Project Proposal
March 18th 2024

Project Partners: Adam Gohain and Sebastian Pucher will be working together

Topic of Study: *Conway's Game of Life (GoL):*

Overview of Project, Topics, & Structure:

1) **Defining REACHES-CONFIGURATION**
    a) **REACHES-CONFIGURATION** will be our basic decision question used to analyze many different patterns in GoL
    b) The decision question asks: Given initial configuration of the grid, will the starting configuration A evolve into confirmation B
    c) We will specifically apply our **REACHES-CONFIGURATION** decision problem to test if certain inputs will evolve into the following patterns: Still life (blocks), Oscillator (Blinkers), and spaceship (Gliders) (*This last one is a challenge)*
    d) More formally: We will analyze the time complexity of the following three decision problems and categorize them. (We will construct an argument they exist in **NP**, but we'll see what happens).
    e) For part 1, (a-e), we will construct our findings use a finite grid
2) **Showing REACHES-CONFIGURATION is undecidable**
    a) We need to read more into sources for detail of the proof/argument here, but it is of the same nature as the halting problem
        i) ds.dynamical systems - Undecidability in Conway's Game of Life - MathOverflow
        ii) The following text includes a general sketch of the proof
    b) *For part 2, we will construct our argument using an infinite grid*
3) **What *can* be computed/decided in Conway's Game of Life?**
    a) Go over the construction of a Turing Machine in Conway's Game of Life
        i) **Turing machine - LifeWiki**
        ii) Turing Machine Simulator - ConwayLife.com
        iii) full-length paper about it
    b) Describe the construction of a universal constructor in GoL
        i) Universal constructor - LifeWiki
    c) Describe a universal computer
        i) Describe how powerful a universal computer with a universal constructor tacked on is
    d) Very broad overview of how a) through c) in this part let us define UTM and universal UTM construction (a full proof is MASSIVE)
    e) Some actual constructions in GoL that compute things (we'll use different variants of prime calculators as some nice examples..maybe find more from the exercises of the main GoL text if there is time)

i) Brief 'lil glider overview if not already one of the construction examples covered in previous parts
ii) Prime Calculator
iii) Twin Prime Calculator
iv) Fermat prime calculator in Conway's Game of Life
    (1) Covered in 8.8 of main textbook
v) Mersenne Prime Calculator
    (1) Exercise 8.4.1 in main text

4) **To supplement our findings, we plan on developing out a demonstration of Conway's game of life in RUST. Come presentation day, we'd like to share with the class a visualization of the game as support to our explanations. (Plan on building as TUI using ratatui framework).**

Project Goals:

- Gain a deeper understanding of configuration paths and time complexity of Still Lifes, Oscillators, and Gliders
    - Synthesize where the **REACHES-CONFIGURATION** question fits into our time complexity hierarchy.
- Demonstrate the limits of Conway's game of life, the problems that it can and cannot solve
    - Explore parallels between the limits of the game and various complexity classes / findings we have discovered throughout the course so far
- Apply some RUST programming skills!

Sources:

conwaylife.com/book/conway_life_book.pdf
texts found in this folder of our repo (to be continually updated as we find more)

Turing machine - LifeWiki
Universal Turing machine - LifeWiki
Universal constructor - LifeWiki