



Compiladores Analizador Léxico



Eduardo Marques

Analizador Léxico

- ▶ Gerar sequência de *tokens* a partir do código fonte
 - ▶ Remover comentários, espaços
- ▶ Pode ser uma sub-rotina ou co-rotina do analisador sintático
- ▶ Interface com o código fonte
 - ▶ Relacionar tokens com número de linha
- ▶ Atenção com performance
 - ▶ Acesso a disco
 - ▶ Buferização



Analizador Léxico

- ▶ Tokens

- ▶ menores unidades de informação que constituem a linguagem

- ▶ Padrão

- ▶ Regra que descreve as cadeias de entrada associado a um *token*

- ▶ Lexema

- ▶ Conjunto de caracteres no programa-fonte que é reconhecido pelo padrão de algum *token*



Analizador Léxico

► Exemplo

► `const pi = 3,1415927;`

Token	Lexema	Padrão
const	const	const
identificador	pi	letra seguida por letras e/ou dígitos
relacional	=	< ou > ou = ou <> ou <= ou >=
numero	3,1415927	Qualquer constante numérica
ponto-e-virgula	;	;



Analizador Léxico

► Erros

- Os léxicos só podem ser detectados na análise sintática ou semântica

- Ex.: `fi (a == 25)`

Token	Lexema	Padrão
Identificador	fi	letra seguida por letras e/ou dígitos
Abreparentese	((
Identificador	a	letra seguida por letras e/ou dígitos
relacional	==	< ou > ou == ou != ou <= ou >=
numero	25	Qualquer constante numérica
fechaparentese))



► Algoritmo Analisador Léxico

Início

Abre arquivo fonte

Enquanto não acabou o arquivo fonte

Faça {

Trata Comentário e Consome espaços

Pega Token

Identifica Token

Coloca Token na Lista de Tokens

}

Fecha arquivo fonte

Fim



► Algoritmo Comentário e Consome espaços

Início

Se início de comentário

Então{

 Enquanto diferente de fim de comentario

 Faça {

 pula comentario

 }

}

Enquanto espaço, quebra de linha ou tabulação Faça {

 pula espaço, quebra de linha ou tabulação

}

Fim



► Algoritmo Pega Token

Inicio

```
Se letra Então{  
    Enquanto letra ou número Faça  
        concatena letra  
    retorna token  
}  
Se número Então{  
    Enquanto número Faça  
        concatena número  
    retorna token  
}  
Se caracter especial Então {  
    trata caracter especial  
    retorna token  
}
```

► Fim

Analizador Léxico - Gerador

- ▶ Ferramenta para gerar analisadores léxicos a partir do conjunto de símbolos terminais da gramática
- ▶ Utilizam expressões regulares
- ▶ Exemplos:
 - ▶ Lex – presente no UNIX (em C)
 - ▶ Flex – presente no Linux (em C)
 - ▶ TPly – parser em PASCAL
 - ▶ JavaCC – parser em Java
 - ▶ Flex++ ou Flexx – parser em C++

