

Programação Funcional

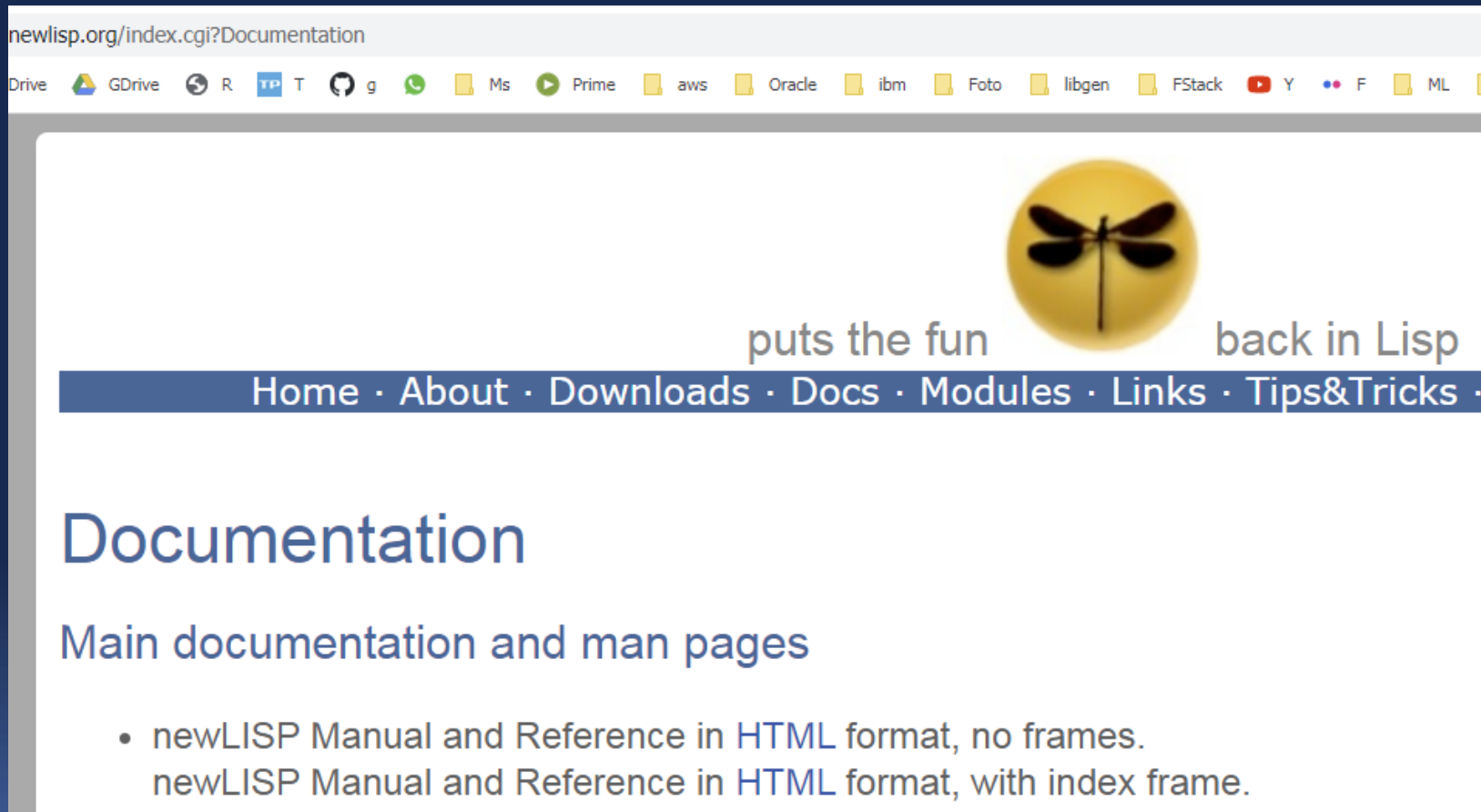
Unidade 3 – Programação Funcional com a Linguagem LISP



Prof. Aparecido V. de Freitas
Doutor em Engenharia
da Computação pela EPUSP
aparecidovfreitas@gmail.com

Bibliografia

<http://www.newlisp.org/index.cgi?Documentation>



The screenshot shows a web browser window with the address bar displaying `newlisp.org/index.cgi?Documentation`. The browser's toolbar includes icons for Drive, GDrive, R, TP, T, g, a green circle, Ms, Prime, aws, Oracle, ibm, Foto, libgen, FStack, Y, F, and ML. The website content features a yellow circular logo with a black dragonfly. Below the logo, the text "puts the fun" and "back in Lisp" is displayed. A dark blue navigation bar contains the links: Home · About · Downloads · Docs · Modules · Links · Tips&Tricks ·. The main heading is "Documentation" in a large, dark blue font. Below it, the text "Main documentation and man pages" is shown. A bulleted list contains two items: "newLISP Manual and Reference in HTML format, no frames." and "newLISP Manual and Reference in HTML format, with index frame."

newlisp.org/index.cgi?Documentation

Drive GDrive R TP T g Ms Prime aws Oracle ibm Foto libgen FStack Y F ML

puts the fun back in Lisp

Home · About · Downloads · Docs · Modules · Links · Tips&Tricks ·

Documentation

Main documentation and man pages

- newLISP Manual and Reference in HTML format, no frames.
- newLISP Manual and Reference in HTML format, with index frame.

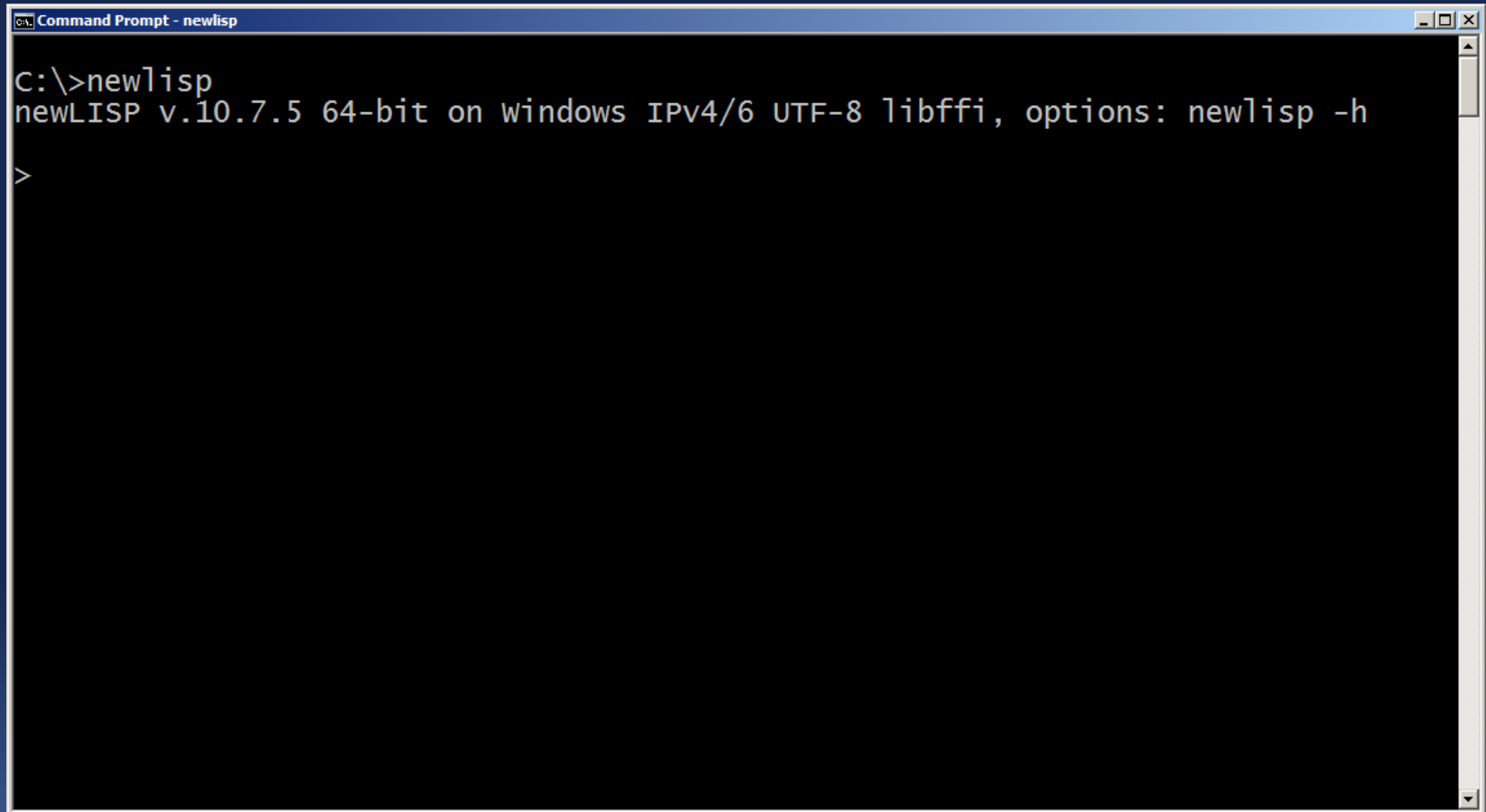


Downloading e Instalação

- ✓ O download do **newLISP** pode ser feito em <http://www.newlisp.org/>
- ✓ Disponível em diversas plataformas;
- ✓ O interpretador poderá ser chamado diretamente pela linha de comandos, em uma console;
- ✓ O código fonte pode ser gerado a partir de um editor de programas, como por exemplo o **Notepad++**.



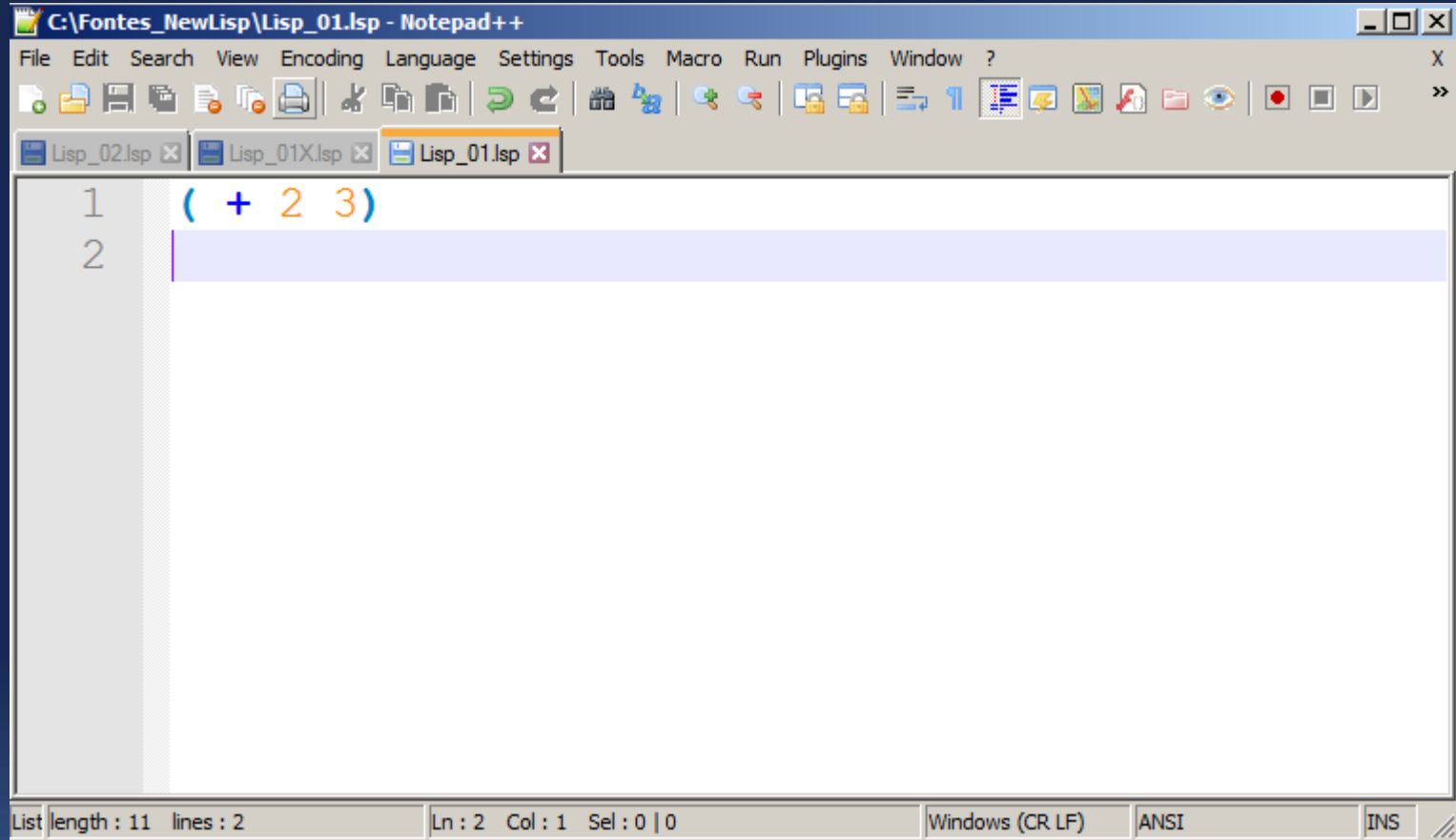
Introdução



```
Command Prompt - newlisp
C:\>newlisp
newLISP v.10.7.5 64-bit on windows IPv4/6 UTF-8 libffi, options: newlisp -h
>
```



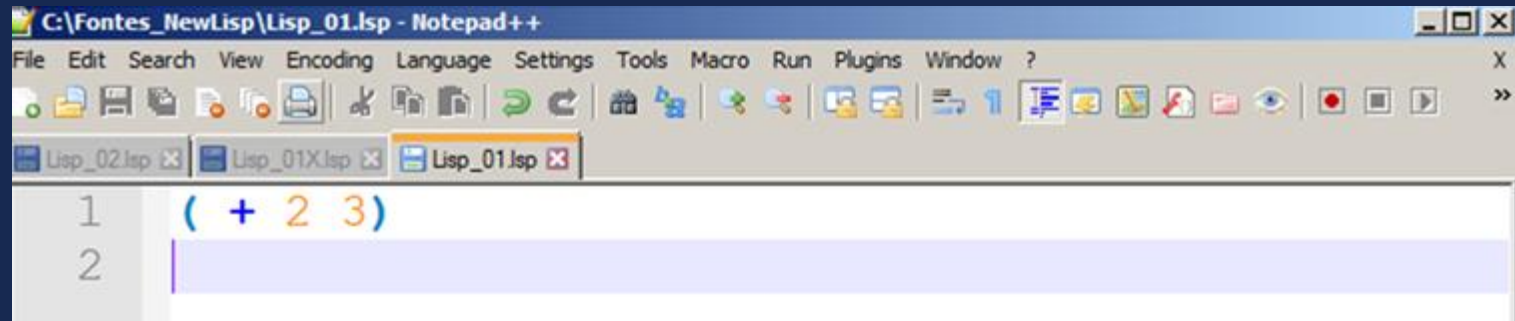
Exemplo – Lisp_O1.lsp



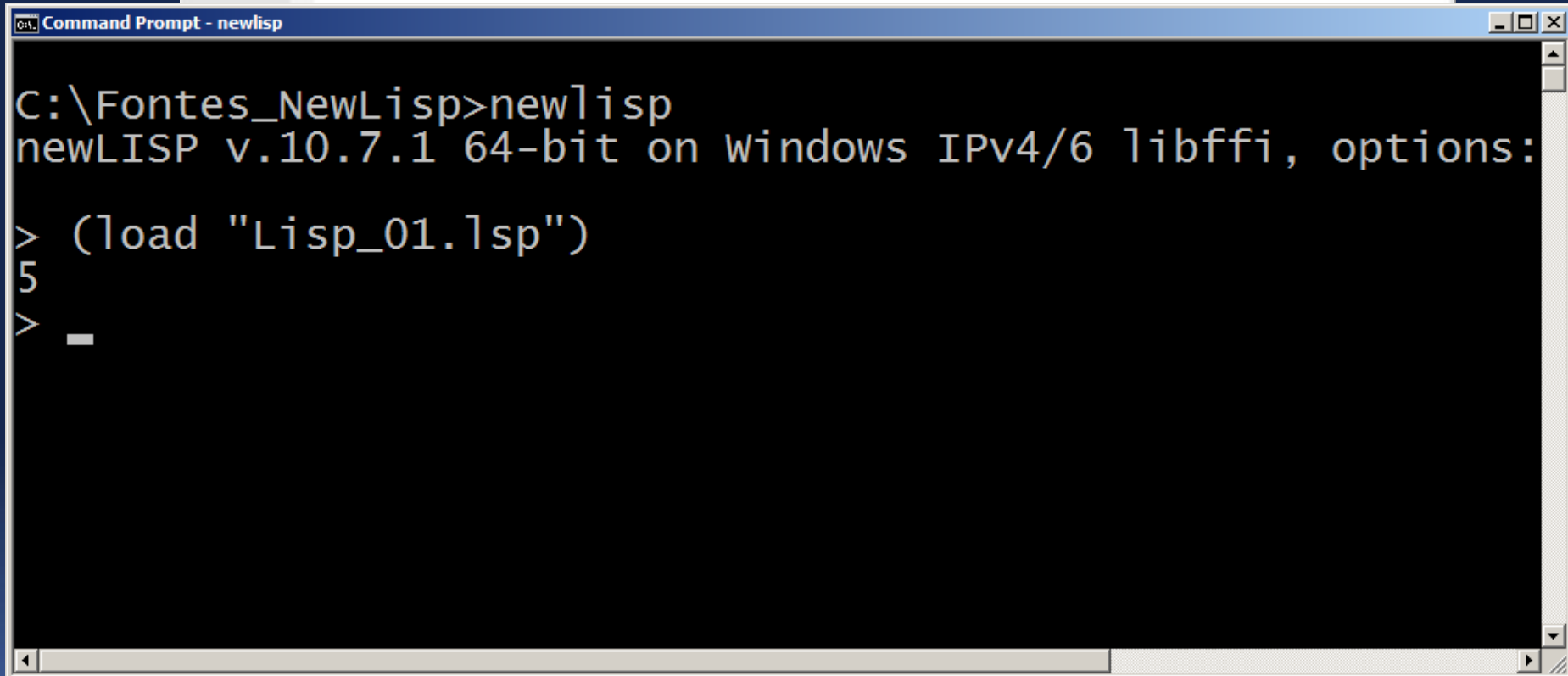
```
1 ( + 2 3 )
2
```



Exemplo – Lisp_O1.lsp



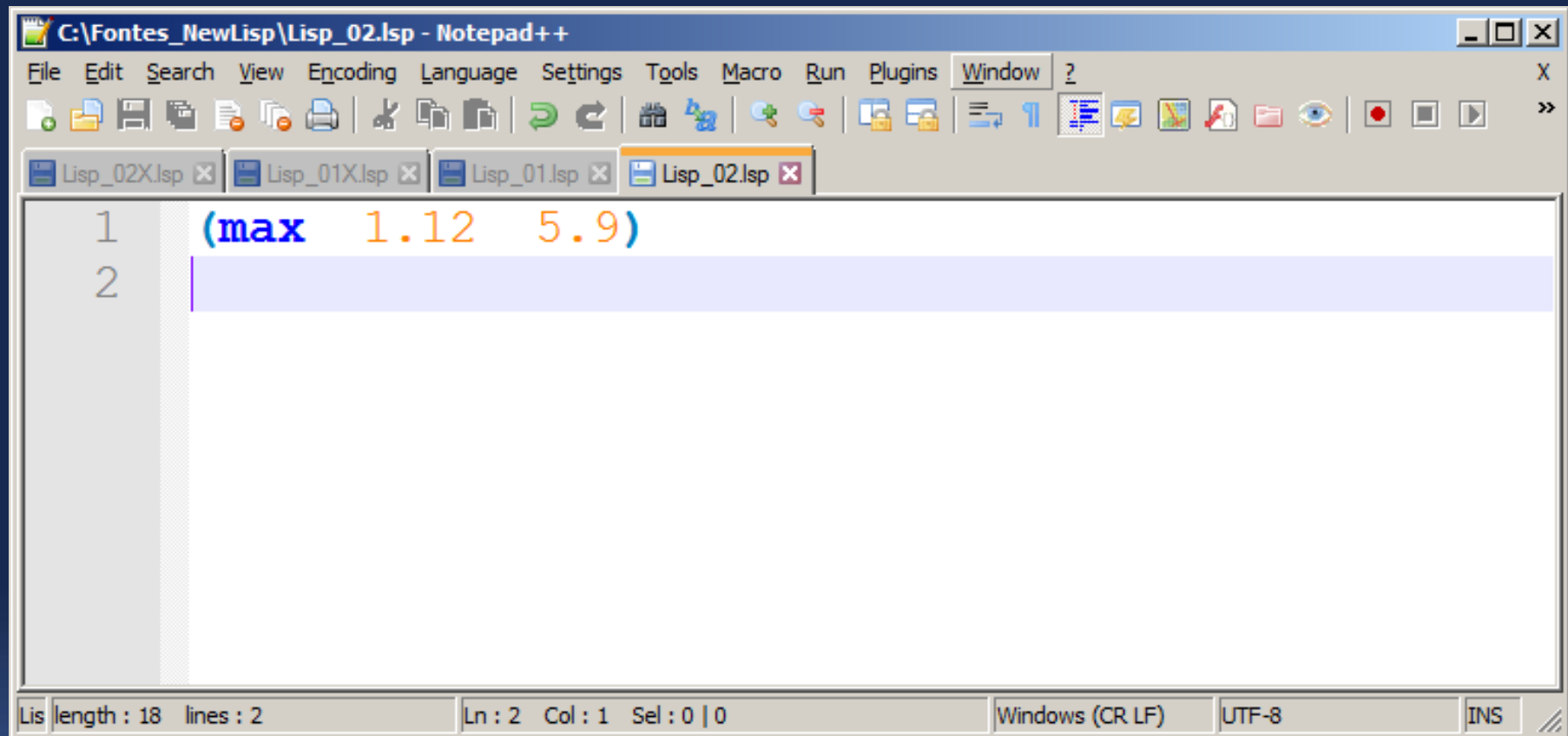
```
C:\Fontes_NewLisp\Lisp_01.lsp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Lisp_02.lsp Lisp_01X.lsp Lisp_01.lsp
1 ( + 2 3 )
2
```



```
C:\Fontes_NewLisp>newLisp
newLISP v.10.7.1 64-bit on windows IPv4/6 libffi, options:
> (load "Lisp_01.lsp")
5
> _
```

