



UTFPR - CAMPUS PONTA GROSSA - COCIC

Disciplina de Compiladores – 2024.1

Trabalho Prático 1 – Construção de um Tradutor da Linguagem Natural NAG
para a Linguagem Jason/AgentSpeak
Professor: Gleifer Vaz Alves

1. Objetivo:

Implementar um tradutor da linguagem natural NAG (*Natural language for AGents*) para a linguagem de programação de agentes Jason/AgentSpeak.

Note que o tradutor deverá corresponder apenas a um subconjunto da linguagem Jason/AgentSpeak.

2. Descrição das Etapas do Trabalho:

(a) Apresentação e Representação da Gramática:

Para representar a gramática criada deve ser usada a notação BNF (*Backus-Naur Form*), ou alguma extensão como a EBNF.

(b) Implementação do Analisador Léxico por meio da ferramenta FLEX.

(c) Implementação do Analisador Sintático por meio da ferramenta BISON.

(d) Elaborar um conjunto de testes significativo que represente todos elementos envolvidos na tradução da Linguagem NAG para Jason/AgentSpeak.

- O tradutor deve ler a entrada de um arquivo “fonte” em NAG e gerar um arquivo de saída no formato Jason/AgentSpeak (extensão `.ASL`).
- **Obs.:** o arquivo de saída deve ser executado com sucesso no interpretador da linguagem Jason/AgentSpeak.
Veja em: <https://jason-lang.github.io/jason/doc/tutorials/getting-started/readme.html> detalhes para instalação da ferramenta.
- Site oficial do Jason para maiores detalhes.
<https://jason-lang.github.io/jason/doc/>
- A BNF da Linguagem de programação de agentes Jason/AgentSpeak pode ser consultada no seguinte documento (ver Seção 3):
<https://jason-lang.github.io/jason/doc/Jason.pdf>
- Outra alternativa é utilizar o JACAMO para rodar os agentes. Neste caso, consulte o site oficial para instalação: <https://github.com/jacamo-lang/jacamo/blob/master/doc/install.adoc>
e também o material complementar disponível no próprio Moodle.

(e) Escrita do relatório do trabalho, conforme modelo de trabalhos acadêmicos da UTFPR. Logo, deve conter Capa, Sumário, Introdução, Referências, etc.

(f) Preparação da apresentação para defesa do trabalho.

3. Gramática da Linguagem NAG:

A Gramática (BNF) que segue descreve as regras sintáticas da linguagem natural NAG e serve de referência para o desenvolvimento do trabalho.

$\langle programa \rangle ::= \langle Lagentes \rangle$

$\langle Lagentes \rangle ::= (\langle agente \rangle \text{ “\%” })^+$

$\langle agente \rangle ::= \text{ “\#” NAME } \text{crencas: } \langle Lcrencas \rangle \text{ objetivos: } \langle Lobjetivos \rangle \text{ planos: } \langle Lplanos \rangle$

$\langle Lcrencas \rangle ::= (\text{ “\{” } \langle nomeCrenca \rangle \text{ “;”}^* \text{ “\}” })$

$\langle nomeCrenca \rangle ::= \text{ NAME }$

$\langle Lobjetivos \rangle ::= (\text{ “\{” } \langle nomeObjetivo \rangle \text{ “;”}^* \text{ “\}” })$

$\langle nomeObjetivo \rangle ::= \text{ NAME }$

$\langle Lplanos \rangle ::= (\text{ “\{” } \langle nomePlano \rangle \text{ “;”}^* \text{ “\}” })$

$\langle nomePlano \rangle ::= \text{ NAME } \langle tuplaPlano \rangle$

$\langle tuplaPlano \rangle ::= \text{ “(” } \langle eventoGatilho \rangle \text{ “;” } \langle contexto \rangle \text{ “;” } \langle corpo \rangle \text{ “)” }$

$\langle eventoGatilho \rangle ::= \text{ NAME }$

$\langle contexto \rangle ::= \langle expressaoLogica \rangle \mid \text{ NAME } \mid \varepsilon$

$\langle expressaoLogica \rangle ::= \text{ NAME } \text{ “E” } \text{ NAME } \mid \text{ NAME } \text{ “OU” } \text{ NAME } \mid \text{ “NAO” } \text{ NAME }$

$\langle corpo \rangle ::= \text{ “\{” } \langle formulasCorpo \rangle \text{ “;”}^* \text{ “\}” }$

$\langle formulasCorpo \rangle ::= \text{ NAME }$

Alguns dos possíveis tokens a serem definidos:

- NUMBER
- NAME

Outros tokens podem ser necessários ou até mesmo não usar todos esses tokens sugeridos.

4. Exemplo de programa na Linguagem NAG:

Segue um exemplo de programa escrito na linguagem NAG, onde o nome do agente é bob.

```
#bob crencas: { estaChovendo ; naotenhoGuardaChuva ; }
objetivos: { comprarGuardaChuva ; naoPegarChuva ; }
planos: { plano1 ( comprarGuardaChuva ; estaChovendo E naotenhoGuardaChuva ;
{ sair; procurarLoja; comprarGuardaChuva; } ) ;
plano2 ( naoPegarChuva ; NAO estaChovendo ; { sair; jogarBola ; } ) ;
plano3 ( naoPegarChuva ; estaChovendo E naotenhoGuardaChuva ;
{ ficarEmCasa ; estudar ; } ) ; }
%
```

Observação: note que na tradução para Jason/AgentSpeak o nome do agente será usado para definir o nome do arquivo. Neste exemplo, seria `bob.asl`. Neste exemplo tem-se somente um único agente, mas o tradutor deve suportar mais agentes, neste caso todos agentes poderiam estar em único arquivo, exemplo `agentes.asl`, ou então em arquivos separados, exemplo `bob.asl`, `alice.asl` e `john.asl`.

5. Formato de envio:

- Note que todos arquivos (texto, relatório, slides) devem ser enviados em formato PDF (ou similar).
- Os arquivos de código fonte também devem ser anexados ao arquivo compactado que será submetido.
- O nome do arquivo submetido via Moodle deve ter o seguinte formato:

COMP-T1-Nome1-Nome2

- Trabalhos que não respeitem as restrições acima citadas estão sujeito a penalizações na nota final da equipe.

6. Observações importantes:

- (a) O trabalho deve ser individualmente ou em dupla (*com dois alunos*).
- (b) A penalização da apresentação pode ser aplicada individualmente ou para os dois integrantes da equipe.
- (c) Faz parte do trabalho pesquisar a respeito da linguagem Jason/AgentSpeak e do framework JACAMO. Porém, será fornecido material de apoio no Moodle além dos links já descritos aqui.