

```

1: program TestHash;
2: uses TabHash;
3:
4: var
5:     Tab : Tab_Hash;
6:
7: procedure GravarTabelaEmArquivo(var T : Tab_Hash);
8: var
9:     Arq : text;
10:    I : Pos_Tab_Hash;
11: begin
12:     {$I-}
13:     assign(Arq, 'TabHash.txt');
14:     reset(Arq);
15:     if IOResult <> 0 then
16:         rewrite(Arq);
17:
18:     append(Arq);
19:     writeln(Arq, 'Situacao da tabela:');
20:
21:     for I := 1 to MAX do
22:     begin
23:         write(Arq, 'TabHash[' , I, ']' = ');
24:
25:         case Tab[I].Situacao of
26:             Ocupada      : writeln(Arq, 'Ocupada (Chave = ', Tab[I].Dado.Chave,
27: '));
28:             Desocupada  : writeln(Arq, 'Desocupada');
29:             NuncaUsada  : writeln(Arq, 'Nunca usada');
30:         end;
31:
32:         writeln(Arq);
33:         close(Arq);
34:     {$I+}
35: end;
36:
37: procedure ChecarCaracteristicas(var T : Tab_Hash; Tam : integer; V, C :
boolean);
38: begin
39:     if Tam <> Tamanho(T) then
40:         writeln('ERRO: Tamanho esperado = ', Tam, ', obtido ', Tamanho(T));
41:     if V <> Vazia(T) then
42:         writeln('ERRO: Vazia esperado = ', V, ', obtido ', Vazia(T));
43:     if C <> Cheia(T) then
44:         writeln('ERRO: Cheia esperado = ', C, ', obtido ', Cheia(T));
45: end;
46:
47: procedure TestarInserir(var T : Tab_Hash);
48: var
49:     I : Pos_Tab_Hash;
50:     Dado : Tipo_do_Dado;
51: begin
52:     for I := 1 to TabHash.MAX do
53:     begin
54:         Dado.Chave := I * TabHash.MAX;
55:         if not Inserir(T, Dado) then
56:             begin
57:                 writeln('ERRO: Nao foi possivel inserir a chave = ',
58: Dado.Chave);
59:                 exit;

```

```

60:         end
61:     end;
62:
63:     Dado.Chave := 1;
64:     if Inserir(T, Dado) then
65:         begin
66:             writeln('ERRO: Nao deveria ser possivel inserir a chave = ',
67:                 Dado.Chave);
68:             exit;
69:         end;
70:
71:         writeln('Ok: Inserir funcionado perfeitamente');
72:     end;
73:
74:
75: procedure TestarObter(var T : Tab_Hash);
76: var
77:     I : Pos_Tab_Hash;
78:     Chave : Tipo_da_Chave;
79:     Dado : Tipo_do_Dado;
80: begin
81:     for I := 1 to TabHash.MAX do
82:         begin
83:             Chave := I * TabHash.MAX;
84:             if not Obter(T, Chave, Dado) then
85:                 begin
86:                     writeln('ERRO: Nao foi possivel obter a chave = ', Chave);
87:                     exit;
88:                 end
89:             end;
90:
91:             Chave := 1;
92:             if Obter(T, Chave, Dado) then
93:                 begin
94:                     writeln('ERRO: Nao deveria ser possivel obter a chave = ',
95:                         Chave);
96:                     exit;
97:                 end;
98:
99:                 writeln('Ok: Obter funcionado perfeitamente');
100:         end;
101:
102: procedure TestarRemover(var T : Tab_Hash);
103: var
104:     I : Pos_Tab_Hash;
105:     Chave : Tipo_da_Chave;
106: begin
107:     Chave := 1;
108:     if Remover(T, Chave) then
109:         begin
110:             writeln('ERRO: Nao deveria ser possivel remover a chave = ', Chave);
111:             exit;
112:         end;
113:
114:         for I := 1 to TabHash.MAX do
115:             begin
116:                 Chave := I * TabHash.MAX;
117:                 if not Remover(T, Chave) then
118:                     begin
119:                         writeln('ERRO: Nao foi possivel remover a chave = ', Chave);
120:                         exit;

```

```

121:         end
122:     end;
123:
124:     writeln('Ok: Remover funcionado perfeitamente');
125: end;
126:
127: procedure TestarInserir2(var T : Tab_Hash);
128: var
129:     I : Pos_Tab_Hash;
130:     Dado : Tipo_do_Dado;
131: begin
132:     for I := 1 to 5 do
133:     begin
134:         Dado.Chave := I;
135:         if not Inserir(T, Dado) then
136:         begin
137:             writeln('ERRO: Nao foi possivel inserir a chave = ', Dado.Chave);
138:             exit;
139:         end
140:     end;
141:
142:     for I := 1 to 5 do
143:     begin
144:         Dado.Chave := TabHash.MAX + I;
145:
146:         if not Inserir(T, Dado) then
147:         begin
148:             writeln('ERRO: Nao foi possivel inserir a chave = ', Dado.Chave);
149:             exit;
150:         end;
151:     end;
152:
153:     for I := 1 to 5 do
154:     begin
155:         if T[I + 1].Dado.Chave <> I then
156:             writeln('ERRO1: T[' , I + 1, ']=', T[I + 1].Dado.Chave,
157:                 ' (Esperado:', I + 1, ')');
158:
159:         if T[I + 6].Dado.Chave <> (TabHash.MAX + I) then
160:             writeln('ERRO2: T[' , (I + 6), ']=', T[I + 6].Dado.Chave,
161:                 ' (Esperado:', (TabHash.MAX + I), ')');
162:         end;
163:
164:         writeln('Ok: Inserir funcionado perfeitamente');
165:     end;
166:
167: BEGIN
168:     Inicializar(Tab);
169:
170:     ChecarCaracteristicas(Tab, 0, true, false);
171:     TestarInserir(Tab);
172:     GravarTabelaEmArquivo(Tab);
173:     ChecarCaracteristicas(Tab, TabHash.MAX, false, true);
174:
175:     TestarObter(Tab);
176:
177:     TestarRemover(Tab);
178:     ChecarCaracteristicas(Tab, 0, true, false);
179:     GravarTabelaEmArquivo(Tab);
180:
181:     TestarInserir(Tab);

```

```
182:    TestarObter (Tab);
183:    TestarRemover (Tab);
184:
185:    TestarInserir2 (Tab);
186:    GravarTabelaEmArquivo (Tab);
187: END.
```