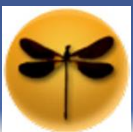


Exemplo – Lisp_02.lsp

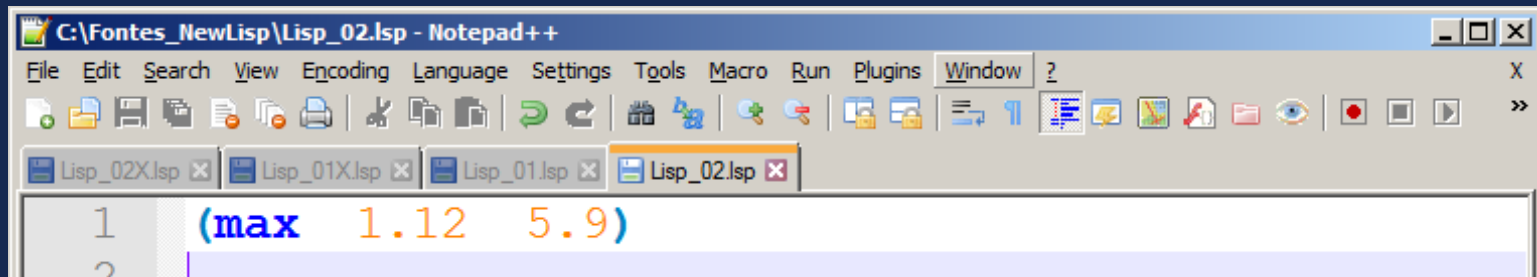


The screenshot shows a Notepad++ window titled "C:\Fontes_NewLisp\Lisp_02.lsp - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, Window, and Help. The toolbar contains various icons for file operations, editing, and development. The tab bar shows four open files: Lisp_02X.lsp, Lisp_01X.lsp, Lisp_01.lsp, and Lisp_02.lsp. The main text area contains two lines of code: line 1 is `(max 1.12 5.9)` and line 2 is empty. The status bar at the bottom displays "Ln : 2 Col : 1 Sel : 0 | 0", "Windows (CR LF)", "UTF-8", and "INS".

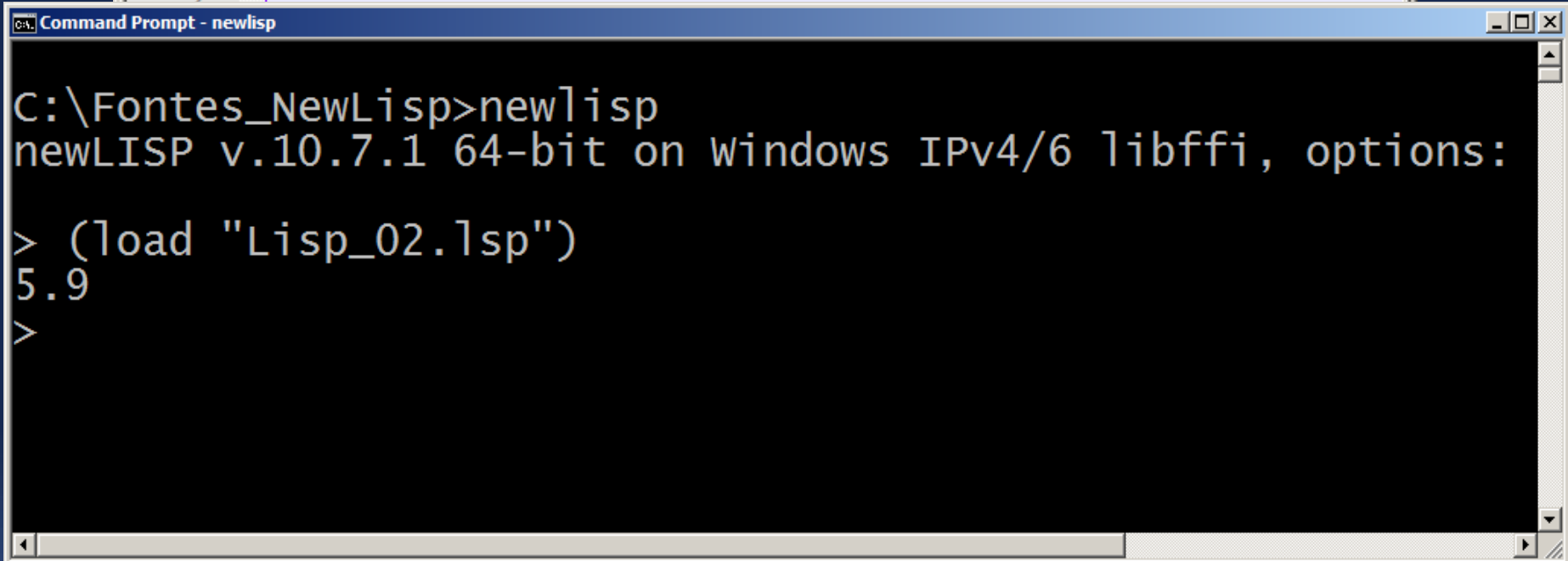
```
1 (max 1.12 5.9)
2
```



Exemplo – Lisp_02.lsp



```
C:\Fontes_NewLisp\Lisp_02.lsp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Lisp_02X.lsp Lisp_01X.lsp Lisp_01.lsp Lisp_02.lsp
1 (max 1.12 5.9)
2
```



```
C:\Fontes_NewLisp>newlisp
newLISP v.10.7.1 64-bit on windows IPv4/6 libffi, options:
> (load "Lisp_02.lsp")
5.9
>
```



As três regras básicas de newLISP

Regra 1

- ✓ Uma **lista** é uma sequência de elementos delimitados entre parênteses;
- ✓ **Lista** é a estrutura básica do newLISP;
- ✓ Exemplos:

```
NGOANDSCS.NOTAS x settings.xml pom.xml server.xml Tarefa_T8_Solucao_SCRIPT.sql Tarefa_T9_Solucao_SCRIPT.sql Tarefa_T10_Solucao_SCRIPT.sql synced.rkt blueboxes.rkt NewLISP_02.lsp x
```

(1 2 3 4 5)	;uma lista de inteiros
("o" "gato" "triste")	;uma lista de strings
(x y z a)	;uma lista de nomes simbólicos
(sin cos tan)	;uma lista de funções newlisp
(1 2 "teste" x sin)	;uma lista mista
(1 2 (1 2 3) 3 4)	;uma lista com uma lista dentro dela
((1 2) (3 4) (5 6))	;uma lista de listas



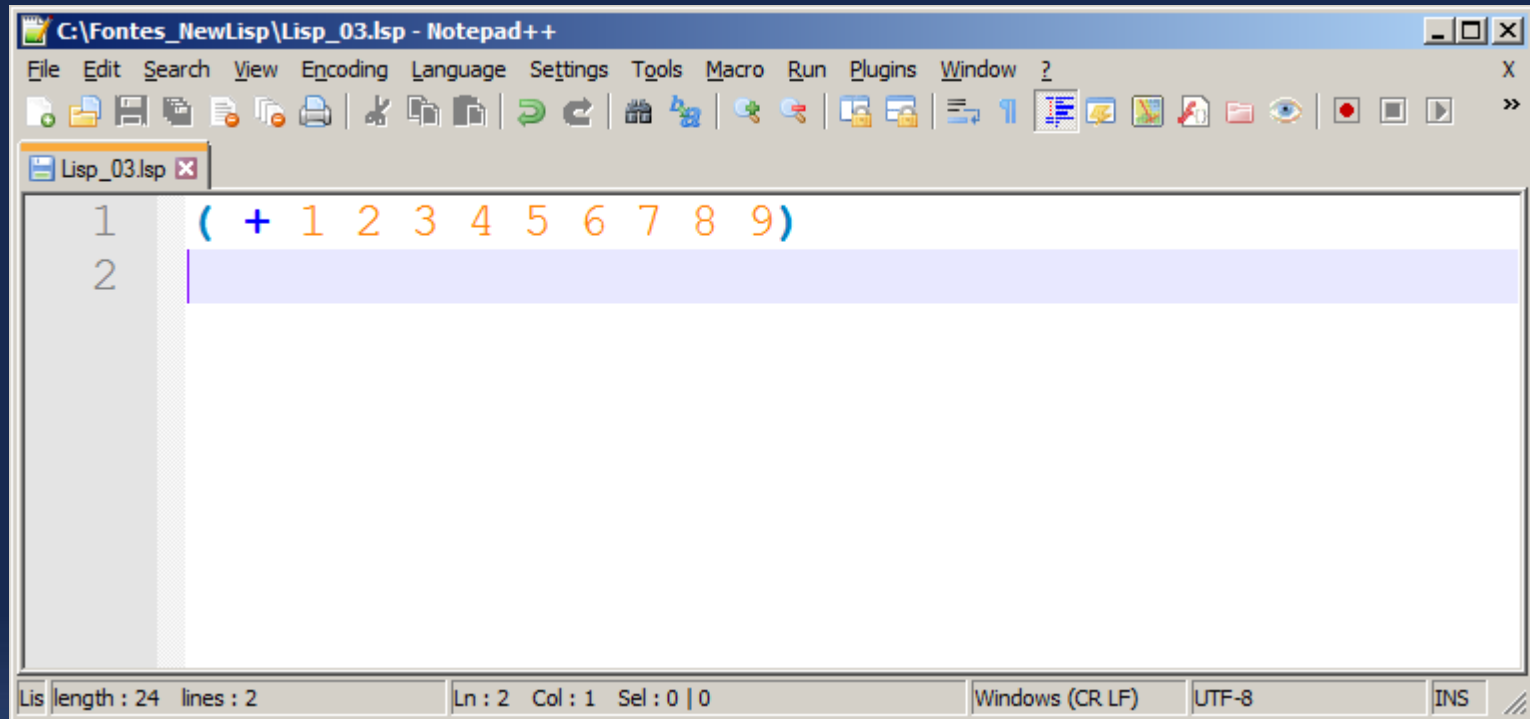
As três regras básicas de newLISP

Regra 2

- ✓ Para **newLISP**, o **primeiro elemento** da lista tem um significado especial;
- ✓ Quando **newLISP** recebe uma lista, ele trata sempre o primeiro elemento como uma função e tenta utilizar os demais elementos como argumentos;
- ✓ **(+ 5 4)** é uma lista com três elementos. Uma função chamada +, seguida por 2 valores numéricos;
- ✓ Quando **newLISP** recebe a lista acima, ele a avalia e retorna o valor **9**;



Exemplo – Lisp_03.lsp



```
C:\Fontes_NewLisp\Lisp_03.lsp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Lisp_03.lsp
1 ( + 1 2 3 4 5 6 7 8 9 )
2
Lisp length : 24 lines : 2 Ln : 2 Col : 1 Sel : 0 | 0 Windows (CR LF) UTF-8 INS
```



Exemplo – Lisp_O2A.lsp

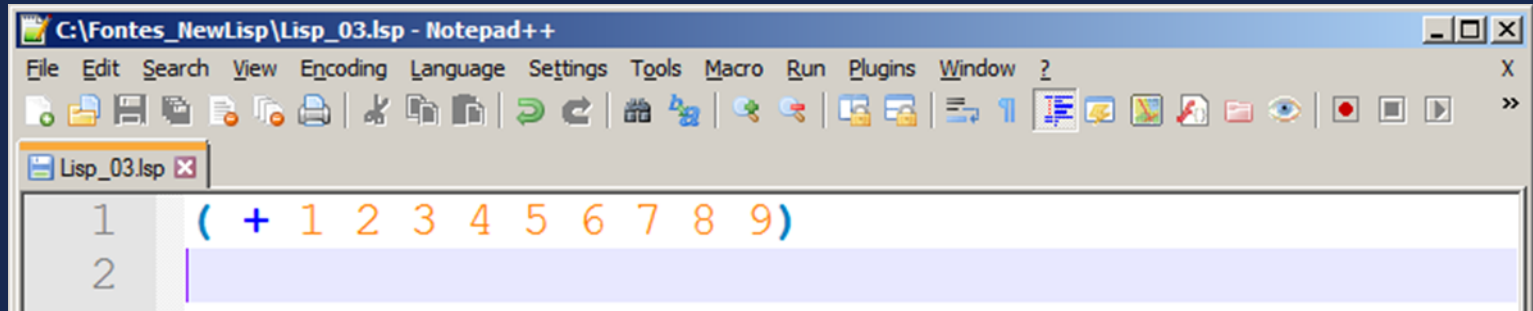
ENG0ANDSCS.NOTAS x settings.xml x pom.xml x server.xml x Tarefa_T8_Solucao_SCRIPT.sql x Tarefa_T9_Solucao_SCRIPT.sql x

```
(max 1 4.5 9.88 12 999 45.000 65 )
```

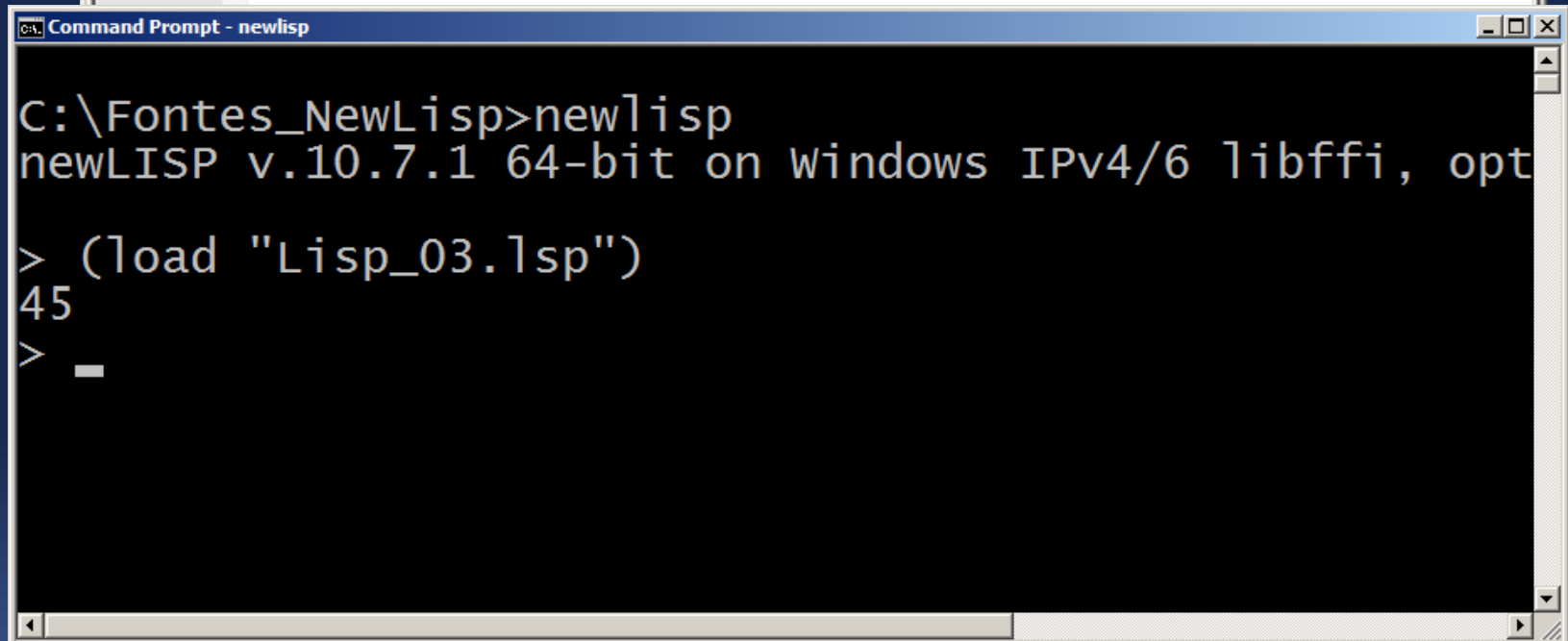
```
Command Prompt - newlisp "Lisp_2A.lsp")
>
>
>
>
>
>
> (load "newlisp_02A.lsp")
999
>
```



Exemplo – Lisp_03.lsp



A screenshot of a Notepad++ window titled "C:\Fontes_NewLisp\Lisp_03.lsp - Notepad++". The window contains two lines of code: line 1 is "(+ 1 2 3 4 5 6 7 8 9)" and line 2 is empty. The code is color-coded: parentheses are blue, the plus sign is red, and the numbers are green.



A screenshot of a Command Prompt window titled "Command Prompt - newlisp". The prompt is "C:\Fontes_NewLisp>". The user has entered "newlisp", and the output is "newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, opt". The user then enters "> (load \"Lisp_03.lsp\")", and the output is "45". The prompt is now "> _".



