

# Project 3

---

## Finite State Machine

Kekoa Riggan - Ling 473 - August 22, 2017

### My Approach

I utilized the skeleton code provided in the resources folder for the assignment. The skeleton provided the initialization of the Thai characters as accepted input for transitions and a blank function for the finite state machine that are used in my script.

First, I initialize the acceptable inputs for each transition. This was provided in the skeleton code. Next in the code is the FSM implementation. The function for the FSM is called repeatedly, once per line in the source file. Upon running, the function sets an int at 0. This int will serve as the saved state for the machine. Then, the function moves through a series of if/then/elif statements that serve as the FSM implementation.

For each character in the string, the state is checked, priority going to the lowest number (0, 1, 2...). Once the if returns True, then the character is checked for which acceptable input it belongs to in priority according to the definition of the FSM. Once a match is found, the machine state is updated to the coordinating state and the character is added to a string that will return at the end of the function.

Because 7, 8, and 9 are accept states, they are grouped in a separate if/then/elif cluster. When a state encounters a transition to an accept state, the blank space and character are added at that time to the save string. The if statements for accept states only update the state to the appropriate initial state.

Once a line is processed, the saved string is written to the output file. Once all the lines are processed, the footer is written to the output file.

### Reflection

1. If/then as an implementation. I fully expect to lose points for efficiency and will be pleasantly surprised if I don't. Prior to this assignment, I had no idea what the implementation of a FSM would look like. As I did some research, I realized that the standard implementation featured classes, which I struggle to implement in python. I considered writing this assignment in java, but I was afraid I would have encoding issues or issues with patas. So, I wrote some psuedocode by hand and thought about the implementation. I finally decided to stop thinking about it and try something. The if/then statements worked very well at the first run so I stuck with it. When my script produced the same results as the other students online, I was completely shocked.

2. Implementing functions for better code. I know I should have used functions instead of rewriting each of my transitions. I had to prioritize some other things in my life right now and this is going to be labeled WONTFIX.
3. I initially ran into encoding problems. I used the 10-minute rule probably 5 or 6 times (making progress on my own right at the 10 minute mark) until I finally pushed all the way through the encoding without asking for help. Given that each student is using different methods, IDE's, and programming languages, this may have been the best decision.