generation, multi-purpose phylogenetic analysis software program along with an open source library toolkit. This manuscript describes the scope of the software's capabilities and compares the capabilities to those of existing related software programs. The software support the two dominant forms of phylogenetic analysis, parsimony and [maximum likelihood][Wiki-Phylo-MaxL Importantly, the manuscript includes a description of how to utilize our program to evaluate phylogenetic networks with the scoring algorithm recently submitted for publication. A notable feature of the software, from my perspective, is that it is written in Haskell.

Finally, I have been engaged in some intermittent consulting related to performance tuning Haskell code within our software program and the accompanying toolkit. One interesting problems involved ensuring large objects are do not have lingering references from a stack-frame of a deep, recursive call stack. Another encompassed $\mathcal{O}(1)$ bidirectional marshalling of data-structures across the C FFI. Undoubtedly, more performance tuning will follow over the coming months as the lab's labors are publicized to the wider world.