

Compiladores, 2017/2018

Aula prática 2

Fernando Lobo

Prazos

O exercício 3 conta para avaliação da componente prática e deve ser entregue pelo mooshak. Devem escolher um grupo através da tutoria até ao final do prazo de entrega. O grupo manter-se-á até ao final da disciplina.

- **Pontuação máxima:** 15 pontos
- **Prazo de entrega:** 20/Fev/2018, 23:59
- **Entrega:** pela mooshak, concurso **Comp1718**, problema A.
<http://mooshak.deei.fct.ualg.pt/~mooshak/>
- **Login mooshak:** o vosso login no mooshak deve ser **Gxx** onde xx é o vosso número de grupo.

Exercício 1

Descreva em linguagem corrente o significado de cada uma das seguintes expressões regulares. [*Dragon book, 2nd edition, pp 125, Exercise 3.3.2 (a,c,d)*]

- $0(0|1)^*0$
- $(0|1)^*0(0|1)(0|1)$
- $0^*10^*10^*10^*$

Exercício 2

Escreva expressões regulares para as seguintes linguagens. [*Dragon book, 2nd edition, pp 125, Exercise 3.3.5 (a,b,h)*]

- Todas as strings de letras minúsculas que contêm algures uma e uma só ocorrência de cada uma das cinco vogais, e cada uma delas aparece por ordem lexicográfica crescente.
- Todas as strings de letras minúsculas, com estas a aparecem por ordem lexicográfica crescente.
- Todas as strings de 0's e 1's que não contêm a substring 011.

Exercício 3

Construa um autómato finito para reconhecer os seguintes tokens de uma linguagem para uma calculadora que para além de operações aritméticas habituais, permite usar variáveis e ainda instruções de leitura (read) e escrita (write). Os tokens da linguagem são definidos pelas seguintes expressões regulares.

```
assign  -->  ':='  
plus    -->  '+'  
minus   -->  '-'  
times   -->  '*'  
div     -->  '/'  
lparen  -->  '('  
rparen  -->  ')'  
keyword -->  'read' | 'write'  
id      -->  letter (letter | digit)* , excepto para 'read' e 'write'  
number  -->  digit digit* | digit* ('.' digit | digit '.') digit*
```

Considere ainda que `letter` e `digit` são definidos da seguinte forma:

```
letter  -->  'a'|'b'|'c'|...|'z'|'A'|'B'|...|'Z'  
digit   -->  '0'|'1'|...|'9'
```

Após construir o autómato, faça um programa em Java para reconhecer os tokens de um determinado input. Devem submeter o vosso programa no Mooshak, concurso **Comp1718**. Deixo-vos 2 exemplos de input e output respectivo.

input 1

```
read A  
read B  
sum := A + B*99  
write sum/2
```

output 1

```
<keyword,"read">  
<id,"A">  
<keyword,"read">  
<id,"B">  
<id,"sum">  
<assign,":=">  
<id,"A">  
<plus,"+">  
<id,"B">  
<times,"*">  
<number,"99">
```

```
<keyword,"write">
<id,"sum">
<div,"/">
<number,"2">
```

input 2

```
read A
read readd
sum := A+B * ( 99.1 +.44      444 1234+++ )
write sum
write sum/232.123459
```

output 2

```
<keyword,"read">
<id,"A">
<keyword,"read">
<id,"readd">
<id,"sum">
<assign,":=">
<id,"A">
<plus,"+">
<id,"B">
<times,"*">
<lparen,"(">
<number,"99.1">
<plus,"+">
<number,".44">
<number,"444">
<number,"1234">
<plus,"+">
<plus,"+">
<plus,"+">
<rparen,")">
<keyword,"write">
<id,"sum">
<keyword,"write">
<id,"sum">
<div,"/">
<number,"232.123459">
```