Dominic Rossillo

2/1/16

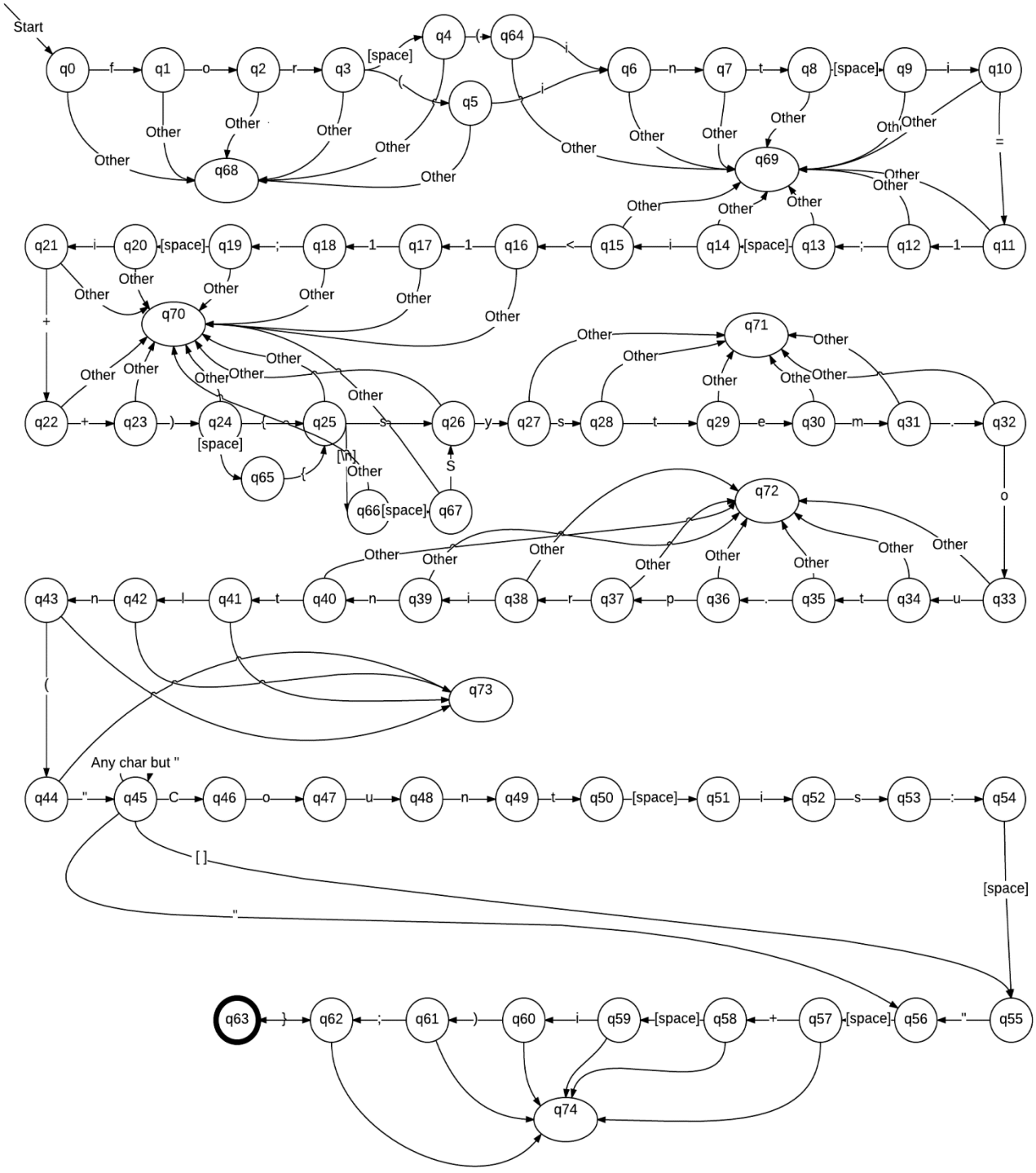
Lab 1

CMPT 440

Essay 1: For this lab I hit a number of problems that required me to think hard about what is the best way to construct my DFA. Primarily at the start I was unsure on how to make a DFA that could handle two different formats of code that technical say the same thing. Because some code being written with different formats but run the same it is important to keep in mind what different kinds of formats that would be used by a programmer when constructing the DFA. My approach to making the DFA was creating the original way of writing the for loop and then looking at it and saying "what can I put into this that would still be right but would error out on what I have." Then I made those changes in order to make sure both formats worked. Similar to this, creating cases for errors was difficult because you don't want the program to error out when a user does not write something word for word. For example inside the " " the users technically does not need to type anything to have the program run as intended. Because of this I added extra paths that would take into account users typing nothing, things other than the given string, and the given string so that errors would not occur.

Essay 2:

In order to create a DFA in java to validate a string I would start by creating a function that takes a string (the command being tested) as a parameter. The function would first use a while loop that would be true if the string.length<= I, I would be used as an increment to move through the string in order to validate each character and command. The while loop would then take each character and append them to a variable called keyword. The loop would run until the string length was reached or until a [space] or a . is reached. An if statement inside the while loop would be used to check if the current character was a [space] or an . . If it is a space or a . the if statement will check to see if the keyword variable equals any of the defined words in the programs dictionary. If it does equal a predefined keyword the if statement will make keyword="" as to check the next keyword in the string. If the keyword is not recognized then the function will set i=string.length resulting in the while loop being exited. In order to give feedback on whether the string is valid or not there will be an if statement to check if i+1>string.length. If this is true then the program will check the word in keywords dictionary setting a variable "isValid" to true if the keyword is acceptable or false if it is not recognized. In order to avoid the program hitting issues with strings that would be printed another if statement will not check or a keyword until a closed " is given and resulting in the end of the string.



Assumptions:

- This is to test only a for loop using I as an increment
- The function requires a system print with the count
- The user is using java