Instituto Tecnológico y de

Estudios Superiores de Monterrey

ACTIVIDAD:

Actividad Integradora 3.4 Resaltador de sintaxis (evidencia de competencia)

Alumno:

- Juan Pablo Montoya A01251887

Campus:

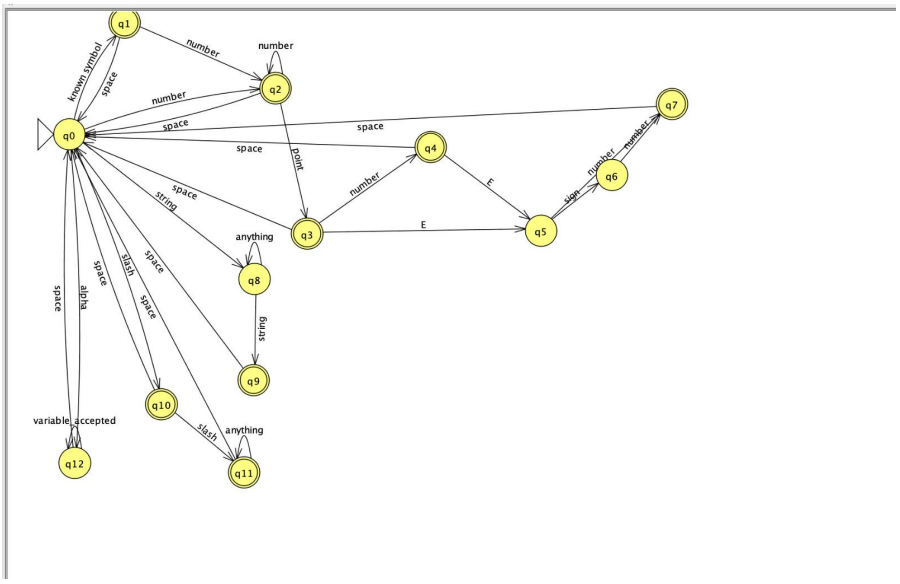
Guadalajara

Introduction

In the following report, the implementation of a deterministic finite automaton for programming language highlighting is evaluated. Computational complexity of algorithms is a major metric to evaluate the performance and resource consumption of a program. In this case the task to solve was the tokenization of specific characters and words in a diversity of programming languages. After the tokenization, a post classification was implemented and finally a text document was generated with the corresponding html and css encoded result. As an input, the user introduces a text file with the content that is to be processed. The dfa engine was written in a csv file, and further converted to json format, to be read by the program. The final result is generated in the index.html and styles.css files. Programming language syntax highlighting is a major help while developing software, thousands of applications use it and it has aided programmers to develop most of the software we know in the current times.

Procedure

- **Design DFA:**



- **Write DFA in excel table**
- **Convert DFA to json**
- **Read dfa as a dictionary in python**
- **Iterate for every letter and append each letter to the state dictionaries.**
- **Post process to get the token that each state represents**
- **Input the result to the Generator class**
- **Create a csv with specific tokens and their class**
- **Convert csv to json and read as a dictionary**
- **Iterate each element of the result from the dfa and lookup in the class dictionary**
- **The class dictionary will contain the class index corresponding to that lexical class**
- **Append the span wrapped text with the specific class to the results array.**
- **Wrap the results array inside of an html and body tag text that contains the link to a stylesheet.**
- **Output a predefined stylesheet with all the classes**

Complexity

In terms of time and space complexity. We have to consider all the steps of the process. To begin implementing the DFA will use constant space, considering that the dfa dictionary is a static memory allocation. Reading the Json file is an O(N) operation being N the number of characters in the dfa json file. Nevertheless the json file will not change, therefore we can say that the time complexity is also O(1) or constant. The DFA runs in O(N) time being N the number of characters in the input file. Everytime we hit a space we run an additional join for the characters stored in the state array. This is to address the memory problem caused by the immutability of python strings. Modifying raw strings in runtime may hit the space complexity of O(N^2). Using an array and joining it when required is a better approach for space. The total space would be an additional O(N) being N the number of characters in the text file. The post processing takes O(M) time being M the number of characters formed. Being that M will always be less than N (number of characters) we can stay with N being the overall complexity of the solution. Doing dictionary lookups has a complexity of O(1). In conclusion the complexity in time and space for the solution would be O(N). There is nothing better than this, because the program needs to see all the characters before classifying them into tokens. Therefore an O(logn) solution would not be possible.

Ethical Considerations

Today, the implementation of DFAs and regular expressions is used to serve the purpose of highlighting text in code editors. Nevertheless this technology could be applied to more robust and complex engines that would be able to find patterns in people's browser searches, messages, posts, etc. Regulating the use of the language analysis and natural language

processing (which is the next level of what we are building) is essential in the modern days. A few large companies own the majority of the private data in the world. And we have accepted their terms to use our information as they will. It is imperative to keep working on worldwide legislations for this systems, in order to ensure the privacy and wellbeing of humanity. It is possible to find people near certain areas with certain interests just by web scraping media posts and analyzing them to find keywords and patterns. This could be a major tool for organized crime groups, who could target people depending on these factors. Targeted campaigns for fraud could also take advantage of these systems. As well as political parties and companies. Understanding language is the key to understanding the people, and understanding the people is the key to controlling them. If we want to live in a world of freedom, we have to enforce limitations to the utilization of NLP and language analysis software.