Exemplo – Lisp_O4.lsp

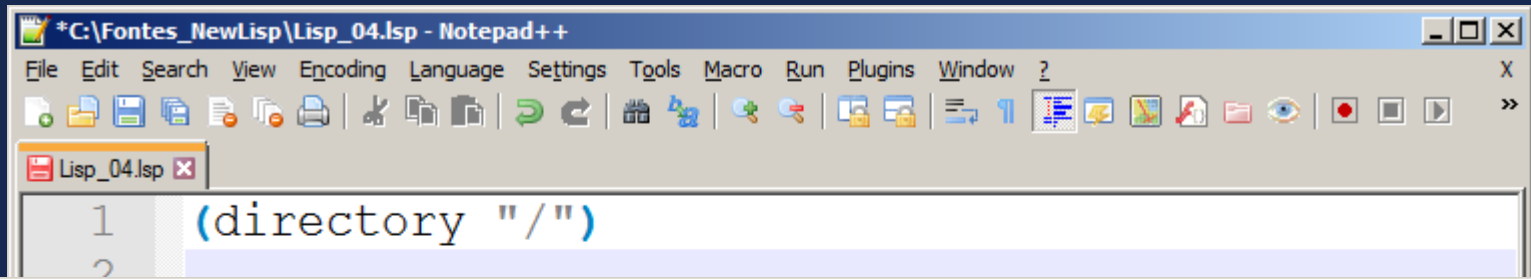
)" data-bbox="131 275 929 775"/>

The screenshot shows a Notepad++ window titled "*C:\Fontes_NewLisp\Lisp_O4.lsp - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, Window, and Help. The toolbar contains various icons for file operations, editing, and development. The editor area shows a single line of Lisp code: `(directory "/")`. The status bar at the bottom displays "Lisp length : 18 lines : 2", "Ln : 2 Col : 1 Sel : 0 | 0", "Windows (CR LF)", "UTF-8", and "INS".

```
1 (directory "/")
2
```

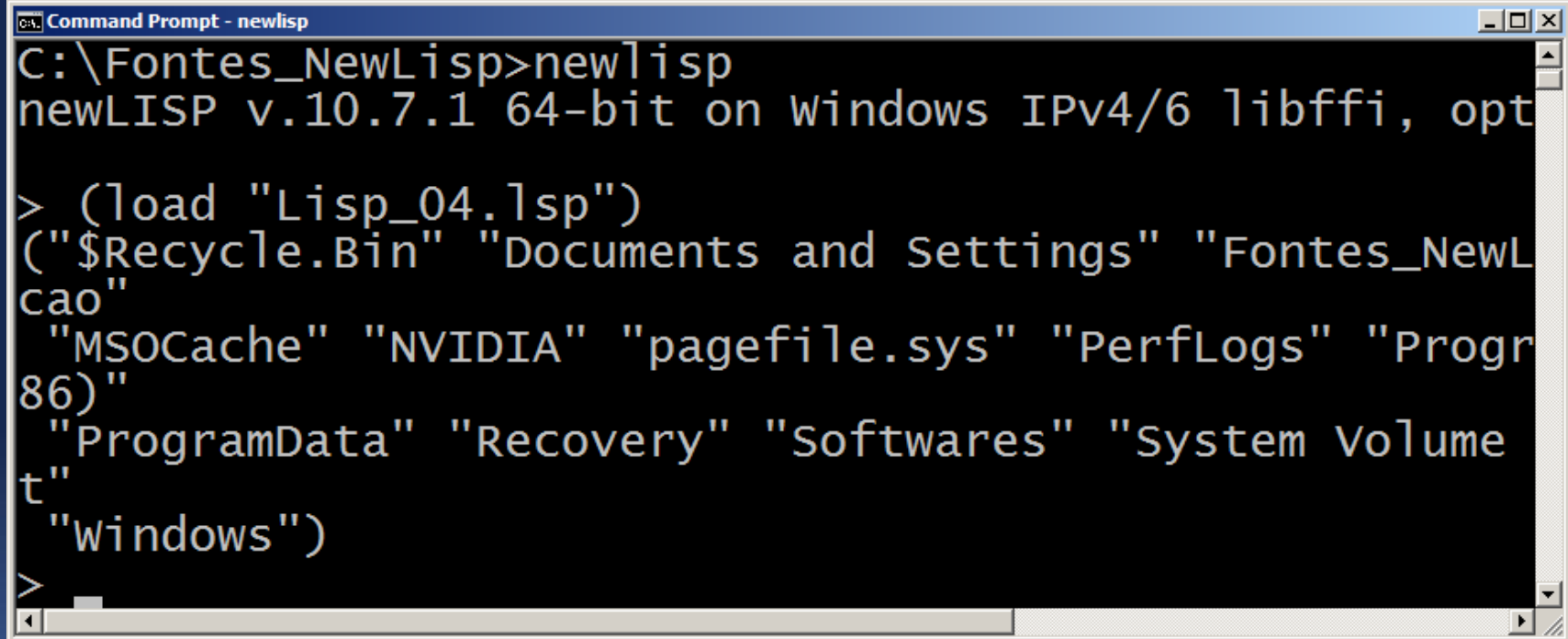


Exemplo – Lisp_O4.lsp



The screenshot shows a Notepad++ window titled '*C:\Fontes_NewLisp\Lisp_O4.lsp - Notepad++'. The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, and Window. The toolbar contains various icons for file operations and editing. The text area shows the following code:

```
1 (directory "/")
2
```



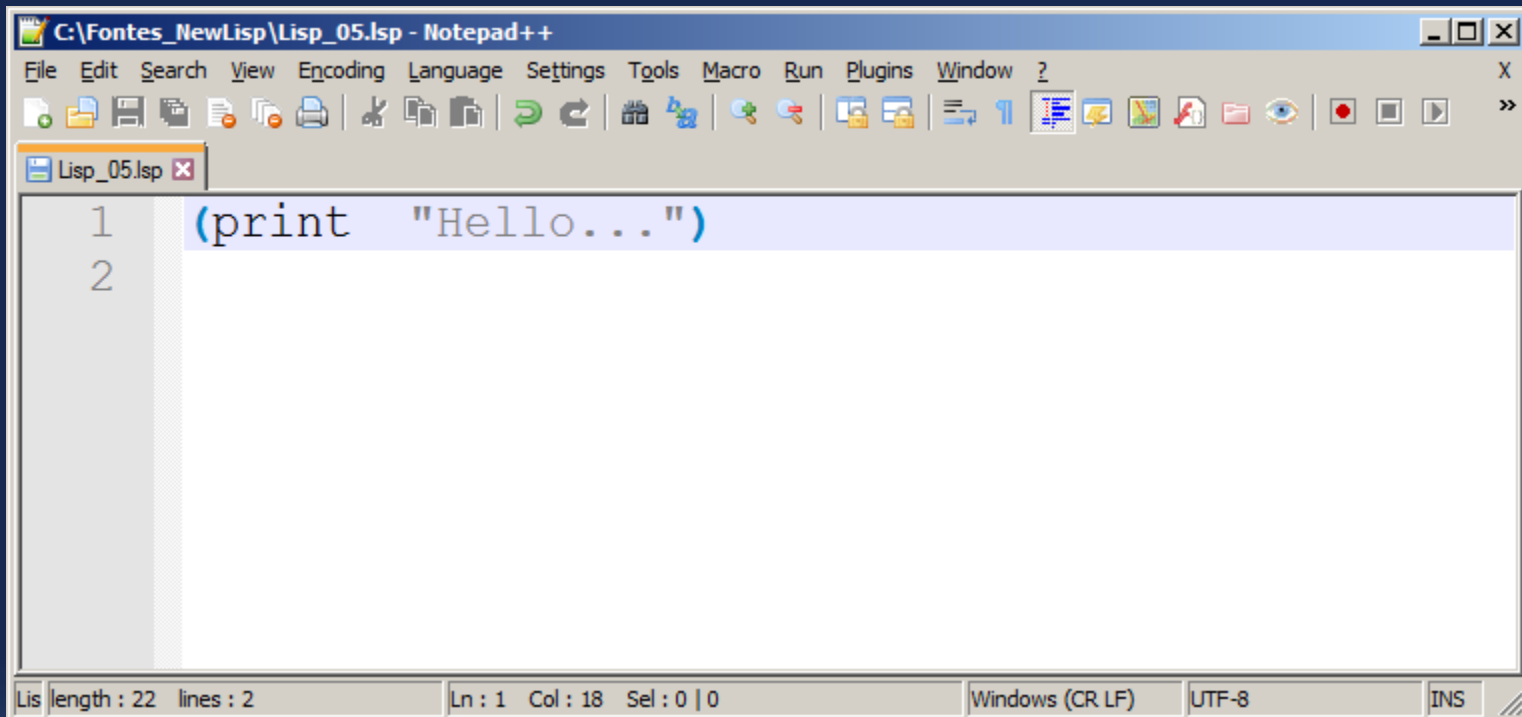
The screenshot shows a Command Prompt window titled 'Command Prompt - newlisp'. The prompt is at 'C:\Fontes_NewLisp>'. The user has entered 'newlisp', and the output is 'newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, opt'. The user then enters '> (load "Lisp_O4.lsp")', and the output lists the contents of the directory: ('\$Recycle.Bin' 'Documents and Settings' 'Fontes_NewLisp' 'cao' 'MSOCache' 'NVIDIA' 'pagefile.sys' 'PerfLogs' 'ProgramData' 'Recovery' 'Softwares' 'System Volume Information' 'Windows').

```
C:\Fontes_NewLisp>newlisp
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, opt

> (load "Lisp_O4.lsp")
("$Recycle.Bin" "Documents and Settings" "Fontes_NewLisp"
"cao"
"MSOCache" "NVIDIA" "pagefile.sys" "PerfLogs" "ProgramData"
"Recovery" "Softwares" "System Volume Information" "Windows")
>
```



Exemplo – Lisp_O5.lsp

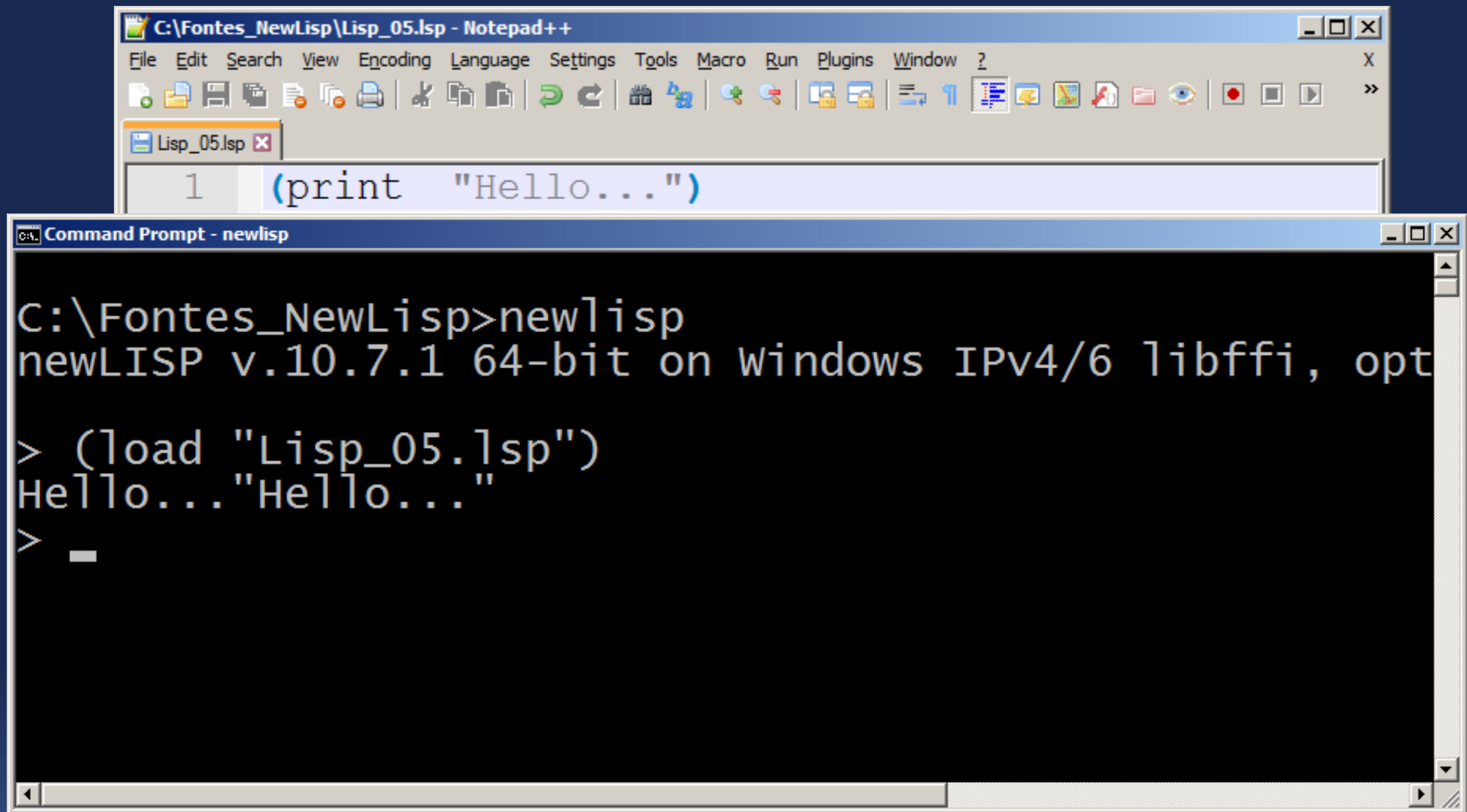


The screenshot shows a Notepad++ window titled "C:\\Fontes_NewLisp\\Lisp_O5.lsp - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, and Window. The toolbar contains various icons for file operations, editing, and development. The text area shows two lines of code: line 1 contains "(print \"Hello...\")" and line 2 is empty. The status bar at the bottom displays "Lis | length : 22 | lines : 2", "Ln : 1 | Col : 18 | Sel : 0 | 0", "Windows (CR LF)", "UTF-8", and "INS".

```
1 (print "Hello...")
2
```



Exemplo – Lisp_O5.lsp



```
C:\Fontes_NewLisp\Lisp_O5.lsp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Lisp_O5.lsp
1 (print "Hello...")

C:\Fontes_NewLisp>newlisp
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, opt
> (load "Lisp_O5.lsp")
Hello..."Hello..."
> _
```

Observação: Além de imprimir o **string**, a função também o retorna.



Exemplo – Leitura de Arquivo

(**read-file** "Arquivo.txt")

Ouviram do Ipiranga as margens plácidas
De um povo heroico o brado retumbante
E o sol da liberdade, em raios fúlgidos
Brilhou no céu da pátria nesse instante

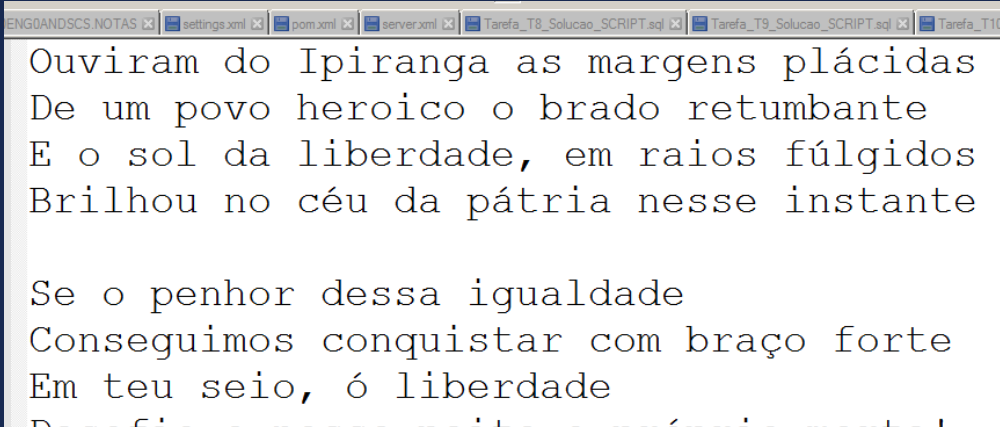
Se o penhor dessa igualdade
Conseguimos conquistar com braço forte

```
Command Prompt - newlisp
E:\NewLisp>newlisp
newLISP v.10.7.5 64-bit on windows IPv4/6 UTF-8 libffi, optio
> (read-file "Arquivo.txt")
"Ouviram do Ipiranga as margens pl|ícidas\r\nDe um povo heroi
te\r\nE o sol da liberdade, em raios f|l|lgidos\r\nBrilhou no
se instante\r\n\r\nse o penhor dessa igualdade\r\nConseguimos
|ºo forte\r\nEm teu seio, | liberdade\r\nDesafia o nosso pei
!\r\n\r\n|ô P|íttria amada\r\nIdolatrada\r\nsalve! salve!\r\n\r
intenso, um raio v|ívido\r\nDe amor e de esperan|ºa |á terra
formoso c|u, risonho e |impido\r\nA imagem do Cruzeiro res
ante pela pr|pria natureza\r\n|és belo, |s forte, imp|ívido
```



Exemplo – Leitura de Arquivo

```
( println ( read-file "Arquivo.txt") )
```



Ouviram do Ipiranga as margens plácidas
De um povo heroico o brado retumbante
E o sol da liberdade, em raios fúlgidos
Brilhou no céu da pátria nesse instante

Se o penhor dessa igualdade
Conseguimos conquistar com braço forte
Em teu seio, ó liberdade

Desafia o nosso peito a própria morte!

Command Prompt - newlisp

>

>

```
(println (read-file "Arquivo.txt"))
```

Ouviram do Ipiranga as margens plácidas
De um povo heroico o brado retumbante
E o sol da liberdade, em raios fúlgidos
Brilhou no céu da pátria nesse instante

Se o penhor dessa igualdade
Conseguimos conquistar com braço forte



Funções newLISP

- ✓ Há cerca de **380** funções **newLISP**;
- ✓ Toda função **newLISP** **retorna** um **valor**;
- ✓ Mesmo um **print** retorna um valor;
- ✓ Diz-se que a ação de imprimir um valor é um "**side-effect**";
- ✓ Assim, quando se executa a função **print** serão vistos 2 valores: um que é o valor impresso e outro o valor retornado.

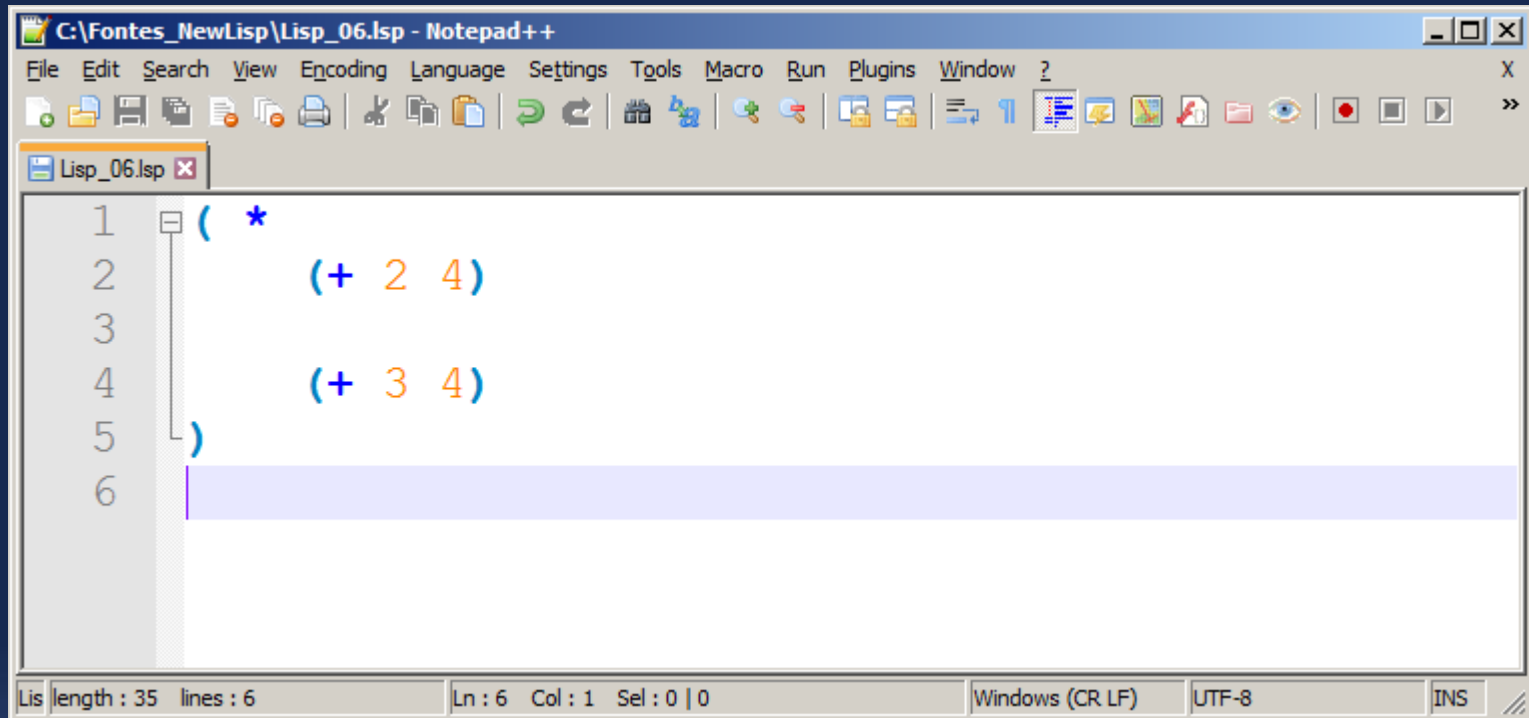


Listas Aninhadas

- ✓ Listas podem conter elementos que também são listas;
- ✓ Exemplo: $(* (+ 2 4) (+ 3 4))$
- ✓ `newLISP` avalia inicialmente a função $(+ 2 4)$ o qual retorna **6**;
- ✓ Em seguida avalia a função $(+ 3 4)$ o qual retorna **7**;
- ✓ Finalmente a expressão $(* 6 7)$ é avaliada, retornando o valor **42**.



