EECS 510 Final Project: The Gym Language

Ginny Ke, Jenna Luong, Nikka Vuong

May 2025

Intent, Structure, and Purpose

The purpose of this formal language is to represent a structured gym routine, using exercises, rest days, and workout splits as the elements. The language is designed to help gym-goers of all levels to understand how to create balanced workout schedules that align with foundational fitness principles. By formalizing gym routines as strings over a set of symbols and rules, we can analyze and ensure that workout plans are optimally structured for health, recovery, and progress. The Gym Language is aimed to prevent common issues such as overtraining, undertraining, and inadequate rest.

The Gym Language models a typical "workout split" format, where various muscle groups are targeted through "push, pull, and leg" routines followed by rest periods. The intent behind the language is to firstly, promote and advocate for a better understanding of workout organization in a consistent and goal-oriented manner, and secondly to reduce common mistakes such as neglecting rest and overworking/underworking certain muscle groups. The language offers a systematic method to planning workouts that is both readable by machines and interpretable by athletes, thus blending the theory of computing with real-world fitness.

The structure of the language is broken down into 4 categories: push(s), pull(p), legs(l), and rest (r). The language utilizes these elements to create an alphabet

$$\Sigma = \{s, p, l, r\} \qquad (1)$$

Each character corresponds to the respective activity or rest, ensuring any valid string directly coincides with the structure of a seven-day week. The language introduces grammar rules and constraints such as: if push is hit, then leg must follow, if pull is hit, then leg must follow, must not repeat the same activity or rest day consecutively (aimed to prevent undertraining and overtraining), and input does not end on push or pull. Due to the combination restrictions the language can be classified as finite and regular. Thus, the Gym Language can be represented by a Nondeterministic Finite Automata (NFA) as the usage of a stack is not needed. These rules help inscribe important fitness principles into the logic of the Gym Language.

In conclusion, the Gym Language closes the gap between computational theory and fitness planning. It provides a visual way to generate and assess workout schedules using a logical and rule based system. With the aid of formal tools this project highlights the application of computing theory to areas outside of math and computer science.

***Please note that every athlete has their own training approach. The routine described here reflects the preferences of the creators of Gym Language and may not be suitable for everyone. This routine may not be suitable for everyone!**

Grammar

Alphabet: Σ = {s, p, l, r}

Workout splits:

s = push
p = pull
l = legs
r = rest

Rules:

- If push is hit, then leg must follow
- If pull is hit, then leg must follow
- No consecutive splits
- Input does not end on push or pull

Production Rules:

S → sF | pF | lL | rR
F → lL | l
L → sF | pF | rR | r
R → sF | pF| lL | l

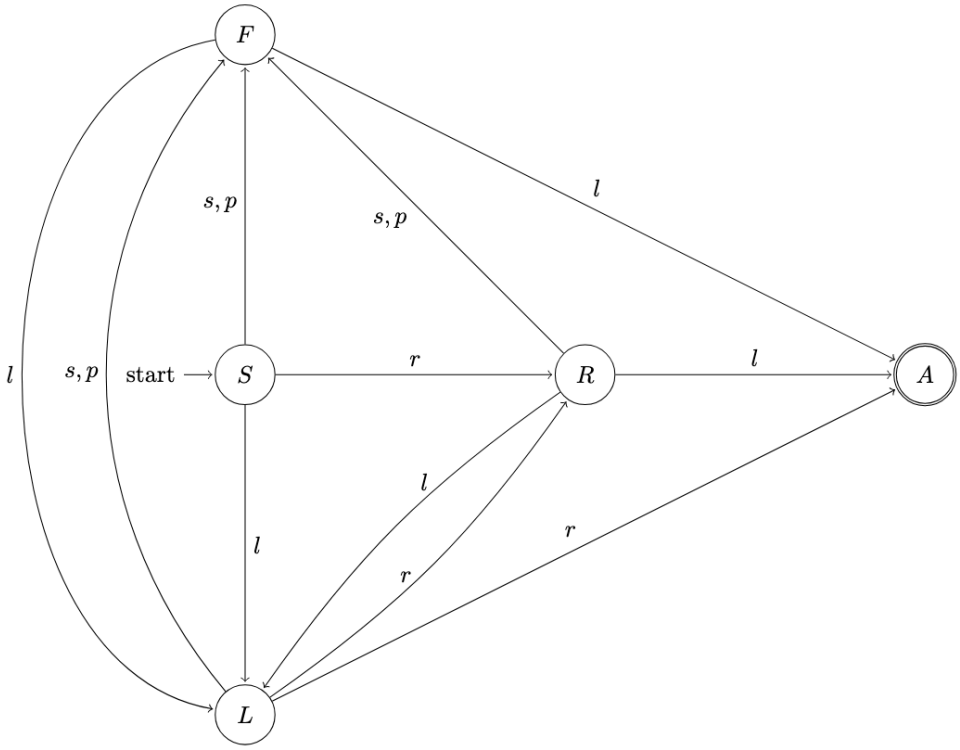
Valid Strings:

s l p l r s l
r p l r s l r

Invalid Strings:

s s p l r s l
s l p l r s l p

Automaton



***NFA diagram representing our production rules and grammar. We opted for an NFA instead of a PDA, as our language did not need a stack to handle any memory.**

Data Structure

gymlanguage.txt is in the github repository. A screenshot of the file is shown below: