

```

1: unit FilaE;
2: interface
3:
4: type
5:     Tipo_do_Dado = record
6:         Dado : char;
7:     end;
8:     PNo = ^No;
9:     No = record
10:         Info : Tipo_do_Dado;
11:         Prox : PNo;
12:     end;
13:     Fila_Enc = record
14:         Inicio,
15:         Fim      : PNo;
16:         Tamanho : integer;
17:     end;
18:
19: procedure Inicializar (var Fila : Fila_Enc);
20: function Vazia (var Fila : Fila_Enc) : boolean;
21: function Cheia (var Fila : Fila_Enc) : boolean;
22: function Tamanho (var Fila : Fila_Enc) : integer;
23: procedure Inserir (Info : Tipo_do_Dado; var Fila : Fila_Enc);
24: procedure Retirar (var Info : Tipo_do_Dado; var Fila : Fila_Enc);
25: procedure Frente (var Info : Tipo_do_Dado; var Fila : Fila_Enc);
26:
27: implementation
28:
29: procedure Inicializar (var Fila : Fila_Enc);
30: {
31:     Objetivos: Inicializa a fila, atribuindo nil ao inicio e ao fim da fila e
32:     zerando o tamanho da fila
33: }
34: begin
35:     Fila.Inicio := nil;
36:     Fila.Fim := nil;
37:     Fila.Tamanho := 0
38: end;
39:
40:
41: function Vazia (var Fila : Fila_Enc) : boolean;
42: {
43:     Objetivos: Retorna true se o numero de Nos na fila for igual a zero. Caso
44:     contrario, retorna false
45: }
46: begin
47:     Vazia := Fila.Tamanho = 0;
48: end;
49:
50:
51: function Cheia (var Fila : Fila_Enc) : boolean;
52: {
53:     Objetivos: Retorna true se o maior bloco de memoria disponivel for menor
54:     que o tamanho de um No. Caso contrario, retorna false.
55: }
56: begin
57:     Cheia := Maxavail < SizeOf(No);
58: end;
59:
60:
61: function Tamanho (var Fila : Fila_Enc) : integer;

```

```

62: {
63:   Objetivos: Retorna o numero de Nos existentes na fila.
64: }
65: begin
66:   Tamanho := Fila.Tamanho;
67: end;
68:
69:
70: procedure Inserir (Info : Tipo_do_Dado; var Fila : Fila_Enc);
71: {
72:   Objetivos: Insere um novo No no final da fila.
73: }
74: var
75:   PTemp : PNo;
76: begin
77:   { Testa se a fila esta cheia. Se estiver da uma mensagem e sai do programa
   }
78:   if Cheia(Fila) then
79:     begin
80:       writeln('ERRO: Overflow da fila. ');
81:       halt;
82:     end;
83:
84:     { Cria um novo No, coloca as informacoes nele e atribui nil a Prox }
85:     New(PTemp);
86:     PTemp^.Info := Info;
87:     PTemp^.Prox := nil;
88:
89:     { Se a fila estiver vazia, sera inserido o primeiro No da fila }
90:     if Vazia(Fila) then
91:       begin
92:         Fila.Inicio := PTemp;
93:         Fila.Fim    := PTemp;
94:       end
95:       { Senao sera inserido no final da fila }
96:     else
97:       begin
98:         Fila.Fim^.Prox := PTemp;
99:         Fila.Fim      := PTemp;
100:      end;
101:
102:      { Contador de Nos aumenta de um. }
103:      inc(Fila.Tamanho);
104: end;
105:
106:
107: procedure Retirar (var Info : Tipo_do_Dado; var Fila : Fila_Enc);
108: {
109:   Objetivos: Retira um No do inicio da fila
110: }
111: var
112:   PTemp : PNo;
113: begin
114:   { Testa se a fila esta vazia. Se estiver da uma mensagem e sai do programa
   }
115:   if Vazia(Fila) then
116:     begin
117:       writeln('ERRO: Underflow da fila. ');
118:       halt;
119:     end;
120:

```

```

121:      { Faz com que Info contenha a informacao do primeiro No da fila. O primeiro
122:        da fila passa a ser o proximo da fila e o primeiro No, desalocado da
123:        memoria. Depois, o contador de Nos diminui de um. }
124:      Info := Fila.Inicio^.Info;
125:      PTemp := Fila.Inicio;
126:      Fila.Inicio := Fila.Inicio^.Prox;
127:      Dispose(PTemp);
128:      dec(Fila.Tamanho);
129:
130:      { Se a fila esta vazia, entao o Fim da fila deve apontar par nil }
131:      if Vazia(Fila) then
132:        Fila.Fim := nil;
133:      end;
134:
135:
136: procedure Frente (var Info : Tipo_do_Dado; var Fila : Fila_Enc);
137: {
138:   Objetivos: Faz com que Info contenha as informacoes guardadas
139:             no No que esta na Frente (Inicio) da Fila
140: }
141: begin
142:   { Testa se a fila esta vazia. Se estiver da uma mensagem e sai do programa
143:   }
143:   if Vazia(Fila) then
144:     begin
145:       writeln('ERRO: Underflow da fila. ');
146:       halt;
147:     end;
148:
149:   { Faz com que Info contenha a informacao do primeiro No da fila }
150:   Info := Fila.Inicio^.Info;
151: end;
152:
153: end.

```