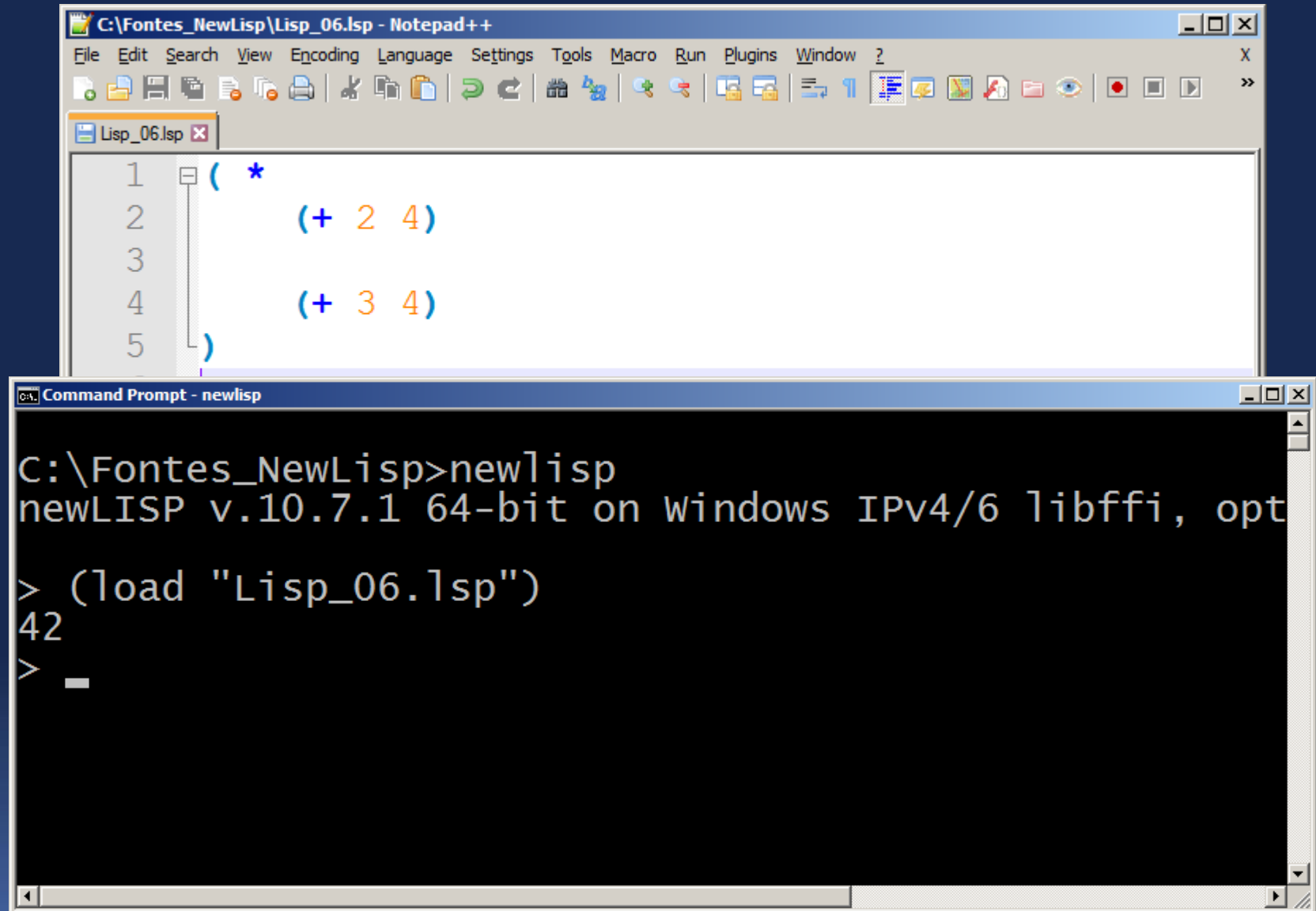
Exemplo – Lisp_O6.lsp



```
C:\Fontes_NewLisp\Lisp_O6.lsp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Lisp_O6.lsp
1 ( *
2   (+ 2 4)
3
4   (+ 3 4)
5 )
6
Lis length : 35 lines : 6 Ln : 6 Col : 1 Sel : 0 | 0 Windows (CR LF) UTF-8 INS
```



Exemplo – Lisp_O6.lsp



The image shows two windows. The top window is Notepad++ editing a file named 'Lisp_O6.lsp' at 'C:\Fontes_NewLisp\Lisp_O6.lsp'. The code in the file is:

```
1 ( *  
2   (+ 2 4)  
3  
4   (+ 3 4)  
5 )
```

The bottom window is a Command Prompt titled 'Command Prompt - newlisp'. It shows the following commands and output:

```
C:\Fontes_NewLisp>newlisp  
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, opt  
> (load "Lisp_O6.lsp")  
42  
> _
```



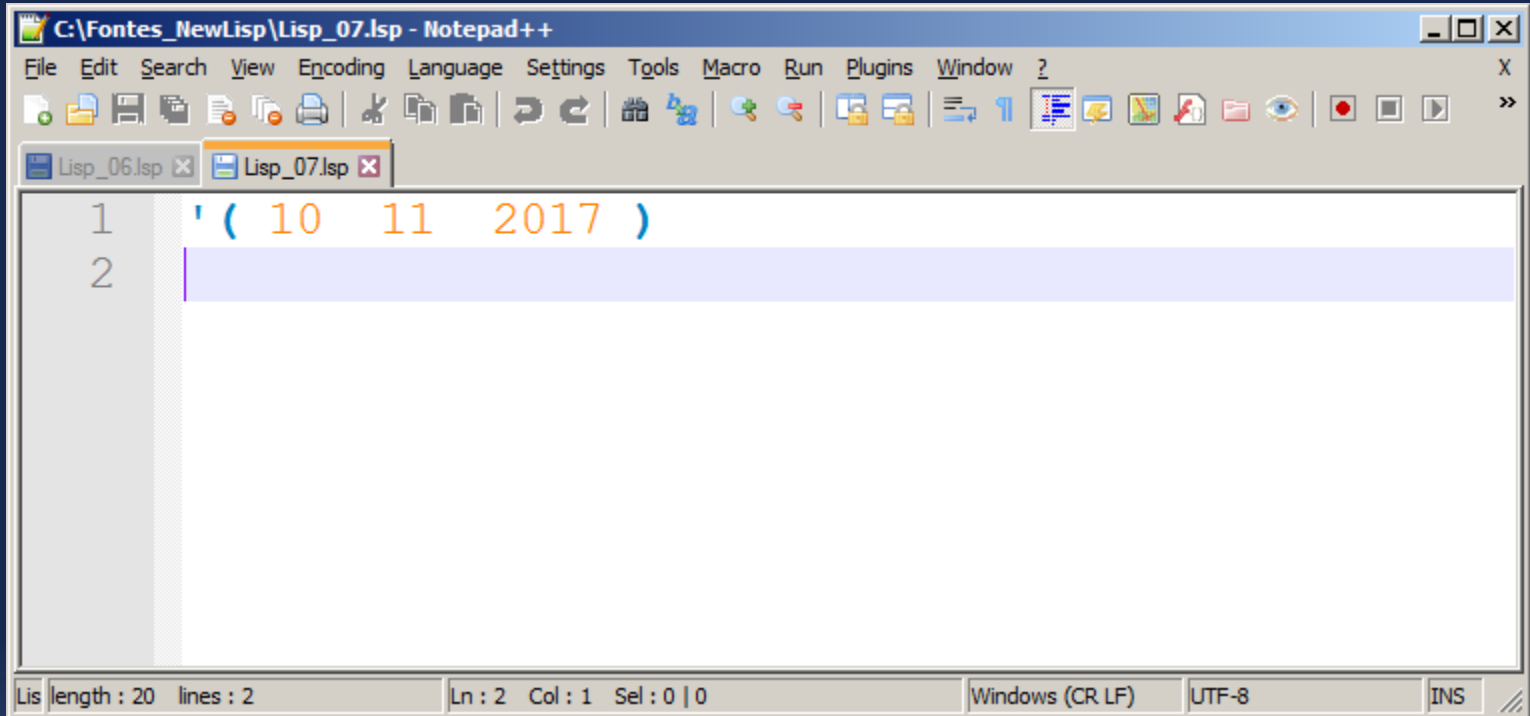
As três regras básicas de newLISP

Regra 3

- ✓ **Apóstrofe** inibe a avaliação da expressão;
- ✓ Assim, **(+ 2 3)** será avaliada pelo interpretador **newLISP** retornando o valor 5;
- ✓ A expressão **'(+ 2 3)** é precedida por **'** o qual previne a avaliação. Nesse caso, a expressão será retornada sem processar a avaliação;
- ✓ Esse procedimento permite que a **lista** seja **tratada** como **dado** e não como chamada de **função** para ser avaliada;
- ✓ Exemplo: **(31 11 2017)** representa uma data e portanto deve ser escrita **'(31 11 2017)**.



Exemplo – Lisp_07.lsp



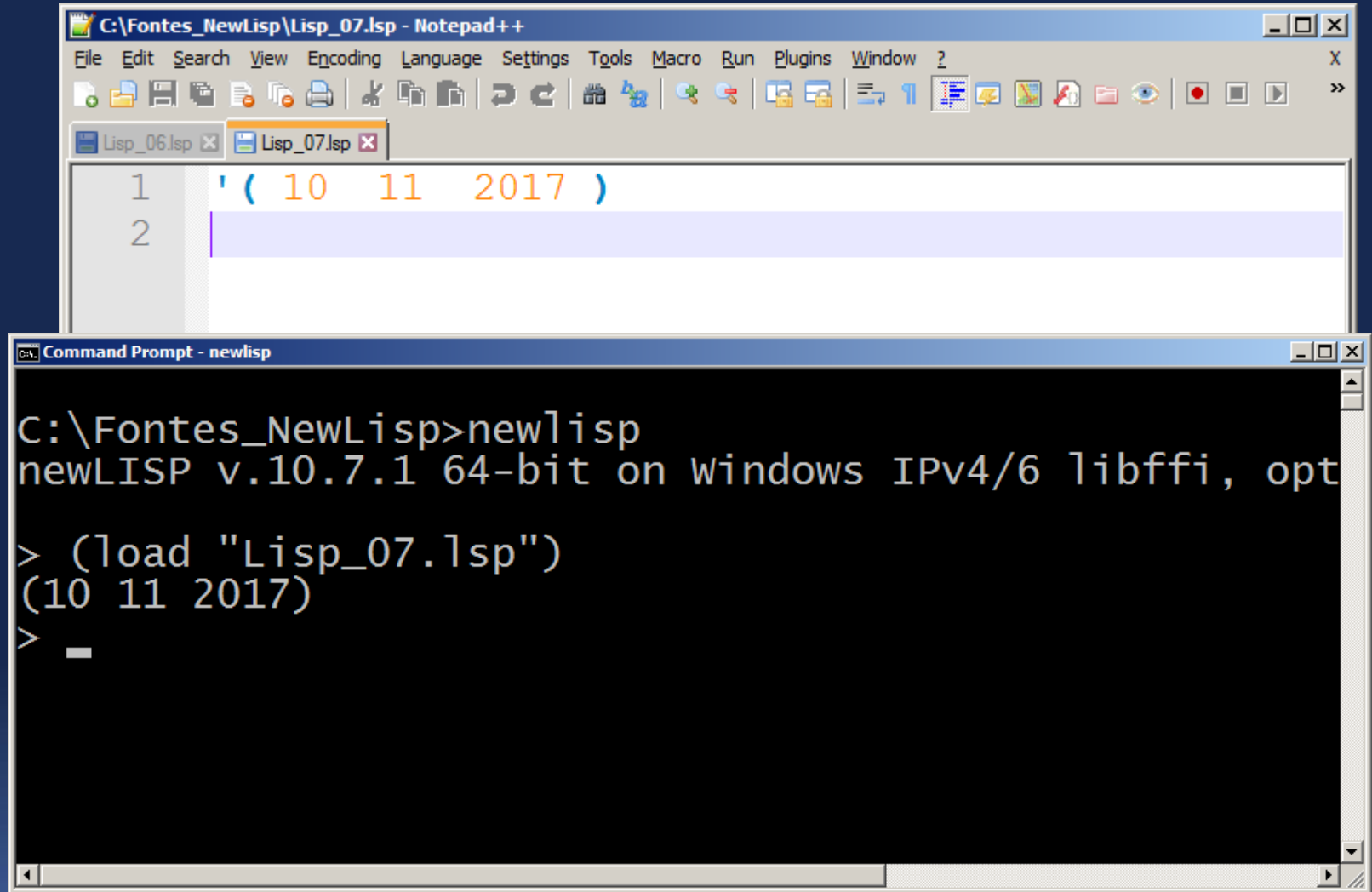
The screenshot shows a Notepad++ window titled "C:\Fontes_NewLisp\Lisp_07.lsp - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, and Window. The toolbar contains various icons for file operations, editing, and development. The tab bar shows two open files: "Lisp_06.lsp" and "Lisp_07.lsp". The main editing area contains two lines of code:

```
1 ' ( 10 11 2017 )  
2
```

The status bar at the bottom displays "Lisp length : 20 lines : 2", "Ln : 2 Col : 1 Sel : 0 | 0", "Windows (CR LF)", "UTF-8", and "INS".



Exemplo – Lisp_07.lsp



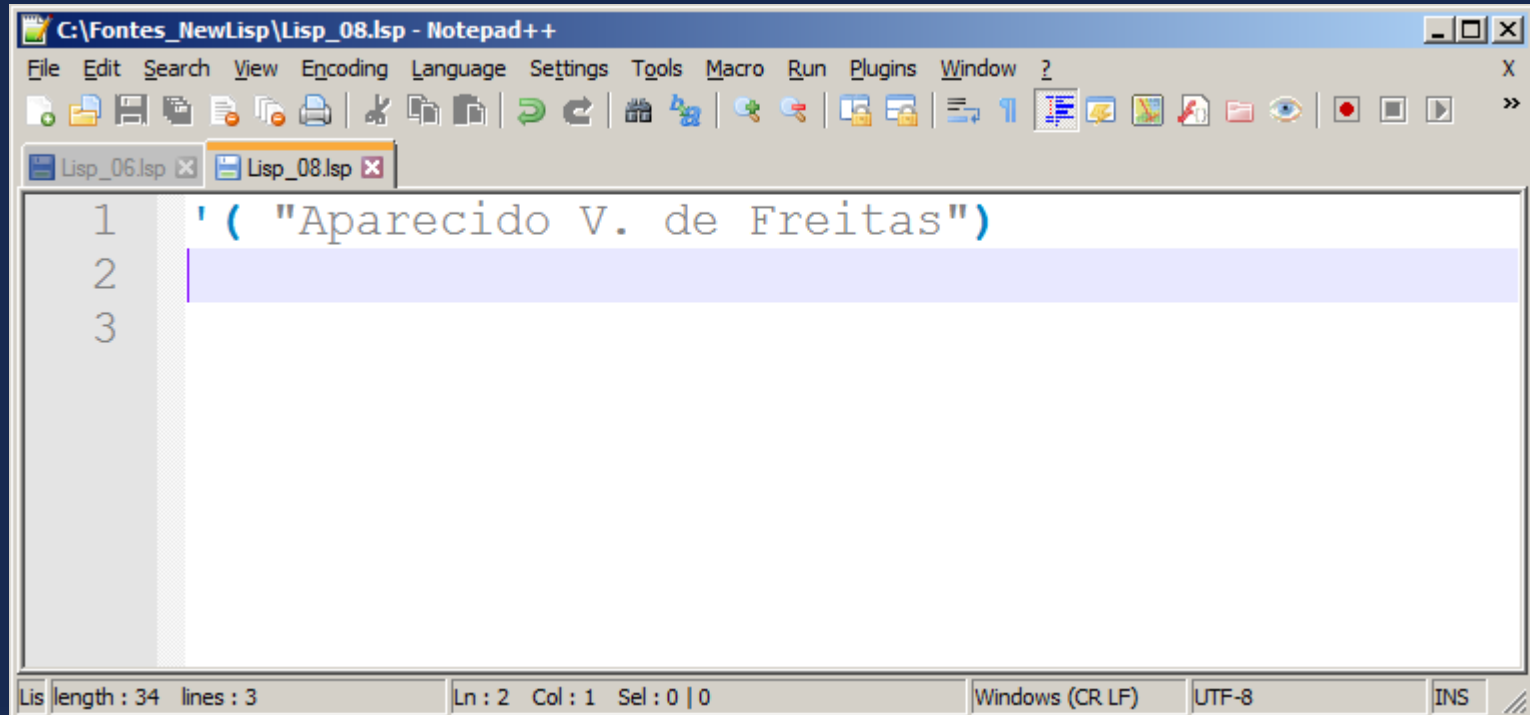
The image shows two windows. The top window is Notepad++ editing 'C:\Fontes_NewLisp\Lisp_07.lsp'. It contains two lines of Lisp code: line 1 is `' (10 11 2017)` and line 2 is empty. The bottom window is a Command Prompt titled 'newlisp'. It shows the command `newlisp` being executed, followed by the version information `newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, opt`. Then, the command `> (load "Lisp_07.lsp")` is entered, and the output `(10 11 2017)` is displayed. A prompt `>` and a cursor are shown on the next line.

```
C:\Fontes_NewLisp\Lisp_07.lsp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Lisp_06.lsp x Lisp_07.lsp x
1 ' ( 10 11 2017 )
2

C:\Fontes_NewLisp>newlisp
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, opt
> (load "Lisp_07.lsp")
(10 11 2017)
>
```



Exemplo – Lisp_08.lsp



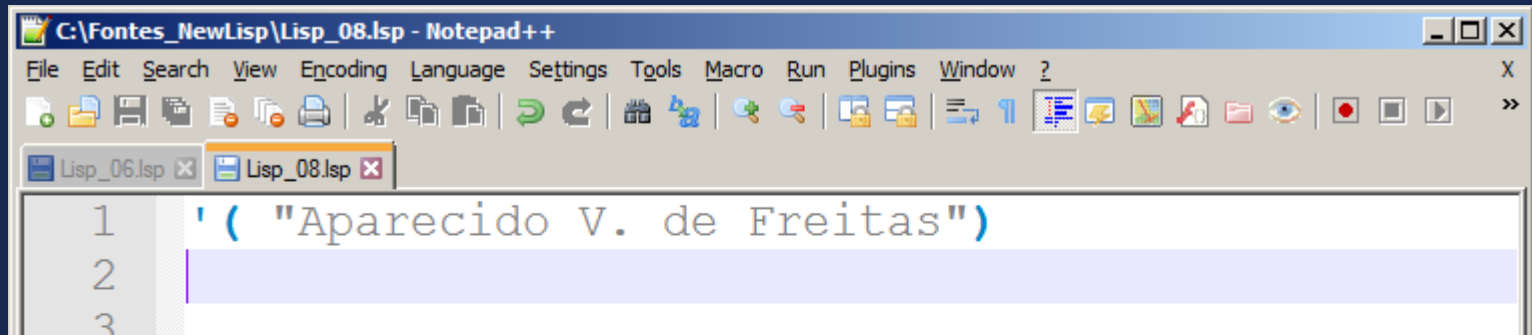
The screenshot shows a Notepad++ window titled "C:\Fontes_NewLisp\Lisp_08.lsp - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, Window, and Help. The toolbar contains various icons for file operations, editing, and development. The tab bar shows two open files: "Lisp_06.lsp" and "Lisp_08.lsp". The main text area contains the following code:

```
1 ' ( "Aparecido V. de Freitas")  
2  
3
```

The status bar at the bottom displays "Lis | length : 34 | lines : 3 | Ln : 2 | Col : 1 | Sel : 0 | 0 | Windows (CR LF) | UTF-8 | INS |".

