

Processamento de Linguagens I
LESI + LMCC (3º ano)

Lavanda
Exercícios com Gramáticas de Atributos (LISA)
Geração de Código da VM

Language.Processing@@di.um.pt
28 de Maio de 2006

Conteúdo

1	Gramática de Atributos em LISA	2
2	Resultado da Execução: código VM gerado	6

Capítulo 1

Gramática de Atributos em LISA

Neste documento pretende-se ilustrar:

- a utilização da máquina virtual VM;
- a utilização de uma GA para automatizar a geração de código.

Para o efeito retoma-se o exercício **Lavanda** e vai escrever-se uma GA para especificar formalmente esse cálculo e a produção do respectivo código assembly da VM de modo a calcular o número de sacos enviados e o número de lotes de cada saco.

Apresenta-se, a seguir a respectiva GA, escrita na sintaxe do sistema LISA de modo a poder-se recorrer a este sistema para gerar automaticamente o compilador pretendido.

```
language Lavanda1
{
  lexicon
  {
    ReservedWord corpo | casa | br | cor | alg | fib | la
    Number      [0-9]+
    Data        [0-2] [0-9] \- [0-9] [0-9] \- [0-2] [0-9] [0-9] [0-9]
    Identifier  [a-z]+
    separa      \( | \) | \, | \> | \-
    ignore      [\x09\x0A\x0D\ ]+
  }

  attributes String  TIPO.name, CLASSE.name, TINTO.name, FIO.name,
                     LAVANDA.output, CABEC.output, SACOS.output, SACO.output,
                     SACOS.outLotes, SACO.outLotes, LOTES.outLotes, LOTE.outLotes, OUTROS.outLotes

  rule Lavanda {
    LAVANDA ::= CABEC SACOS compute {
      LAVANDA.output = escreveFicheiro("// " + CABEC.output + "\n" +
        "START\n" +
        SACOS.output +
        "STRI\n" +
        "PUSHS \"Numero de sacos: \"\n" +
        "CONCAT\n" +
        "WRITES\n" +
        SACOS.outLotes +
        "STOP\n");
    }
```

```

    };
}

rule Cabec {
    CABEC ::= #Data #Identifier compute {
        CABEC.output = #Data.value() + " PR: " + #Identifier.value();
    };
}

rule Sacos1 {
    SACOS ::= SACO compute {
        SACOS.output = SACO.output;
        SACOS.outLotes = SACO.outLotes;
    };
}

rule Sacos2 {
    SACOS ::= SACOS SACO compute {
        SACOS[0].output = SACOS[1].output + SACO.output + "ADD\n" ;
        SACOS[0].outLotes = SACOS[1].outLotes + SACO.outLotes;
    };
}

rule Saco {
    SACO ::= #Number #Identifier \( LOTES \) compute {
        SACO.output = "PUSHI 1\n";
        SACO.outLotes = LOTES.outLotes +
            "STRI\n" +
            "PUSHS \"Numero de Lotes no sacco n. \" + #Number.value() + \": \"\n\" +
            "CONCAT\n\" + "WRITES\n";
    };
}

rule Lotes {
    LOTES ::= LOTE OUTROS compute {
        LOTES.outLotes = LOTE.outLotes + OUTROS.outLotes;
    };
}

rule Lote {
    LOTE ::= TIPO #Number compute {
        LOTE.outLotes = "PUSHI 1\n";
    };
}

rule Outros {
    OUTROS ::= compute {
        OUTROS.outLotes = "";
    }
    | \, LOTES compute {
        OUTROS.outLotes = LOTES.outLotes + "ADD\n";
    }
}

```

```

        };
    }

rule Tipo {
    TIPO ::= CLASSE \- TINTO \- FIO compute {
        TIPO.name = CLASSE.name + "/" + TINTO.name + "/" + FIO.name;
    };
}

rule Classe {
    CLASSE ::= corpo compute {
        CLASSE.name = "corpo";
    }
    | casa compute {
        CLASSE.name = "casa";
    };
}

rule Tinto {
    TINTO ::= br compute {
        TINTO.name = "br";
    }
    | cor compute {
        TINTO.name = "cor";
    };
}

rule Fio {
    FIO ::= alg compute {
        FIO.name = "alg";
    }
    | la compute {
        FIO.name = "la";
    }
    | fib compute {
        FIO.name = "fib";
    };
}

method Print
{
    import java.io.*;

    public String escreveFicheiro(String res)
    {
        try
        {
            FileWriter fw = new FileWriter("LavandaResultado.vm");
            fw.write(res);

```

```
        fw.close();
        return "Ficheiro criado com sucesso!\n";
    }
    catch (Exception e)
    {
        return (e.toString());
    }
}

}
```

Capítulo 2

Resultado da Execução: código VM gerado

Consideremos como teste o seguinte exemplo:

```
10-11-2005 today 1 dani  (corpo-cor-la 1 , casa-cor-alg 2)
                  2 pedro (casa-br-fib 4)
                  3 celina (corpo-cor-alg 2, corpo-cor-la 3, corpo-cor-fib 1,
                           casa-cor-alg 2, casa-cor-la 3, casa-cor-fib 1)
```

O compilador, gerado automaticamente pelo LISA a partir da GA apresentada na sec. anterior, ao receber, como texto de entrada, o ficheiro de teste acima vai gerar o seguinte código assembly da máquina virtual VM:

```
// 10-11-2005 PR: ecosky
START
PUSHI 1
PUSHI 1
ADD
PUSHI 1
ADD
STRI
PUSHS "Numero de sacos: "
CONCAT
WRITES
PUSHI 1
PUSHI 1
ADD
STRI
PUSHS "Numero de Lotes no saco n. 1: "
CONCAT
WRITES
PUSHI 1
PUSHI 1
PUSHI 1
ADD
```

```
ADD
STRI
PUSHS "Numero de Lotes no saco n. 2: "
CONCAT
WRITES
PUSHI 1
STRI
PUSHS "Numero de Lotes no saco n. 3: "
CONCAT
WRITES
STOP
```