

Sprint 3 – Transformação de estruturas

Objetivo e entregas

O objetivo da Sprint 3 é a construção de algoritmos para a transformação de estruturas. Cada grupo será responsável pela definição do algoritmo que recebe uma estrutura de um tipo (gerador ou reconhecedor) e a transforma na estrutura solicitada, de forma que as linguagens geradas/reconhecidas nas duas estruturas sejam iguais. Todos os algoritmos de transformação estão disponíveis na bibliografia da disciplina e somente precisam ser implementados.

Os algoritmos devem levar em consideração as classes já construídas nas sprints anteriores, sendo que essas estruturas somente podem ser modificadas, nessa sprint, em acordo com os demais grupos que estão trabalhando com a mesma representação. É papel dos *scrum masters* dos grupos organizarem as alterações, de forma que todos devem trabalhar sobre as mesmas representações.

Atividades prévias

Cada grupo, considerando os dispositivos geradores/reconhecedores atribuídos, deve ler o material do referencial bibliográfico referente aos dispositivos, conforme sugerido nas sprints anteriores. É possível usar outras referências. Evitar usar códigos prontos da Internet. O processo de adaptação de uma estrutura em outra é mais custoso do que a construção do programa a partir dos algoritmos já definidos.

Atividades da *sprint*

O objetivo da *sprint* 3 é construir os algoritmos de transformação de estruturas que deverão fazer parte do sistema LF-Studio. Particularmente, um novo método deverá ser implementado no dispositivo gerador/reconhecedor que será transformado. O método deverá fazer uso das estruturas de dados definidas pela classe; quaisquer modificações (permitidas) deverão ser acordadas entre os grupos que vão trabalhar sobre a mesma estrutura de origem, para não haver inconsistências. Caso haja modificações, estas deverão ser comunicadas a todos os grupos, visto poderem ter impacto nas estruturas que serão resultados da transformação. Note-se que o princípio de oclusão da informação (*information hiding*) da programação orientada a objetos, se foi corretamente implementada anteriormente, não deve gerar problemas nas estruturas-alvo, visto que não deve haver acesso direto aos atributos da classe, que somente serão acessados por meio de métodos.

Nesse ponto do trabalho, já deve estar claro para todos os discentes a importância dos elementos fundantes da programação orientada a objetos (encapsulamento, oclusão da informação, herança e polimorfismo de subclasses, bem como sobrescrita e sobrecarga de métodos). Da mesma forma, a importância do planejamento e do gerenciamento do projeto também deve estar claro para todos.

Os métodos devem ser construídos a partir dos algoritmos de transformação presentes na literatura. Os grupos podem trocar ideias e informações sobre a matéria e sobre os elementos que estão construindo, preferencialmente por meio de seus respectivos *scrum masters*. Cada grupo, contudo, será responsável por um único algoritmo.

Os grupos, para a construção dos algoritmos de transformação de estruturas, estão assim definidos:

- G1** : Expressões regulares em autômato finito não determinístico (ER->AFND)
- G2** : Autômato finito não determinístico em autômato finito determinístico (AFND->AFD)
- G3** : Autômato finito determinístico em gramática regular (AFD->GR)
- G4** : Gramática livre de contexto em autômato de pilha *bottom up* (GLC->APBU)
- G5** : Gramática livre de contexto em autômato de pilha *top down* (GLC->APTD)
- G6** : Autômato de pilha determinístico em máquina de Turing (APD->MT)
- G7** : Autômato finito determinístico em autômato determinístico mínimo (AFD->AFDMin)

A entrega corresponde ao seguinte conjunto de artefatos:

1. O código **da classe Java**, em um arquivo .java com todos os elementos documentados (pode ser usado o javadoc ou documentação no código) – **NÃO ENTREGAR PROJETOS COMPLETOS**.
2. Os casos de teste projetados, com definição de entradas, procedimentos e saídas esperados.
3. Os resultados da execução dos casos de teste, com comparação entre a saída esperada e a saída real.
4. O relatório do *Scrum master* detalhando o planejamento e a execução do trabalho. Particularmente, no relatório devem constar:
 - Os dados de planejamento da *sprint*, com datas e equipes responsáveis.
 - As datas e registros das reuniões (horário de início e fim e decisões tomadas) de planejamento, revisão ou outras.
 - Uma análise qualitativa do processo de gestão do grupo e dos resultados atingidos, explicitando pontos fortes e pontos fracos do grupo e sugestões para a melhoria do processo.
5. O referencial bibliográfico utilizado para o desenvolvimento do trabalho.

Grupos de trabalho

- G1** : Expressões regulares em autômato finito não determinístico (ER->AFND)

EDSON AZEVEDO DE CAMARGO (*Scrum master*)
 ERIC DIAS DA SILVA ROSSO
 KELVIN CLOVIS MONTOLI DE SOUZA
 MICHEL ALMEIDA DA SILVA
 RITIELE MOREIRA ALDEBURG

- G2** : Autômato finito não determinístico em autômato finito determinístico (AFND->AFD)

CHARLES RODRIGO RIBAS ALMEIDA
 GABRIEL RITTA CORREIA (*Scrum master*)
 ITIBERE GONCALVES SILVA FILHO
 MATEUS ROSA PADRAO
 MATHEUS DE JESUS MARQUES

G3 : Autômato finito determinístico em gramática regular (AFD->GR)

FERNANDO LUIZ COLOSIO
GUILHERME DOMINGUES GOULART (*Scrum master*)
LOURENCO NATANIEL PINHEIRO PORTELLA
MARCO ANTONIO JORGE TICONA

G4 : Gramática livre de contexto em autômato de pilha *bottom up* (GLC->APBU)

HERBERT CAETANO DA SILVA (*Scrum master*)
MATEUS OLIVA SOARES
THAMIRES SAMPAIO PONTES DE MATOS
YANN DA SILVA

G5 : Gramática livre de contexto em autômato de pilha *top down* (GLC->APTD)

CHARLES CRUZ DE OLIVEIRA
LUIS ALBERTO VITAL MARQUEZZINI JUNIOR
MARIANA POMPEO FREITAS (*Scrum master*)
PEDRO GABRIEL MOTA PEREIRA

G6 : Autômato de pilha determinístico em máquina de Turing (APD->MT)

LUCAS FARIAS PAIVA (*Scrum master*)
MARCELO DE AZEVEDO SILVA
SAMARA BUENO MARQUES
THIAGO PORTO MENDES
TIAGO LEITE BRITO
TIAGO MACHADO JARDIM

G7 : Autômato finito determinístico em autômato determinístico mínimo (AFD->AFDMin)

GERSON DE MUNHOS CONCILIO
GIULIANA OLIVEIRA DE MATTOS LEON
JOAO ANTONIO DOS SANTOS ANTUNES
MARIA ELIZABETH BARCENA SILVA (*Scrum master*)
THIAGO FERREIRA LEAL