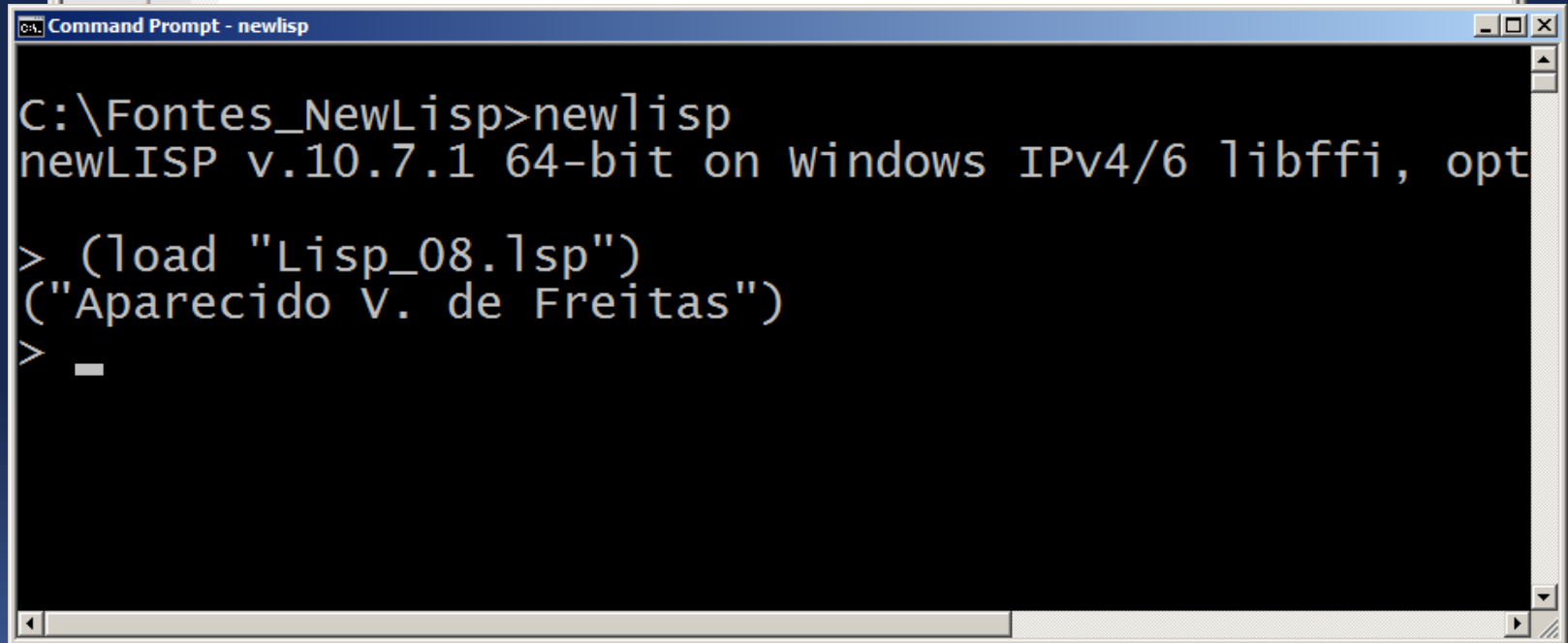


Exemplo – Lisp_08.lsp



A screenshot of the Notepad++ text editor. The title bar reads "C:\Fontes_NewLisp\Lisp_08.lsp - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, Window, and Help. The toolbar contains various icons for file operations and editing. The editor has two tabs open: "Lisp_06.lsp" and "Lisp_08.lsp". The content of "Lisp_08.lsp" is as follows:

```
1  ' ( "Aparecido V. de Freitas")
2
3
```



A screenshot of a Windows Command Prompt window titled "Command Prompt - newlisp". The prompt is at "C:\Fontes_NewLisp>". The user has entered "newlisp", and the system has responded with "newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, opt". The user then enters "> (load \"Lisp_08.lsp\")", and the system outputs the string "Aparecido V. de Freitas". The prompt is now "> _".

```
C:\Fontes_NewLisp>newlisp
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, opt

> (load "Lisp_08.lsp")
"Aparecido V. de Freitas"
> _
```

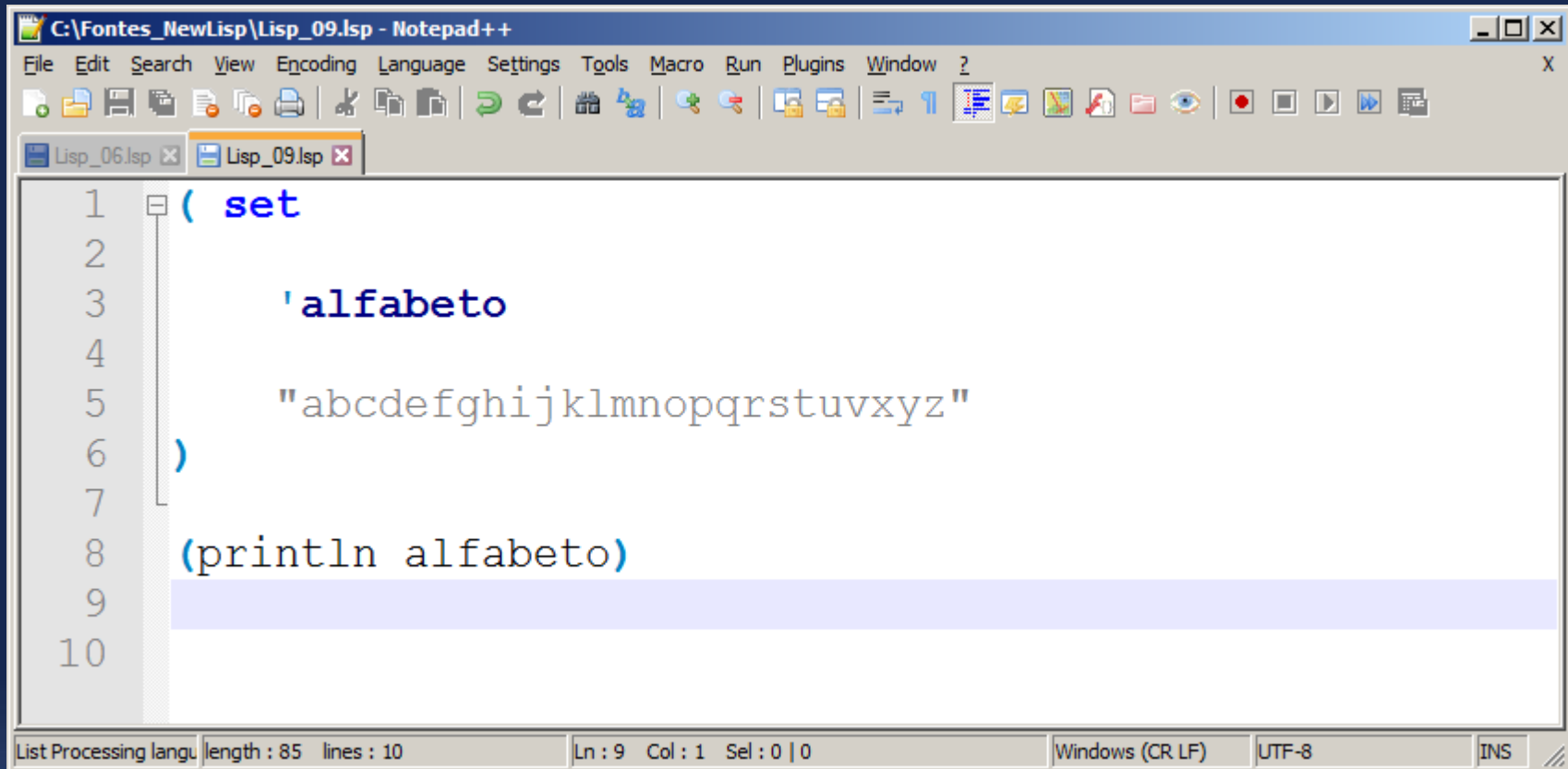


Símbolos

- ✓ Em **newLISP**, um **símbolo** representa algo associado a um **nome**;
- ✓ Pode-se definir esse **nome** representando alguma coisa no código e, posteriormente, usar-se o nome ao invés de seu conteúdo;
- ✓ Por exemplo: (**set** 'alfabeto "abcdefghijklmnopqrstuvxyz")
- ✓ No exemplo, definiu-se por meio da função **set** um nome chamado 'alfabeto cujo valor é o string que contém as 26 letras do alfabeto;
- ✓ A partir dessa definição, o símbolo **alfabeto** pode ser usado em qualquer ponto do programa e será avaliado sempre que for chamado ;



Exemplo – Lisp_09.lsp



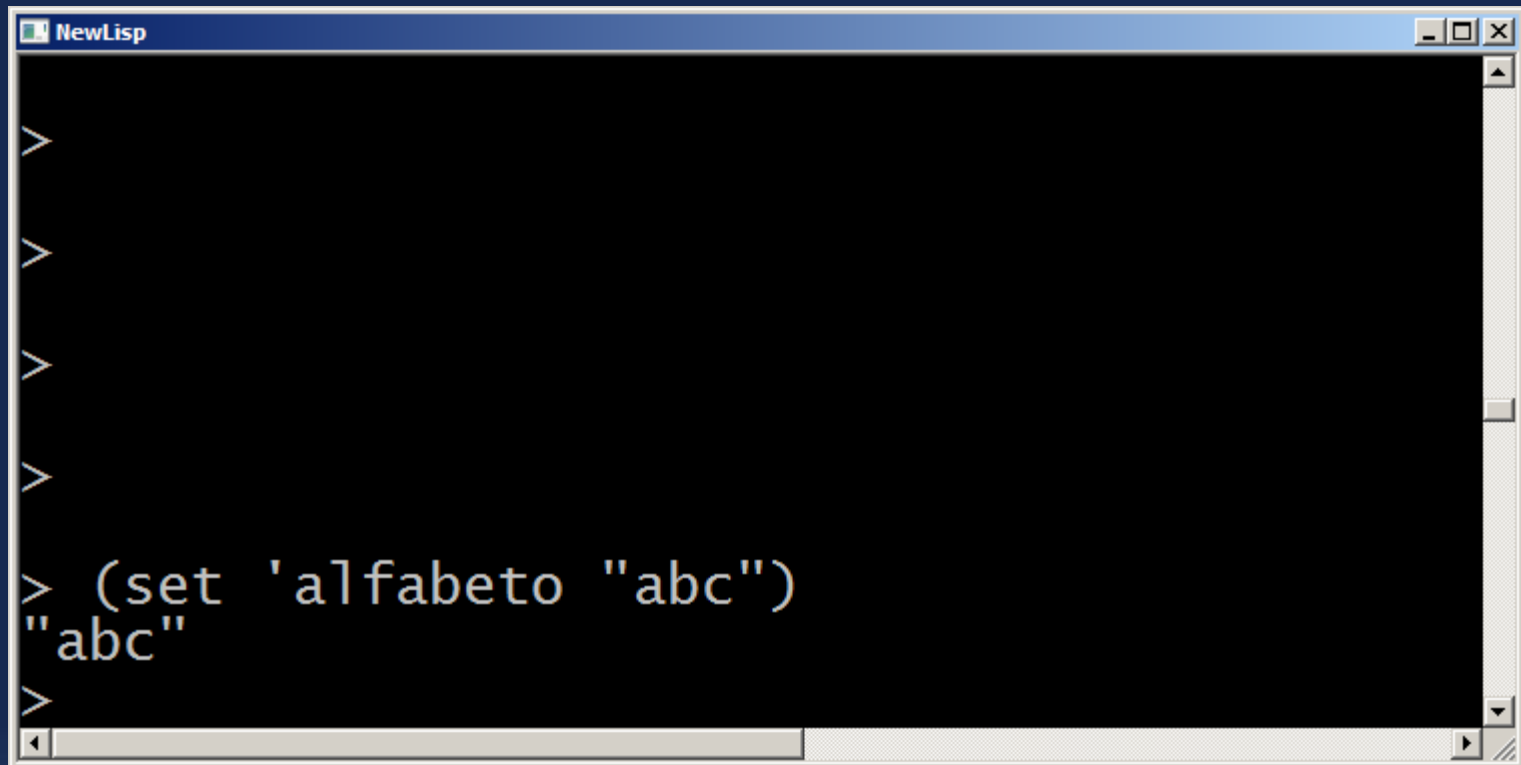
```
1 ( set
2
3     'alfabeto
4
5     "abcdefghijklmnopqrstuvxyz"
6 )
7
8 (println alfabeto)
9
10
```

List Processing langu length : 85 lines : 10 Ln : 9 Col : 1 Sel : 0 | 0 Windows (CR LF) UTF-8 INS





Exemplo – Lisp_10.lsp

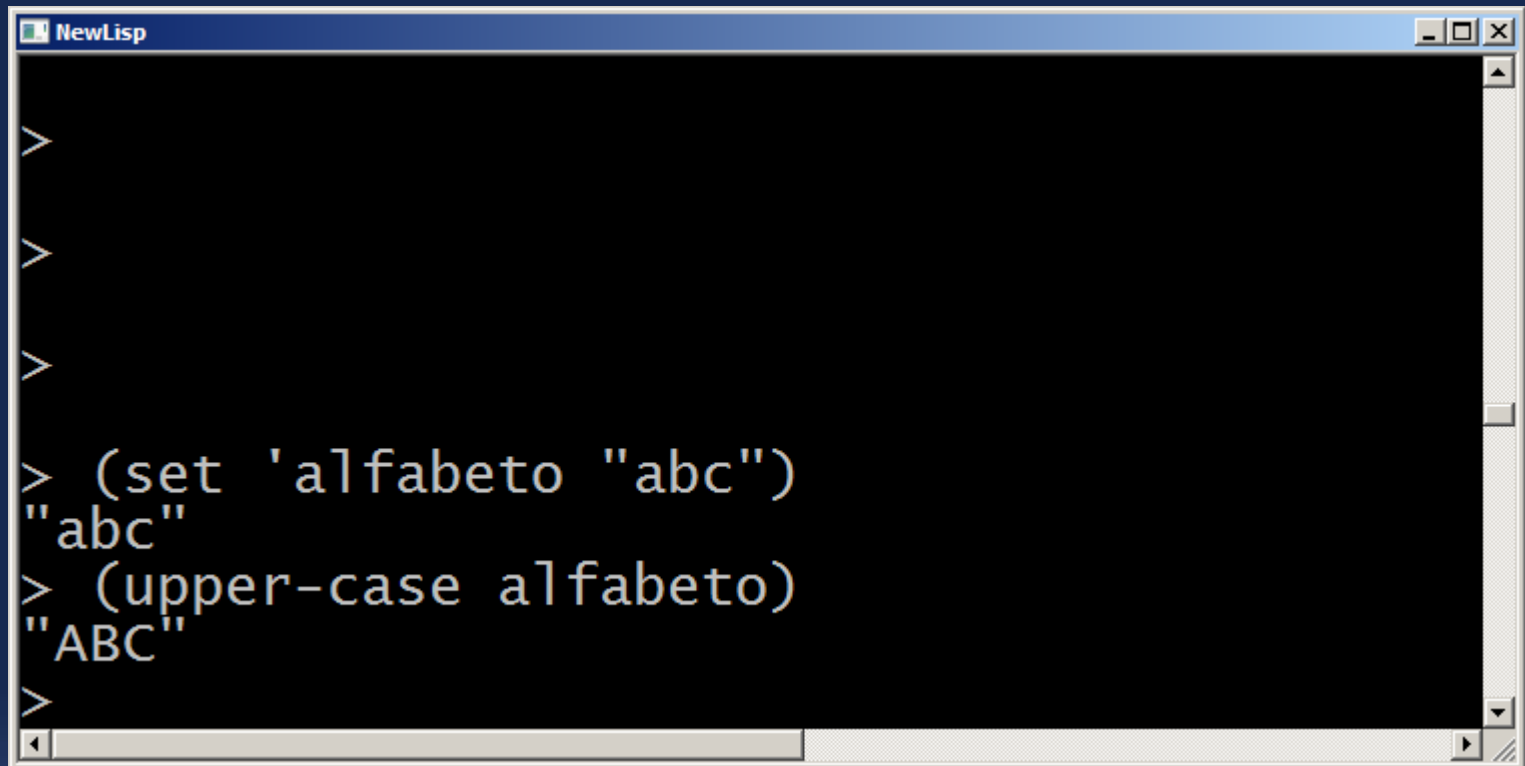


```
>  
>  
>  
>  
> (set 'alfabeto "abc")  
"abc"  
>
```

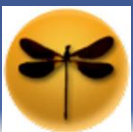




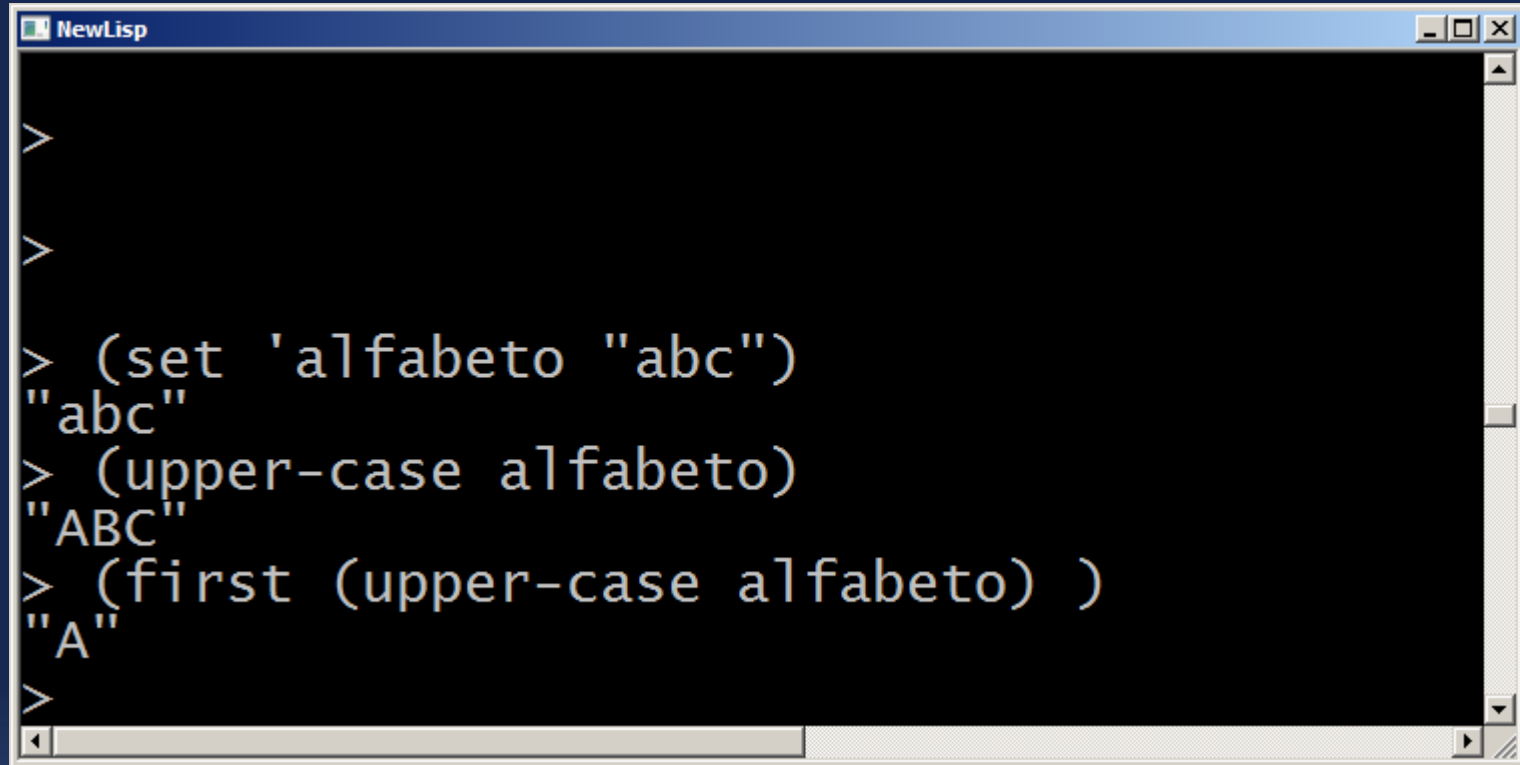
Exemplo – Lisp_10.lsp



```
>  
>  
>  
> (set 'alfabeto "abc")  
"abc"  
> (upper-case alfabeto)  
"ABC"  
>
```



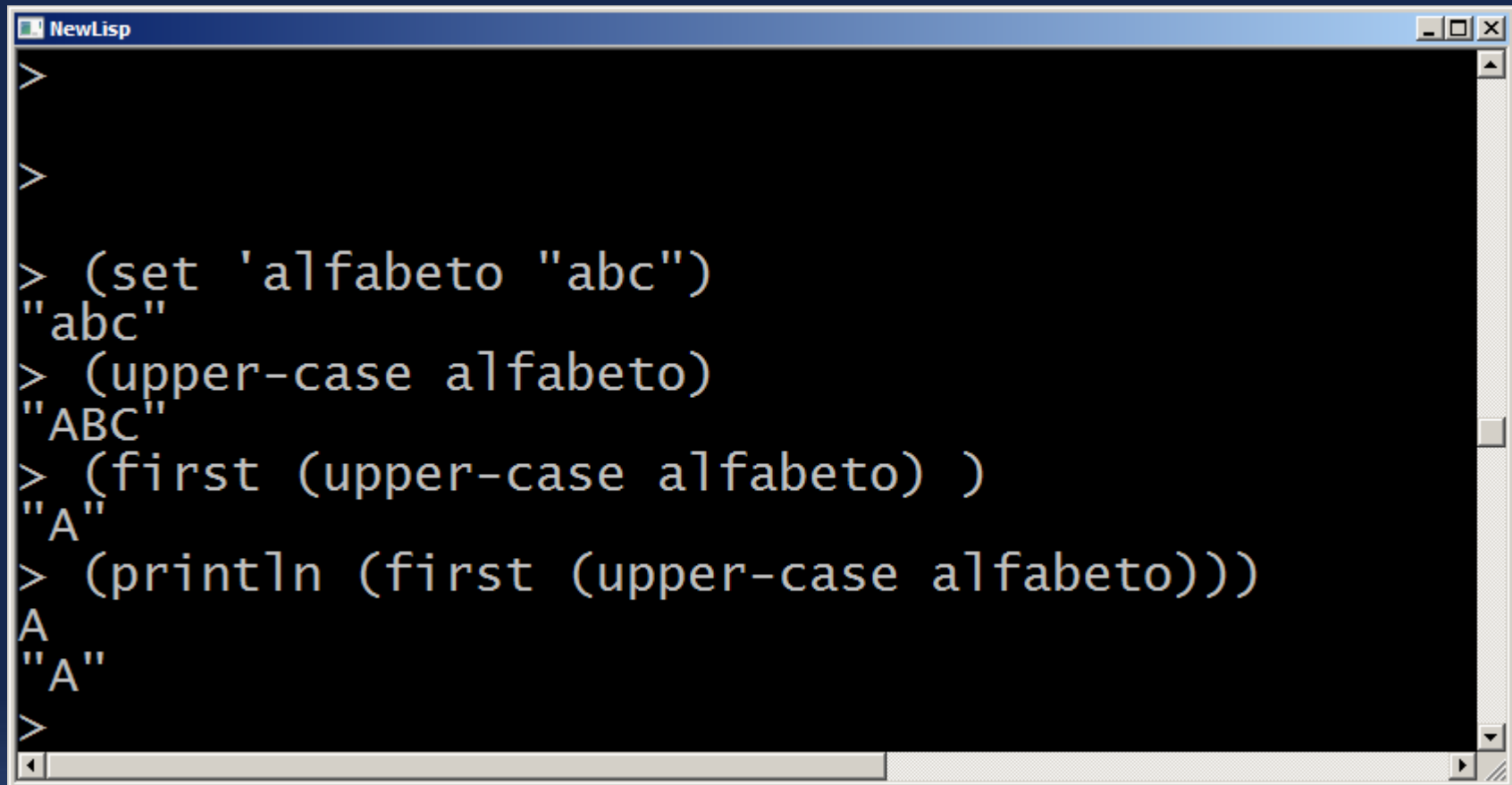
Exemplo – Lisp_10.lsp



```
>  
>  
> (set 'alfabeto "abc")  
"abc"  
> (upper-case alfabeto)  
"ABC"  
> (first (upper-case alfabeto) )  
"A"  
>
```



