

```

1: program TestExp;
2: uses CRT, PilhaE;
3:
4: function LerExpressao : string;
5: var
6:   Key : char;
7:   Exp : string;
8: begin
9:   writeln('Entre com caracteres '(' ou ')' ou <ENTER> para sair');
10:  Exp := '';
11:
12:  repeat
13:    Key := ReadKey;
14:
15:    if Key = #13 then
16:      break;
17:
18:    if Key in ['(', ')'] then
19:      begin
20:        write(Key);
21:        Exp := Exp + Key;
22:      end;
23:
24:  until false;
25:
26:  writeln;
27:  LerExpressao := Exp;
28: end;
29:
30: function TestarParenteses(Exp : string) : boolean;
31: var
32:   I, K : byte;
33:   Pilha : Pilha_Enc;
34:   Info : Tipo_Dô_Dado;
35: begin
36:   TestarParenteses := false;
37:   Inicializar(Pilha);
38:
39:   for I := 1 to length(Exp) do
40:     if Exp[I] = '(' then
41:       begin
42:         { Empilha a posicao do caractere '(' }
43:         Info.Dado := char(I);
44:         Empilhar(Info, Pilha)
45:       end
46:     else
47:       if Exp[I] = ')' then
48:         { Se a pilha estiver vazia, a expressao esta errada pois deveria
49:         ter pelo menos uma posicao de um caractere '(' na pilha }
50:         if Vazia(Pilha) then
51:           begin
52:             for K := 1 to I - 1 do
53:               write(' ');
54:               writeln('^');
55:             exit;
56:           end
57:         else
58:           { Retira a posicao do caractere '(' do topo da pilha }
59:           Desempilhar(Info, Pilha);
60:
61:   if Vazia(Pilha) then

```

```

62:     TestarParenteses := true
63: else
64:     { Se a Pilha nao estiver vazia, ha pelo menos um caractere
65:       '(' sobrando e a sua posicao esta no topo da pilha }
66:     begin
67:         Desempilhar(Info, Pilha);
68:
69:         if byte(Info.Dado) > 1 then
70:             write(' ': byte(Info.Dado) - 1);
71:
72:             writeln('^');
73:         end;
74: end;
75:
76: BEGIN
77:     if not TestarParenteses(LerExpressao) then
78:         writeln('Encontrado um erro na expressao');
79: END.

```