



Documentação Léxica da Linguagem GO

1. Introdução

Go é uma linguagem de programação criada pela Google e lançada em código livre em novembro de 2009. É uma linguagem compilada e focada em produtividade e programação concorrente, baseada em trabalhos feitos no sistema operacional chamado Inferno.

A sintaxe de Go é semelhante a C; uma variação é a declaração de tipos, a ausência de parênteses em volta das estruturas *for* e *if*. Seu modelo de concorrência é baseado no CSP de Tony Hoare, além de possuir características do cálculo pi.

2. Lexemas da GO

2.1 Comentários

Em GO podemos ter dois tipos de comentários. Comentários de linha e comentários de bloco.

```
// Exemplo de comentário em linha
/* Exemplo
   De Comentário
   Multi-linha */
```

2.2 Palavras reservadas

A seguir apresentamos a lista de palavras reservadas dessa linguagem:

break	default	func	make	case
defer	map	struct	else	package
switch	const	fallthrough	if	range
type	continue	for	import	return
var				

2.3 Operadores

A seguir são apresentados os operadores válidos na linguagem

+	+=	&&	==	!=	-=
	-	<	<=	*	*=
<-	>	>=	/	++	/=
%=	--	=	:=	%	!

2.4 Tipos de dados primitivos

```
var x int = 0
var y float64 = 5.8
var str string = "Olá"
var par bool = true
```

- Boolean - *true* ou *false*
- Numéricos - valores do tipo inteiro, ponto flutuante
- Strings - arrays imutáveis

2.5 Tipos de dados compostos

➤ Structs

```
type Aluno struct {
    Nome string
    matricula int
}
```

➤ Arrays

```
pares := [3]int{2, 6, 8}
```

➤ Slice

```
p := make([]string, 3)
```

➤ Mapas

```
m := make(map[string]int)
```

2.6 Variáveis

- Comando *var* declara uma ou mais variáveis

```
var a string = "nome" //char a[]="nome";  
var a, b int 1, 2 // int a=1; int b=2;
```
- Há inferência de tipo

```
var c = false //infere booleano
```
- Variáveis declaradas sem valor recebem valor-zero

```
var d int //int d=0;
```
- O comando de atribuição *:=* também pode ser utilizado para declarar e inicializar uma variável

```
e := 5 //Equivalente a var e int = 5
```