Exemplo - Lisp_11.lsp



```
>  
>  
> (set 'alfabeto "abc")  
"abc"  
> (upper-case alfabeto)  
"ABC"  
> (first (upper-case alfabeto) )  
"A"  
> (println (first (upper-case alfabeto)))  
A  
"A"  
>
```





O valor do símbolo 'alfabeto **não** foi permanentemente alterado, ou seja permaneceu com os valores em minúsculo, pois a função upper-case criou e retornou um novo string com os valores em maiúsculo. (O símbolo 'alfabeto permaneceu inalterado)

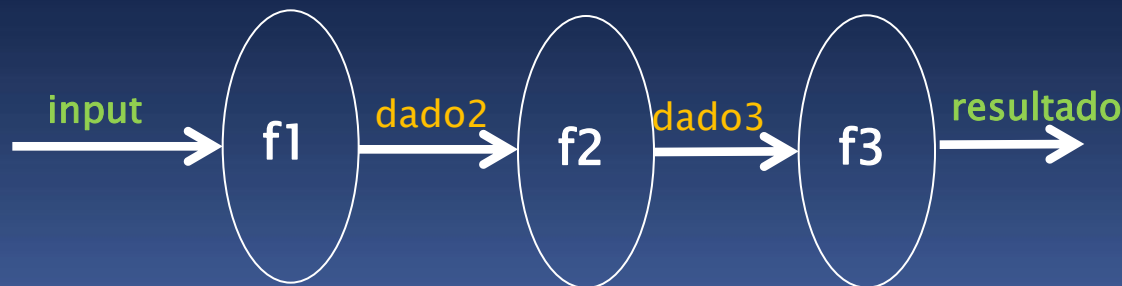
```
NewLisp
```

```
>  
>  
>  
>  
>  
>  
>  
(set 'alfabeto "abc")  
"abc"  
> (upper-case alfabeto)  
"ABC"  
> (first (upper-case alfabeto) )  
"A"  
> alfabeto  
"abc"  
>
```



Símbolos – Mais observações

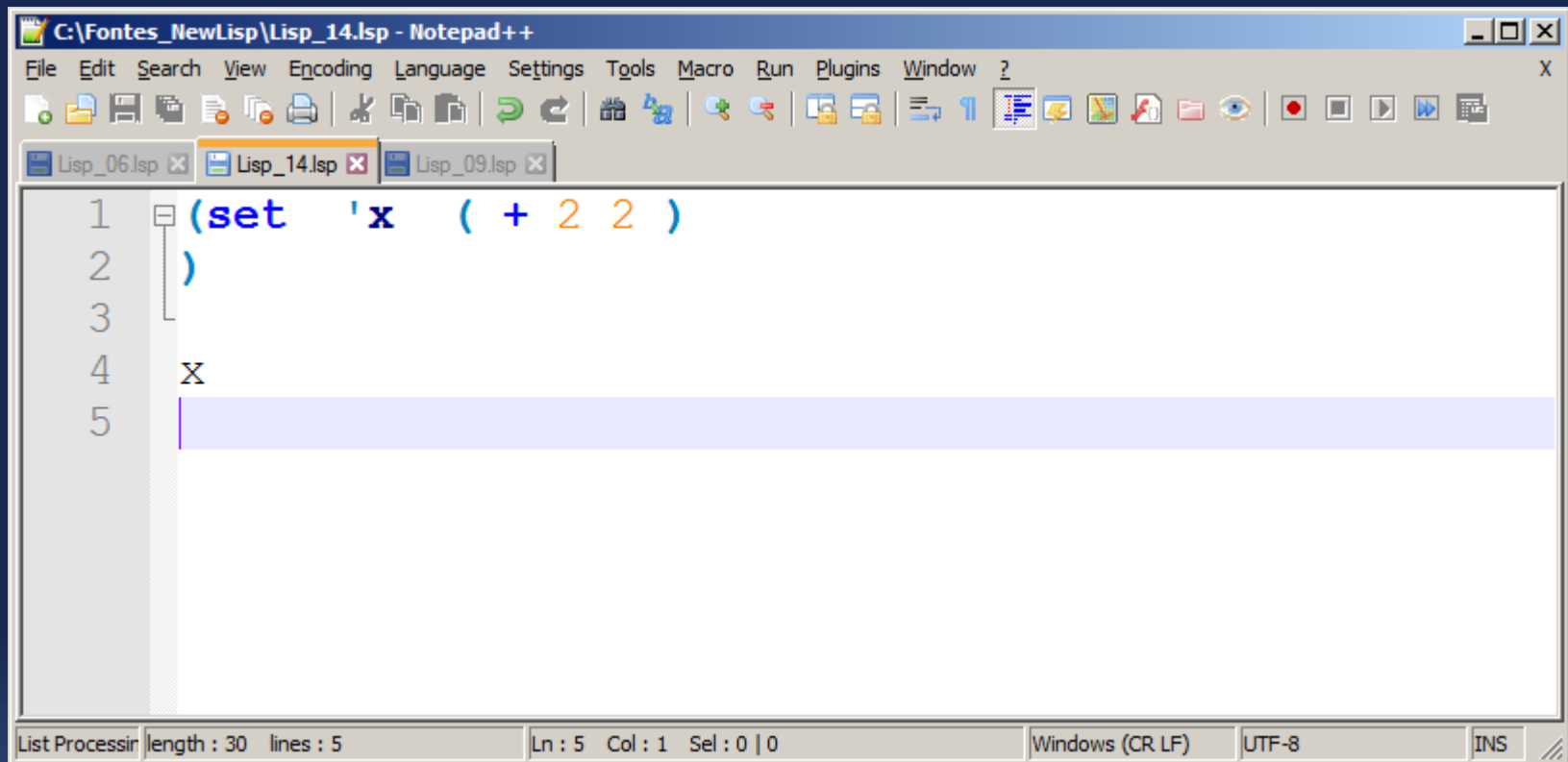
- ✓ **Símbolos** correspondem à “variáveis” em outras linguagens de programação;
- ✓ Na verdade, **newLISP** **não** usa símbolos de forma extensiva como se usa em outras linguagens de programação;
- ✓ A razão disto é que valores retornados por uma função estarão sendo continuamente entrados para outras funções, sem portanto a necessidade de armazená-los;
- ✓ No paradigma funcional o foco está na **transformação dos dados** de forma sucessiva até atingir-se o resultado final esperado.





Exemplo – Lisp_14.lsp

Exemplo



The screenshot shows a Notepad++ window titled "C:\Fontes_NewLisp\Lisp_14.lsp - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, and Window. The toolbar contains various icons for file operations, editing, and running. The tab bar shows three open files: Lisp_06.lsp, Lisp_14.lsp (active), and Lisp_09.lsp. The editor content is as follows:

```
1 (set 'x ( + 2 2 )  
2 )  
3  
4 x  
5
```

The status bar at the bottom displays "List Processir", "length : 30", "lines : 5", "Ln : 5", "Col : 1", "Sel : 0 | 0", "Windows (CR LF)", "UTF-8", and "INS".



Exemplo – Lisp_14.lsp

C:\Fontes_NewLisp\Lisp_14.lsp - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

Lisp_06.lsp x Lisp_14.lsp x Lisp_09.lsp x

```
1 (set 'x (+ 2 2))
2 )
3
4 x
5
```

NewLisp

```
>
>
>
>
> (set 'x (+ 2 2))
4
> x
4
>
```

List P

x tem um binding ao valor 4

Exemplo – Lisp 15.lsp

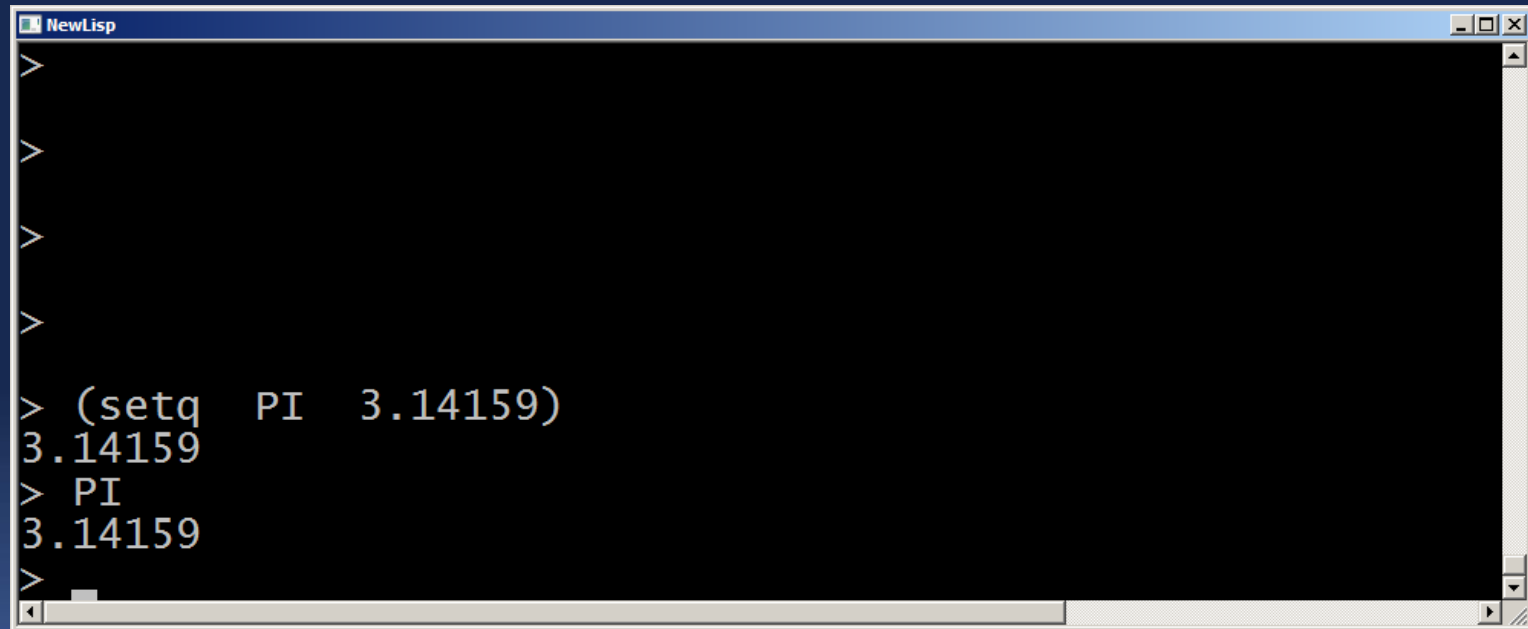
```
C:\Fontes_NewLisp\Lisp_15.lsp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Lisp_06.lsp x Lisp_15.lsp x Lisp_09.lsp x
1 (set 'y '( + 2 2 ))
2 )
3
4 y
5

NewLisp
>
>
>
>
> (set 'y '( + 2 2 ))
> (+ 2 2)
> y
> (+ 2 2)
>
```

y tem um binding à lista (+ 2 2)

Definindo símbolos

- ✓ **Símbolos** também podem ser definidos pela função **setq**;
- ✓ A função **setq** considera que primeiro argumento é um **símbolo**, e dessa forma, não se exige que o argumento seja **quoted** (com apóstrofe).

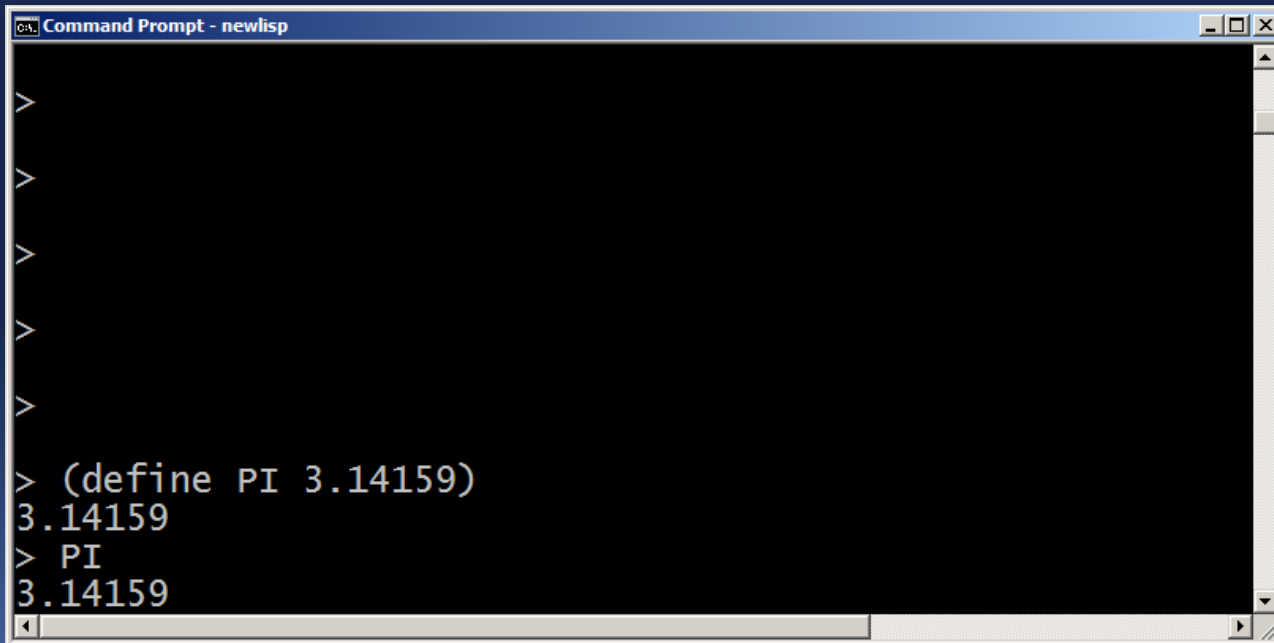


```
NewLisp
>
>
>
>
> (setq PI 3.14159)
3.14159
> PI
3.14159
>
```



Definindo símbolos

- ✓ **Símbolos** também podem ser definidos pela função **define**;
- ✓ A função **set** vista anteriormente exige que o símbolo seja definido com **'**;
- ✓ Mas, **define** não exige que o argumento seja **quoted**;
- ✓ Define também é usado para definição de funções.



```
Command Prompt - newlisp
>
>
>
>
>
> (define PI 3.14159)
3.14159
> PI
3.14159
```



Definindo símbolos com define

```
Command Prompt - newlisp

>
> (define trab (+ 4 3) )
7
> trab
7
> _
```

```
Command Prompt - newlisp

>
>
> (define trab '(+ 4 3) )
(+ 4 3)
> trab
(+ 4 3)
>
>
>
>
```



Controlando o Fluxo de Execução

- ✓ Há diferentes formas de se controlar o Fluxo de Execução de nosso código;
- ✓ Todo o controle de fluxo em **newLISP** seguem determinadas regras da linguagem;
- ✓ A forma geral de definir o controle de fluxo é por meio de uma **lista** no qual o primeiro elemento é uma **keyword** (palavra reservada) seguida por uma ou mais expressões para serem avaliadas.

```
( keyword      expressao2      expressao3 ... )
```



if

- ✓ Consiste de um teste e uma ação;
- ✓ A grande maioria das coisas já conhecidas em **newLISP** são **true**;
- ✓ Há duas importantes coisas em **newLISP** que são **falsas**: **nil** e a lista vazia **()**.
- ✓ Qualquer coisa que **newLISP** não conhece também é retornado como **falso**.

```
Command Prompt - newlisp
>
> (define PI 3.14)
3.14
> (if PI)
3.14
> (if (= PI 3.14) "Verdade" "Falso")
"Verdade"
> (define A 3.14)
3.14
> (if A)
3.14
> (if (= A PI) "Verdade" "Falso")
"Verdade"
>
```

if

- ✓ O símbolo **x** é conhecido, portanto (if x) retorna **999** que é **true**;
- ✓ O símbolo **y** é desconhecido, portanto (if y) retorna **nil**, que corresponde a **false**.

```
Command Prompt - newlisp
C:\Users\VB_Aparecido>newlisp
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, options:

> (set 'x 999)
999
> x
999
> (if x)
999
> (if a)
nil
> _
```



if

- ✓ O símbolo **var1** é conhecido, portanto **(if var1 "ola" "mundo")** retorna **"ola"** que é **true**;
- ✓ Mas, o símbolo **var1** após o **if**, continua com o valor original, **"teste"**
- ✓ A terceira expressão, representa uma ação **else**.

```
Command Prompt - newlisp
C:\Users\VB_Aparecido>(newlisp)
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, options:

> (define var1 "teste")
"teste"
> var1
"teste"
> (if var1 "ola" "mundo")
"ola"
> var1
"teste"
> _
```



if

- ✓ O símbolo **var3** não é conhecido, portanto (if **var3** "ola" "mundo") retorna "**mundo**";
- ✓ O símbolo **var2** após o if, continua com o valor original, "**uscs**";
- ✓ O símbolo **var3** não está definido e, portanto, ao acessá-lo será retornado **nil**.

```
Command Prompt - newlisp
C:\Users\VB_Aparecido>newlisp
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, options: n
> (define var2 "uscs")
"uscs"
> var2
"uscs"
> (if var3 "ola" "mundo")
"mundo"
> var3
nil
> var2
"uscs"
>
```



Testes com if

```
Command Prompt - newlisp
C:\Users\VB_Aparecido>(newlisp)
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, options: n
> (if (> 4 3) "hello" "uscs")
"hello"
>
> (if (> 2 9) "hello" "uscs")
"uscs"
> _
```



Testes com if

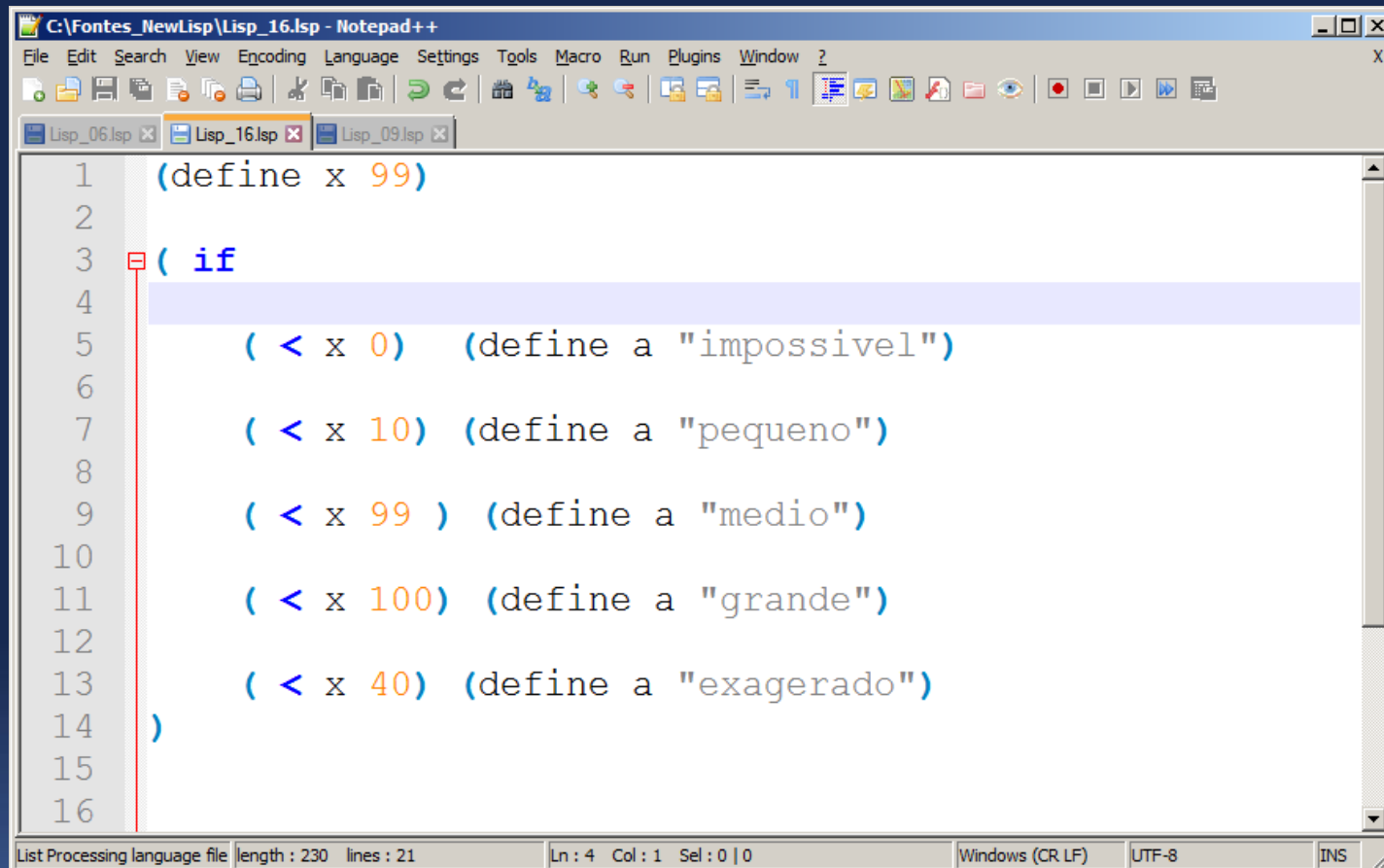
- ✓ Se um símbolo é avaliado como **nil** (talvez porque ele **não** exista ou não tenha sido assinalado a algum valor) **newLISP** o considera **falso** e o teste retorna **nil**.

```
Command Prompt - newlisp
C:\Users\VB_Aparecido>newlisp
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, options:
> (define x "uscs")
"uscs"
> (if x "sim" "nao")
"sim"
> (if y "sim" "nao")
"nao"
> (if y)
nil
>
```



