

Relatório Sprint 2

AFD - Autômatos Finitos Determinísticos

Charles R. Almeida, Edson Camargo, Marco Antônio J. Ticona

Curso de Engenharia de Computação
Universidade Federal do Pampa (UNIPAMPA) - Bagé, RS
{charlesalmeida,edsoncamargo,marcoticona}.aluno@unipampa.edu.br

1. Objetivo

Construção de um método para o autômato finito reconhecedor da forma *bool/Reconhecer (String, config *)*, onde o primeiro argumento é a palavra que deverá ser reconhecida e o segundo argumento, a computação que levou ao reconhecimento, ou não, da palavra.

2. Processos da Sprint

Após a definição da atividade e conhecimento dos integrantes do grupo, definimos as ferramentas para comunicação (*Discord*), *Github* como repositório do código e a *IDE Netbeans* como ambiente de programação. Os papéis dos membros do grupo nas fases do processo estão dispostos na tabela 1:

Tabela 1: Papéis dos membros do grupo na Sprint

Fases do processo	Responsáveis
Planejamento da Sprint	Charles Almeida (<i>Scrum Master</i>)
Codificação	Marco Antônio e Edson
Definição dos casos de testes	Charles e Marco Antônio
Revisão do código	<i>Scrum Master</i> e equipe
Testes	Marco Antônio e Edson
Documentação	Marco Antônio e Edson
Relatório	Charles Almeida (<i>Scrum Master</i>)

As reuniões e definições durante a Sprint estão dispostas na tabela abaixo:

Data	Horário (início-fim)	Descrição da reunião
09/10	20:00hs às 21:30hs	Fizemos a primeira análise do código e discutimos a forma de implementar a função sugerida. Definimos o github como forma de compartilhar o código. Sugerido solicitar a documentação do código do AFD da Sprint 1 e confirmação da data de entrega da Sprint.
14/10	18:00hs às 19:00hs	Após análise e compreensão do código da Sprint 1, definimos a implementação da função booleana usando como parâmetros de entrada os resultados obtidos na função da Sprint 1. Definimos a data da próxima reunião para o dia 17/10, sábado, às 20:00hs.
17/10	20:00hs às 21:00hs	Na reunião 3 definimos dividir o código em duas funções para ingressar com os dados do alfabeto, estados e palavra e outra para reorganizar o reconhecedor e mostrar a computação da palavra. Mostrar na função as transições passo-a-passo da palavra, se reconhecida.
28/10	18:00 às 19:00hs	Com as alterações propostas do código prontas, decidimos realizar os testes e finalizar o artigo na próxima reunião marcada para o dia 31/10, às 19hs.
31/10	21:00hs às 21:45hs	Nessa reunião, analisamos as alterações implementadas no código e iniciamos a testagem do mesmo utilizando os autômatos apresentados em aula, comparando os resultados esperados com os obtidos. Definimos uma nova reunião para o dia 01/11, às 21:00hs, para uma última análise antes da entrega do projeto.

01/11	19:30hs às 20:30hs	Considerações finais.
-------	--------------------	-----------------------

3. Conclusões

Conforme as alterações propostas na reunião 3, modificamos a classe principal dividindo-a em funções mais específicas para receber os dados do usuário, como o alfabeto, os estados inicial e final, as transições e a palavra a ser reconhecida (*public void iniciaAFD()*), outra função *organiza* (que é uma matriz de char) que recebe o vetor de strings com as transições inseridas pelo usuário e coloca-as em uma matriz. Na função booleana *reconhecer*, a palavra e a matriz de char são recebidas como argumento e testada para saber se seu último símbolo coincide com o estado ou estados finais informados. Caso seja reconhecida, o autômato é finalizado mostrando o estado final e as transições dos estados até o reconhecimento da palavra. Foi acrescentada um print das transições realizadas pelo autômato para um melhor entendimento do assunto.

4. Análise qualitativa

Com o código do reconhecedor de linguagens produzido na Sprint1 e a definição da função boolean da Sprint2, coube ao grupo modificar a estrutura inicial do código para adequá-lo à função proposta. Houve grande aceitação e disponibilidade dos integrantes do grupo para as reuniões, assim como, na solução do problema apresentado.

Referências bibliográficas:

LEWIS, Harry R.,PAPADIMITRIOU, Christos H. Elementos de teoria da Computação. 2ª Ed. Bookman, 2004.