Philip Hasse

5/9/17

Final Write up

Abstract

This paper goes into detail around my baseball themed DFA and how I plan to implement it. A detailed description of the DFA is given as well describing exactly what the user will experience when using the system. With a detailed description of how the DFA works, we give requirements that the user will need to follow as well as a user manual that will give exact commands that the user will need to put in to move through the DFA. Along with all of the specifics to this specific application, the paper will go into a little detail on how similar application have used a DFA like this one.

Introduction

The DFA described in this paper will be bringing us through a half inning in a game of baseball. It will take into account every action a batter can go through, however will not be keeping track of the score or the amount of players on base. This idea is useful for creating large scale applications where actions of the batter need to be recorded. In this paper users will learn how to interact with the system, be given requirements needed to run the application, be given a

brief overview of other work in this field, and finally a user manual to get rid of any and all confusion within the application.

Detailed System Description

The DFA being implemented will bring a user through a half inning of baseball. They will be acting as the batter at the plate where they will input what action happened after each pitch. The pitch could result in a ball, strike, hit, hit by pitch, etc. If the batter gets on base the DFA will go back to the state where the count is 0-0-X where X is the number of outs. When the number of outs hits 3, the DFA will have reached its accepting state. The user will be inputting commands as the application runs and be updated on the count as well as how many outs there are in the inning. This application won't keep track of score of who's on base; the idea is to just keep account of what the current batter does and the state of the inning.

Requirements

The requirements in order to use this application are very simple. Having Java installed on your computer and being able to run it through the command line or terminal is all that will be needed. The application will be taking input and sending feedback based on that input.

Literature Survey

The information on an implementation of a DFA like mine isn't much documented and out there, however there are many games and applications that use a similar idea that need information that I am gathering. Typically it will be in more detail and may even contain more than one DFA to accomplish this goal. Applications like MLB live give a virtual game experience telling users the count, the score, where the ball was hit, etc. This takes my proposed DFA and up scales it to be used in an actual baseball game experience.

User Manual

1. Run the application through java in the command line or terminal

2. You will be prompted with a small message that you started

3. Complete a full count of an at bat until you input 3 strikes before you get 4 balls

4. Type any string with 's's and 'b's

5. If you type 3 's's before you type 4 'b's the DFA ends and you're out

5. If you type 4 'b's before 3 's's you walk, which mean the count starts over and we continue until we get 3 's's before 4 'b's

Conclusion

In conclusion, This DFA isn't a revolutionary idea; however it exemplifies the idea of a DFA and is a great example on how they can be implemented on a variety of things. It takes into account many different inputs and can move through the process smoothly. It's a great example of how a simple DFA can be expanded and implemented in complex ways like the MLB does for their live update system.