If com diversos testes

- ✓ Pode-se empregar **if** com um número ilimitado de testes e ações;
- ✓ Neste caso, a lista de **if**'s consiste de uma série de pares teste-ação;



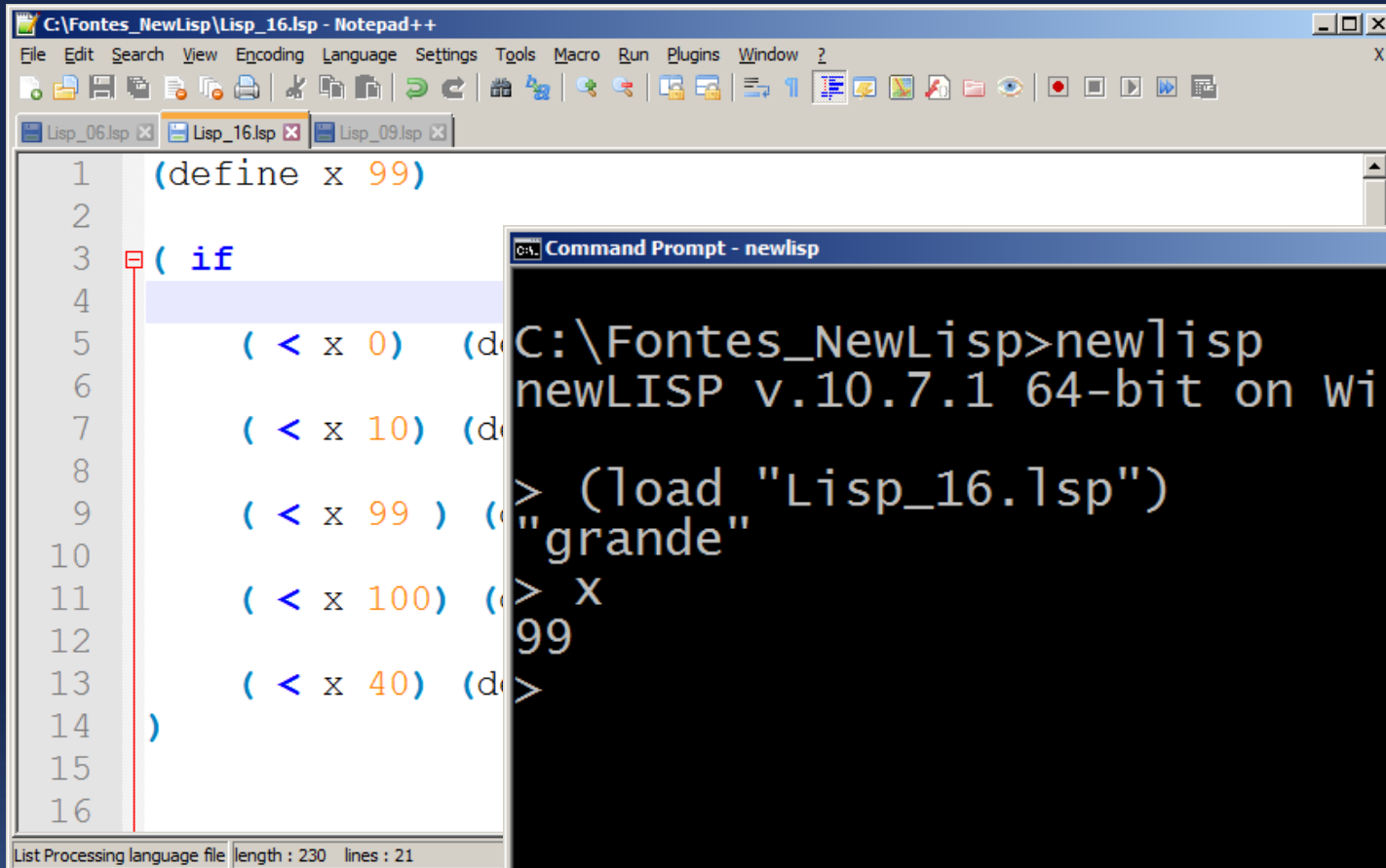
```
1 (define x 99)
2
3 (if
4
5     (< x 0) (define a "impossivel")
6
7     (< x 10) (define a "pequeno")
8
9     (< x 99) (define a "medio")
10
11    (< x 100) (define a "grande")
12
13    (< x 40) (define a "exagerado")
14 )
15
16
```

List Processing language file | length : 230 | lines : 21 | Ln : 4 | Col : 1 | Sel : 0 | 0 | Windows (CR LF) | UTF-8 | INS



If com diversos testes

- ✓ Pode-se empregar **if** com um número ilimitado de testes e ações;
- ✓ Neste caso, a lista de **if's** consiste de uma série de pares teste-ação;



The image shows a Notepad++ window editing a file named `Lisp_16.lsp` at the path `C:\Fontes_NewLisp\Lisp_16.lsp`. The code defines a variable `x` to 99 and then uses an `if` statement with multiple conditions and actions. The Command Prompt window shows the execution of `newlisp` and `(load "Lisp_16.lsp")`, resulting in the value of `x` being printed as 99.

```
(define x 99)

(if
  (< x 0) (do)
  (< x 10) (do)
  (< x 99) (do)
  (< x 100) (do)
  (< x 40) (do)
)
```

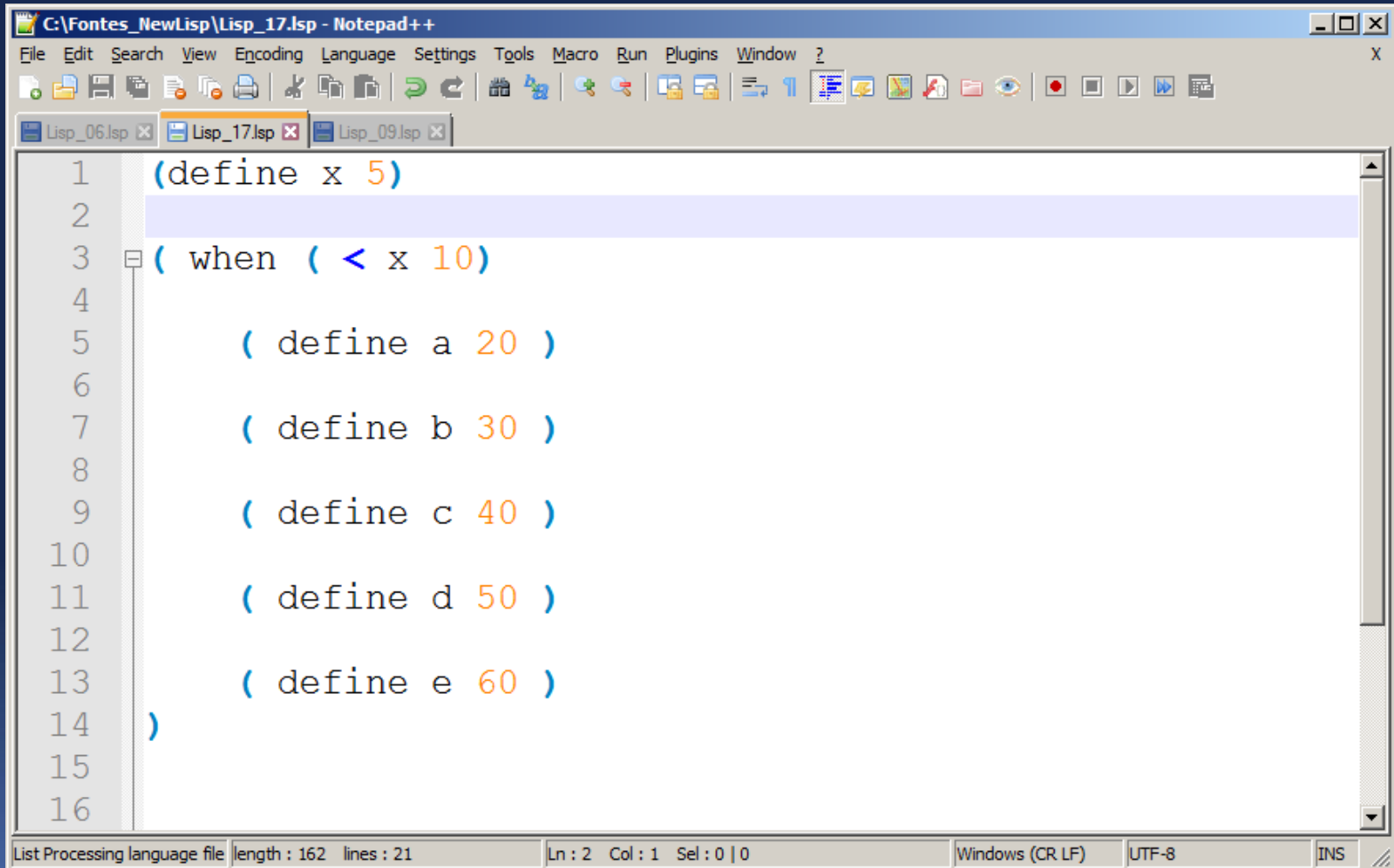
```
C:\Fontes_NewLisp>newlisp
newLISP v.10.7.1 64-bit on Windows IPv4/6

> (load "Lisp_16.lsp")
"grande"
> x
99
>
```



Função when

- ✓ A função **when** permite que duas ou mais ações sejam processadas se um teste for satisfeito;
- ✓ Pode-se usar **when** como um **if** sem a parte 'else'.



```
C:\Fontes_NewLisp\Lisp_17.lsp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Lisp_06.lsp x Lisp_17.lsp x Lisp_09.lsp x
1 (define x 5)
2
3 ( when ( < x 10)
4
5     ( define a 20 )
6
7     ( define b 30 )
8
9     ( define c 40 )
10
11    ( define d 50 )
12
13    ( define e 60 )
14 )
15
16
List Processing language file length : 162 lines : 21 Ln : 2 Col : 1 Sel : 0 | 0 Windows (CR LF) UTF-8 INS
```



Função when

The screenshot shows a Notepad++ window titled "C:\Fontes_NewLisp\Lisp_17.lsp - Notepad++" with the following code:

```
1 (define x 5)
2
3 (when (< x 10)
4
5   (define a 20)
6
7   (define
8
9   (define
10
11   (define
12
13   (define
14 )
15
16
```

Below the Notepad++ window is a Command Prompt window titled "Command Prompt - newlisp" showing the execution of the LISP code:

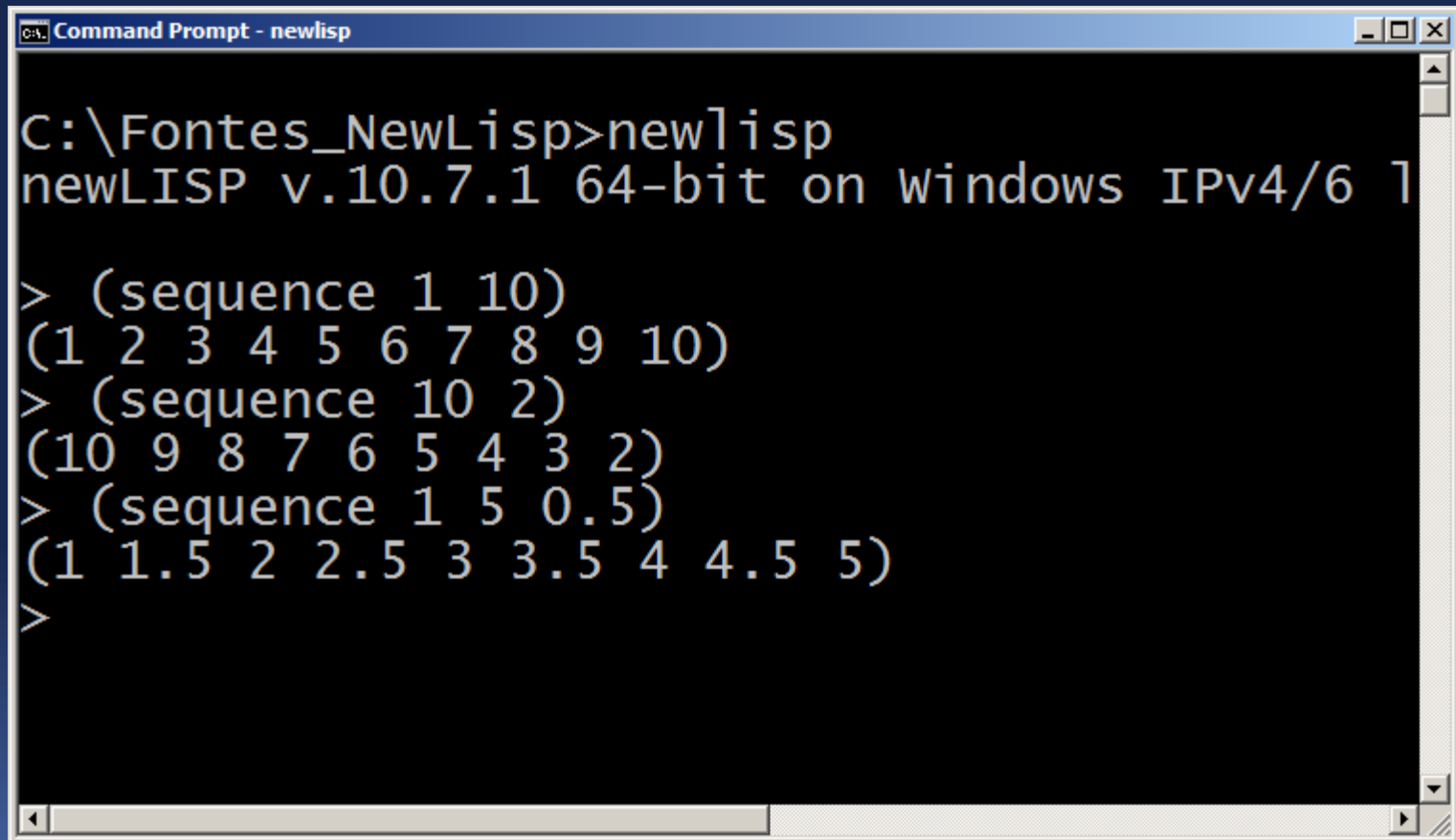
```
C:\Fontes_NewLisp>newlisp
newLISP v.10.7.1 64-bit on Windows IPv4/6 libffi, optio
> (load "Lisp_17.lsp")
60
> a
20
> b
30
> c
40
> d
50
> e
60
>
```

The status bar at the bottom of the Notepad++ window indicates: "List Processing language file | length : 162 | lines : 21".



Trabalhando com Listas

- ✓ A função **sequence** gera uma **sequência** de números em um intervalo, com um valor opcional definindo o passo da sequência.
- ✓ Quando o passo da sequência for omitido, o valor **1** é assumido.



```
Command Prompt - newlisp
C:\Fontes_NewLisp>newlisp
newLISP v.10.7.1 64-bit on Windows IPv4/6 1

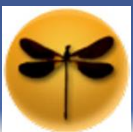
> (sequence 1 10)
(1 2 3 4 5 6 7 8 9 10)
> (sequence 10 2)
(10 9 8 7 6 5 4 3 2)
> (sequence 1 5 0.5)
(1 1.5 2 2.5 3 3.5 4 4.5 5)
>
```



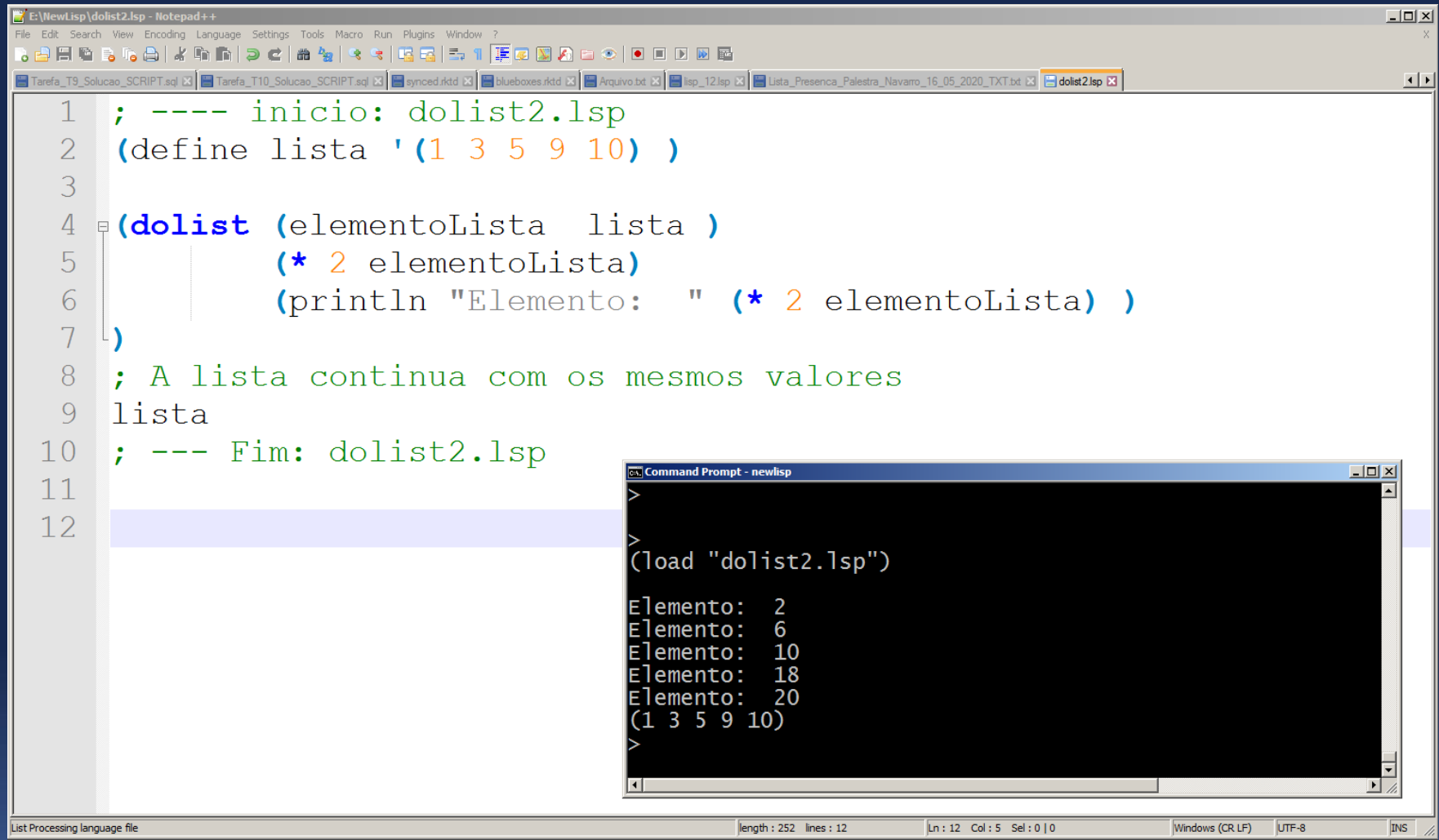


Função dolist

- ✓ A função **dolist** seta um símbolo local (“variável”) para cada item da lista e permite que se executem ações em cada elemento da lista.

[illegible]

Função dolist



The image shows a Notepad++ window with the file 'E:\NewLisp\dolist2.lsp' open. The code defines a function 'dolist' that iterates over a list, doubling each element and printing the result. The list starts with (1 3 5 9 10). The output of the function is shown in a Command Prompt window, which displays the doubled elements (2 6 10 18 20) and the original list (1 3 5 9 10).

