

Universidade Federal de São Carlos – Departamento de Computação
Construção de Compiladores e Construção de Compiladores 1
Profa. Helena Caseli

Primeira Lista de Exercícios – Introdução e Análise Léxica

- 1) O que é um compilador?
- 2) Quais são as 6 etapas em uma estrutura padrão de tradução realizada por um compilador? Descreva brevemente o que acontece em cada uma delas.
- 3) O que vem a ser “passada” e qual a vantagem e a desvantagem de um compilador que realiza várias passadas em relação a um compilador de apenas uma passada?
- 4) O que é um interpretador e quais suas diferenças em relação a um compilador?
- 5) Qual é a função da etapa de análise léxica no processo de tradução de um compilador?
- 6) Identifique todos os *tokens* que compõem os programas seguintes informando, para cada um, a cadeia (lexema) correspondente e sua classe (identificador, palavra ou símbolo reservado, número etc.).
 - a) Pascal

```
function max(i, j: integer): integer;  
{ retorna o maior dos inteiros entre i e j}  
begin  
    if i > j then max := i  
    else max := j  
end;
```
 - b) C

```
int max(i, j) int i, j;  
/* retorna o maior dos inteiros entre i e j */  
{  
    return i > j ? i : j  
}
```
 - c) Em qual dos dois programas apresentados nas letras acima (a e b) foram identificados mais *tokens*?
- 7) Se a análise léxica é feita sob o comando da análise sintática, então quais são os motivos para se separar conceitualmente a análise léxica da sintática? Explique cada um dos motivos.
- 8) Escreva expressões regulares para os conjuntos de caracteres a seguir ou se não for possível escrever uma expressão regular para um determinado conjunto de caracteres, justifique.
 - a) Cadeias de letras maiúsculas começando e terminando com *a* (minúsculo).
 - b) Cadeias de dígitos que representam números pares.
 - c) Cadeias de 0s e 1s com um número par de 0s.
 - d) Cadeias de 0s e 1s nas quais os 0s ocorrem em pares (um 0 seguido de outro 0).
 - e) Cadeias de 0s e 1s compostas por um único 1 rodeado pelo mesmo número de 0s à esquerda e à direita.

f) Cadeias de dígitos tais que todos os dígitos ímpares, se ocorrerem, ocorrem antes de todos os dígitos pares (se ocorrerem).

9) Construa os autômatos correspondentes para as expressões regulares descritas no exercício 8. Assuma que os autômatos construídos serão implementados em um sistema de varredura (análise léxica) e que, portanto, a regra de reconhecimento da maior cadeia possível deve ser seguida (cuidado ao definir o estado final).

10) Escreva o pseudocódigo (algoritmo) para o reconhecimento de cada conjunto de cadeias do exercício 8, ou seja, faça o mapeamento dos autômatos descritos no exercício 9 em código usando ifs ou cases como apresentado em aula.