Part 1 - Design a Formal Language Writeup

Creators: Anna Lin, Kusuma Murthy, Sophia Jacob

The language that Group 7 chose to create is called LEEP of Fate: Rock Chalk Chronicles. While the language itself represents a fun game with a KU-related theme, it also provides a deeper understanding of finite automata and the foundation that goes into creating a unique language. Through this project, Kusuma, Sophia, and Anna aim to create an interactive language to gain better insights into the grammar of a context-free language in the form of Pushdown Automata. This helps simulate the real-life examples of languages Computer Science students interact with in their programs while making it an engaging experience in the form of a game.

The inspiration for this language came from our shared experiences of having pride in KU and our love to see our talented students represent the student body. We want to represent the language of the KU students by showcasing our spirit for our university. In this game, our famous Jayhawk aims to defeat the Wildcats through the correct series of spells, quite similar to the athletic games our students love to watch. These spells constitute a formal context-free language, with its own set of rules.

Our team wanted to incorporate our love of magic and spells into this language, as we are all avid Harry Potter fans and found it fascinating how each spell in the books had its own meaning. Now, with this language, we can incorporate a fantasy element in which the two rival mascots of Kansas, battle each other with spells to create a valid string in our language. This is much like how spells must be cast in a certain sequence.

This ancient language consists of four basic symbols and one special character. The four basic symbols are the main elements that help create powerful spells to protect our realm from being attacked by the Wildcats of the West. Relating to the KU values, our language symbols closely follow the ethics our students strive to uphold. With these morals, our KU students are armed to enter the real world, much like how the Jayhawk needs these four elements to defeat the Wildcat.

f represents fire which can be used to wrangle the wild beasts of the West, just like KU students strive for excellence in the work that we commit ourselves to. w represents water which can be used to fight with poise but with much resilience. This represents the value of innovation,

especially for Engineering students who want to make an impact in the real-time processes in our society. a represents air that can blow our enemies away with a single gust of wind. This aligns with integrity, our next KU value, as we want to stay firm in our honesty and beliefs. Lastly, e represents Earth which uses Mother Nature to power our way through the battle. We associate this symbol with respect, as we want to care for those around us, while still being strong like Mother Nature.

Additionally, our language has the * character. The star signifies the Jayhawk's victories, their strides towards freedom, their pride for KU, and lastly, their ability to come together as one Jayhawk nation. We related this symbol to the last KU value: stewardship, as students rise together to promote our university. Therefore, our alphabet is $\sum = \{f, w, a, e, *\}$.

Gaining victory, however, is no easy task, as the Jayhawks must work hard together, day and night, march up the hills, and through the fields to cast the winning battle spells. In order to be considered as a battle-winning spell, the length of the number of f's and w's combined should be odd if they are cast in the spell. Likewise, the length of the number of a's and e's combined must be even if they are cast in the spell. A single run of a's and e's must be consecutively next to each other and must be even in total for each sub-section. For example, a valid string would be: $aafeaaewf$ *, but $afefeef$ * is invalid.

Finally, our rules require that there must be only one * symbol that appears at the end of the string. Following the rules of this ancient language, the Jayhawks will be able to defeat the Wildcats and uphold our KU pride. If they decide to betray the rules, however, the consequences will lead to the loss of our KU spirit, meaning they will be deadlocked in our automaton.

This language requires a Pushdown Automata, as we need to keep track of the number of occurrences of certain elements in a string, by using a stack. Additionally, the language has non-determinism since the PDA does not immediately know if there is another element in the rest of the string that will complete it, making it either odd or even. Therefore, this CFL can be represented well with a PDA, and the string must end with an empty stack to be recognized.

Finally, relating back to our EECS 510: Introduction to Theory of Computing class, this language is a great way to showcase the skills that we have learned regarding designing a language, creating grammar, developing an automaton, programming a data structure, and testing the language.

Condensed Version of the Rules

LEEP of Fate: Rock Chalk Chronicles

- **Purpose:** To defeat the wildcats using the correct series of spells that will lead to the ultimate demise of the Wildcat.
- **Alphabet** $\sum = \{ f, w, a, e, * \}$
- **Rules:**
  - The string must have only one * in the string that appears as the last character.
    - The smallest valid string is *.
  - If there are $f's$ and $w's$ in the string, then the combined length has to be odd.
  - If there are, $a's$ and $e's$ in the string, then the combined length has to be even.
    - Additionally, each run of a's and e's must be even in length, meaning they must be consecutively next to each other. For example, $aafeaaewf$ * is valid, but $afefeef$ * is not valid.
- **Definitions:**
  - f = Fire
  - w = Water
  - a = Air
  - e = Earth
  - The * means Rock Chalk Jayhawk!

- **Example Strings:**
  - Valid Strings
    - $fwaaeef$ *
    - $faawaaeewww$ *
    - $aefwfaeff$ *
    - $weefeafaaaa$ *
    - $aeaefwf$ *
    - f *
    - *
  - Invalid Strings:
    - $fwaaee$
    - $fawe$ *
    - $afeff$ *
    - efe *
    - $afefewa$ *