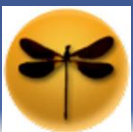
```
1 ; ---- inicio: dolist2.lsp
2 (define lista '(1 3 5 9 10) )
3
4 (dolist (elementoLista lista )
5         (* 2 elementoLista)
6         (println "Elemento: " (* 2 elementoLista) )
7     )
8 ; A lista continua com os mesmos valores
9 lista
10 ; --- Fim: dolist2.lsp
11
12
```

Command Prompt - newlisp

```
>
>
(load "dolist2.lsp")

Elemento: 2
Elemento: 6
Elemento: 10
Elemento: 18
Elemento: 20
(1 3 5 9 10)
>
```

List Processing language file | length : 252 | lines : 12 | Ln : 12 Col : 5 Sel : 0 | 0 | Windows (CR LF) | UTF-8 | INS





Função dolist

```
E:\NewLisp\dolist3.lsp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Tarefa_T9_Solucao_SCRIPT.sqi Tarefa_T10_Solucao_SCRIPT.sqi synced rkttd blueboxes rkttd Arquivo.txt lsp_12.lsp Lista_Presenca_Palestra_Navaro_16_05_2020_TXT.txt dolist3.lsp

1 ; ---- inicio: dolist3.lsp
2 (define lista '(1 3 5 9 10) )
3 (define contador 0)
4
5 (dolist (elementoLista lista)
6
7     (println "Elemento " contador ": " elementoLista)
8     (define contador (+ contador 1) )
9 )
10 ; --- Fim: dolist2.lsp
11
12
```

```
Command Prompt - newlisp
>
>
(load "dolist3.lsp")

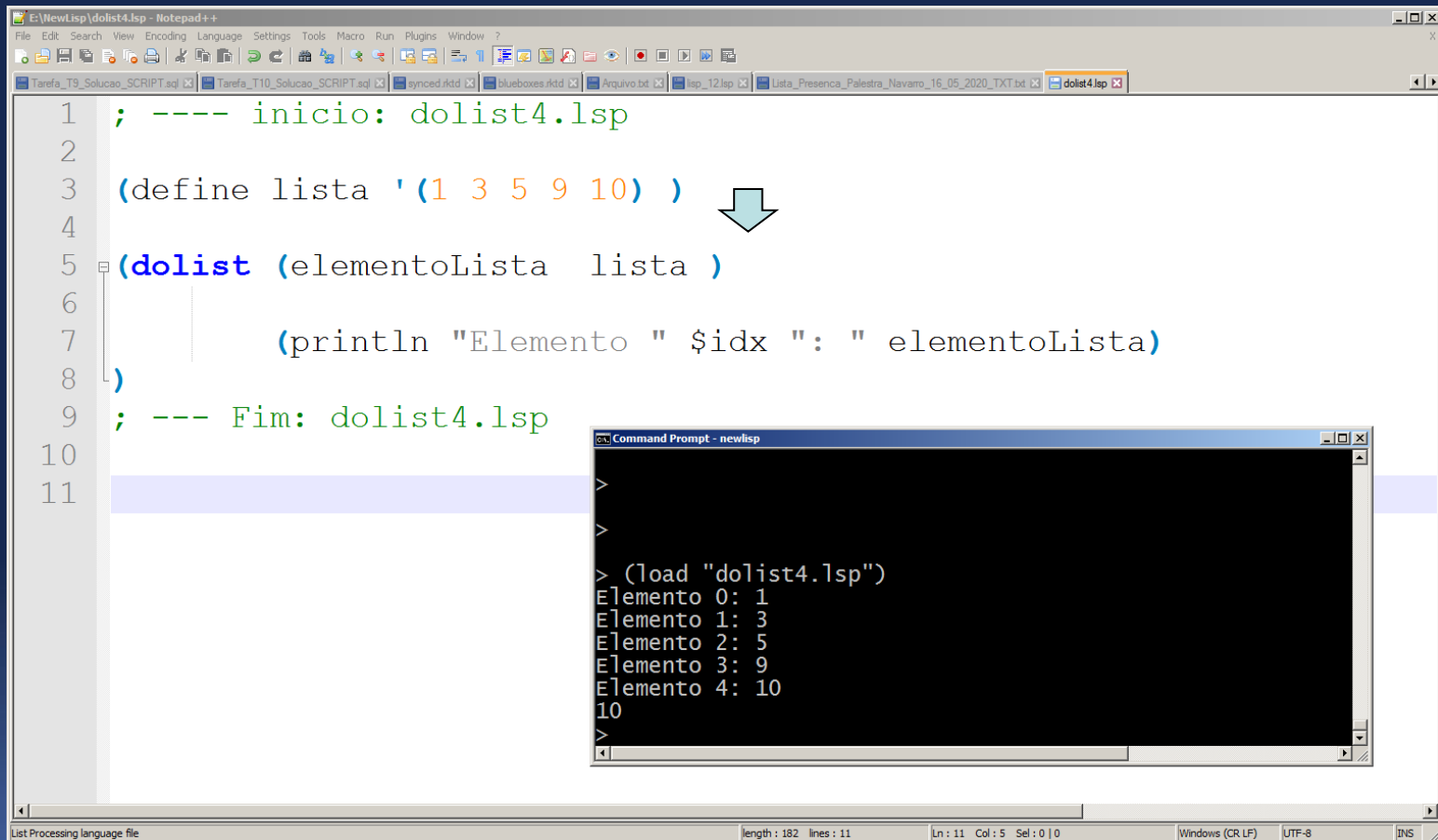
Elemento 0: 1
Elemento 1: 3
Elemento 2: 5
Elemento 3: 9
Elemento 4: 10
5
>
>
```

List Processing language file | length : 243 lines : 12 | Ln : 12 Col : 5 Sel : 0 | 0 | Windows (CR LF) UTF-8 INS



Contador de Loop mantido por newLISP

- ✓ newLISP automaticamente mantém um contador de loop chamado **\$idx**



The screenshot displays a Notepad++ editor window with a newLISP script named `dolist4.lsp`. The script defines a list `lista` containing the values 1, 3, 5, 9, and 10. It then uses the `dolist` function to iterate over this list, printing each element along with the index stored in the special variable `$idx`. A blue arrow points from the `$idx` variable in the script to the Command Prompt window. The Command Prompt window shows the output of the script, displaying each element and its corresponding index from 0 to 4. The status bar at the bottom of the Notepad++ window indicates the file is a "List Processing language file" with a length of 182 characters and 11 lines.

```
1 ; ---- inicio: dolist4.lsp
2
3 (define lista '(1 3 5 9 10) )
4
5 (dolist (elementoLista lista )
6
7     (println "Elemento " $idx ": " elementoLista)
8 )
9 ; --- Fim: dolist4.lsp
10
11
```

Command Prompt - newlisp

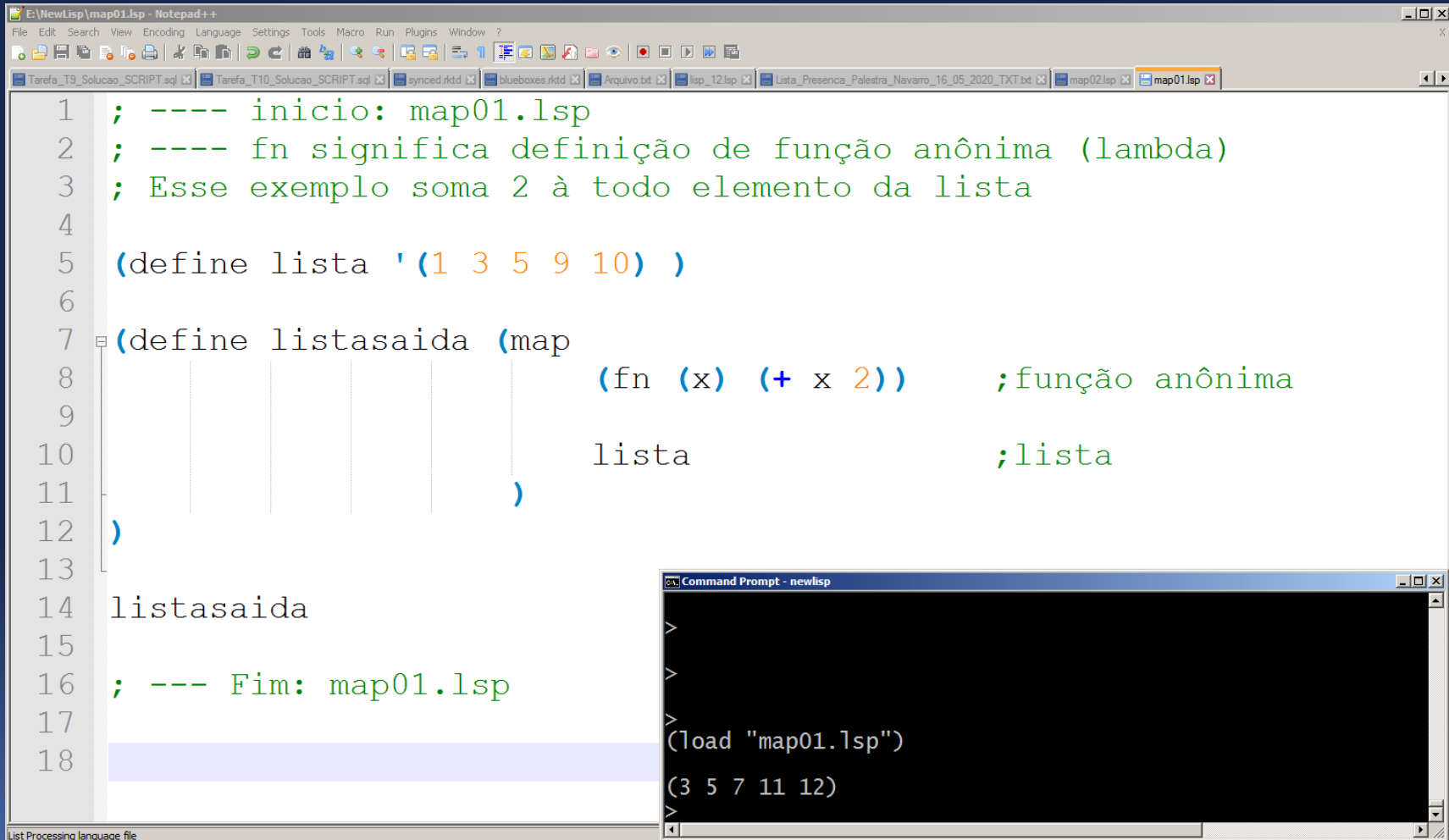
```
>
>
> (load "dolist4.lsp")
Elemento 0: 1
Elemento 1: 3
Elemento 2: 5
Elemento 3: 9
Elemento 4: 10
10
>
```

List Processing language file | length : 182 | lines : 11 | Ln : 11 Col : 5 Sel : 0 | 0 | Windows (CR LF) UTF-8 | DNS



Função map

- ✓ A função **map** pode ser usada para se aplicar uma **função** à todo elemento de uma lista;
- ✓ A **função** a ser aplicada aos elementos da lista é definida de forma anônima por **fn**.



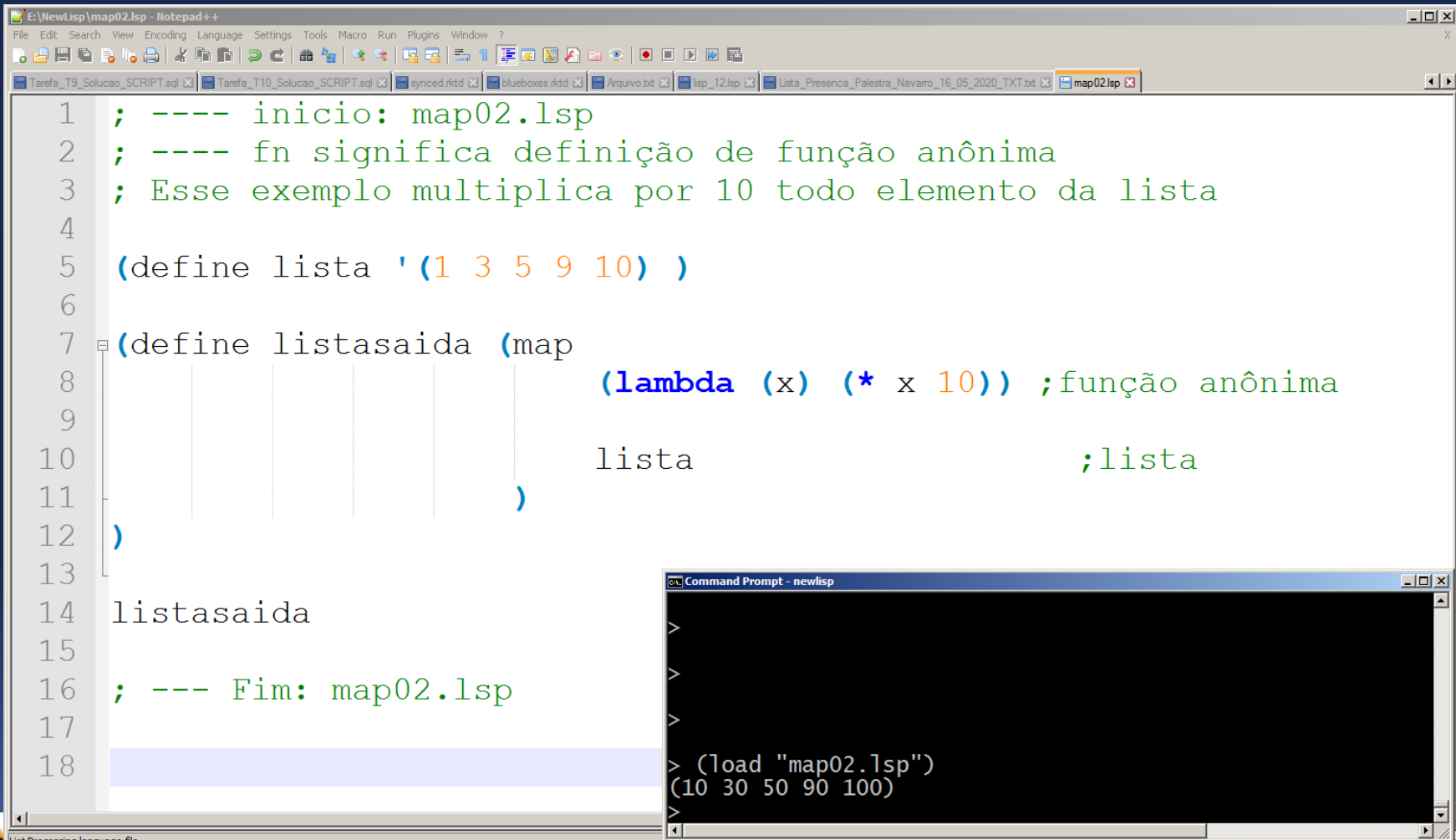
```
1 ; ---- inicio: map01.lsp
2 ; ---- fn significa definição de função anônima (lambda)
3 ; Esse exemplo soma 2 à todo elemento da lista
4
5 (define lista '(1 3 5 9 10) )
6
7 (define listasaida (map
8   (fn (x) (+ x 2)) ;função anônima
9   lista ;lista
10 )
11 )
12
13
14 listasaida
15
16 ; --- Fim: map01.lsp
17
18
```

```
>
>
>
(load "map01.lsp")
(3 5 7 11 12)
>
```



Função map

- ✓ A função **map** pode ser usada para se aplicar uma **função** à todo elemento de uma lista;
- ✓ A **função** a ser aplicada aos elementos da lista pode ser anonimamente definida por expressão **lambda** (que equivale ao **fn**)

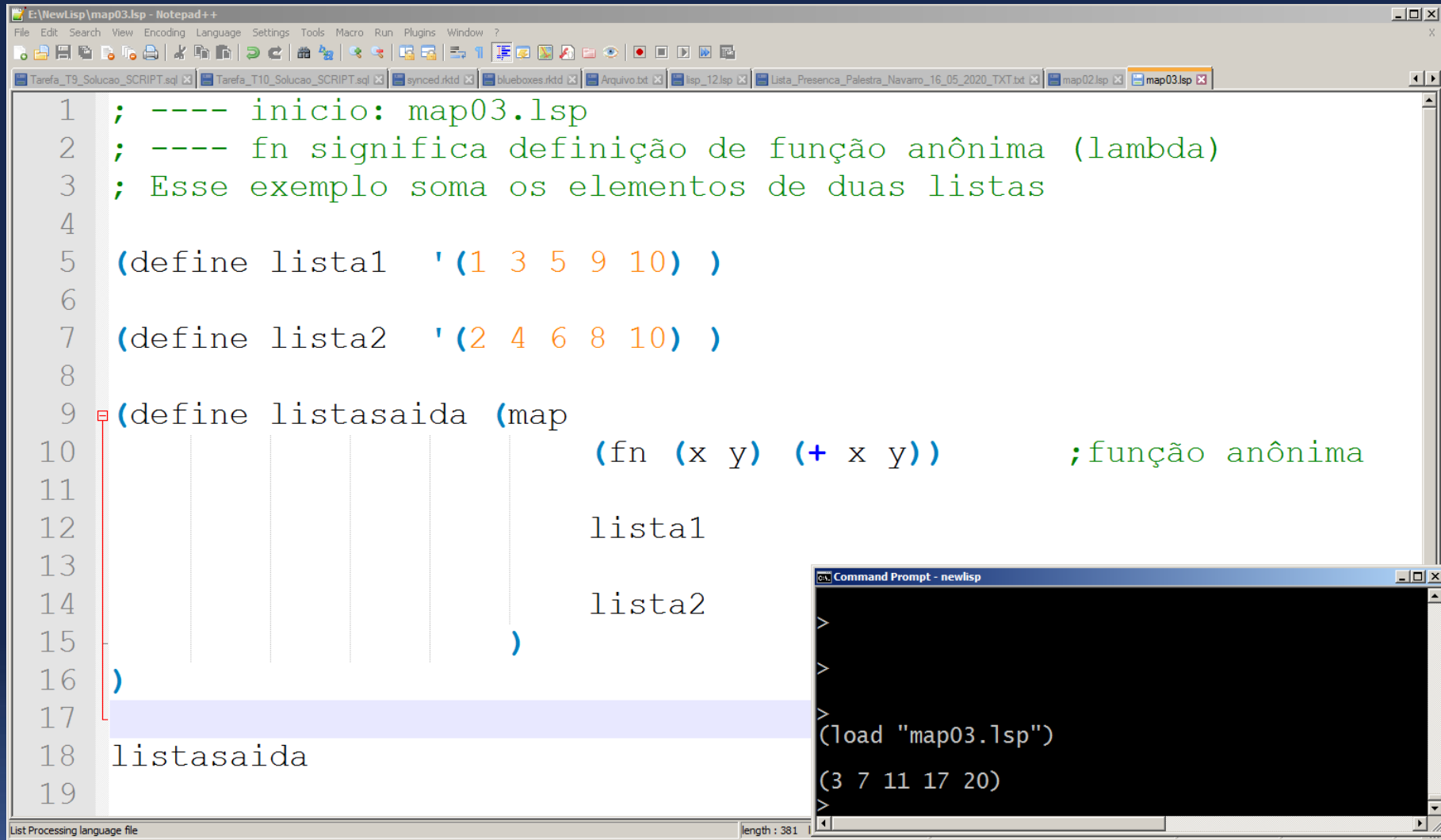


```
1 ; ---- inicio: map02.lsp
2 ; ---- fn significa definição de função anônima
3 ; Esse exemplo multiplica por 10 todo elemento da lista
4
5 (define lista '(1 3 5 9 10) )
6
7 (define listasaida (map
8   (lambda (x) (* x 10)) ;função anônima
9   lista                  ;lista
10 )
11 )
12
13 listasaida
14
15
16 ; --- Fim: map02.lsp
17
18
```

```
Command Prompt - newlisp
>
>
>
> (load "map02.lsp")
(10 30 50 90 100)
>
```



Função map



The image shows a Notepad++ editor window titled "E:\NewLisp\map03.lsp - Notepad++" with a menu bar (File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, Window, ?) and a toolbar. The editor contains the following Lisp code:

```
1 ; ---- inicio: map03.lsp
2 ; ---- fn significa definição de função anônima (lambda)
3 ; Esse exemplo soma os elementos de duas listas
4
5 (define lista1 '(1 3 5 9 10) )
6
7 (define lista2 '(2 4 6 8 10) )
8
9 (define listasaida (map
10                     (fn (x y) (+ x y)) ;função anônima
11                     lista1
12                     lista2
13                     )
14 )
15
16
17
18 listasaida
19
```

The Command Prompt window, titled "Command Prompt - newlisp", shows the execution of the script:

```
>
>
> (load "map03.lsp")
(3 7 11 17 20)
>
```

The status bar at the bottom of the Notepad++ window indicates "List Processing language file" and "length : 381".

