

cedimento Pesquisa ignora completamente os campos BitE e BitD. Logo, nenhum tempo adicional é necessário para pesquisar na árvore SBB.

Os quatro procedimentos EE, ED, DD e DE são utilizados nos procedimentos Insere e Retira, com o objetivo de eliminar dois apontadores horizontais sucessivos. O Programa 5.12 mostra a implementação desses procedimentos.

Programa 5.12 Procedimentos auxiliares para árvores SBB

```

procedure EE (var Ap: TipoApontador);
var Apl: TipoApontador;
begin
  Apl := Ap^.Esq;      Ap^.Esq := Apl^.Dir;  Apl^.Dir := Ap;
  Apl^.BitE := Vertical; Ap^.BitE := Vertical; Ap := Apl;
end; { EE }

procedure ED (var Ap: TipoApontador);
var Apl, Ap2: TipoApontador;
begin
  Apl := Ap^.Esq;      Ap2 := Apl^.Dir;      Apl^.BitD := Vertical;
  Ap^.BitE := Vertical; Apl^.Dir := Ap2^.Esq; Ap2^.Esq := Apl;
  Ap^.Esq := Ap2^.Dir; Ap2^.Dir := Ap;      Ap := Ap2;
end; { ED }

procedure DD (var Ap: TipoApontador);
var Apl: TipoApontador;
begin
  Apl := Ap^.Dir;      Ap^.Dir := Apl^.Esq;  Apl^.Esq := Ap;
  Apl^.BitD := Vertical; Ap^.BitD := Vertical; Ap := Apl;
end; { DD }

procedure DE (var Ap: TipoApontador);
var Apl, Ap2: TipoApontador;
begin
  Apl := Ap^.Dir;      Ap2 := Apl^.Esq;      Apl^.BitE := Vertical;
  Ap^.BitD := Vertical; Apl^.Esq := Ap2^.Dir; Ap2^.Dir := Apl;
  Ap^.Dir := Ap2^.Esq; Ap2^.Esq := Ap;      Ap := Ap2;
end; { DE }

```

O procedimento **Insere** tem uma interface idêntica à interface do procedimento **Insere** para árvores sem balanceamento, conforme pode ser visto no Programa 5.13. Para que isso seja possível, o procedimento **Insere** simplesmente chama outro procedimento interno de nome **IInsere**, cuja interface contém dois parâmetros a mais que o procedimento **Insere**, a saber: o parâmetro **IAp** indica que a inclinação do apontador toma o valor horizontal sempre que um nó é elevado para o nível seguinte durante uma inserção, e o parâmetro **Fim** toma o valor **true** quando a propriedade SBB é restabelecida e nada mais é necessário fazer.

Programa 5.13 Procedimento para inserir na árvore SBB

```

procedure Insere (x: TipoRegistro; var Ap: TipoApontador);
var Fim: boolean; IAp: TipoInclinacao;
procedure IInsere (x: TipoRegistro; var Ap: TipoApontador;
  var IAp: TipoInclinacao; var Fim: boolean);
begin
  if Ap = nil
  then begin
    new (Ap); IAp := Horizontal; Ap^.Reg := x;
    Ap^.BitE := Vertical; Ap^.BitD := Vertical;
    Ap^.Esq := nil; Ap^.Dir := nil;
    Fim := false;
  end
  else
    if x.Chave < Ap^.Reg.Chave
    then begin
      IInsere (x, Ap^.Esq, Ap^.BitE, Fim);
      if not Fim
      then if Ap^.BitE = Horizontal
        then begin
          if Ap^.Esq^.BitE = Horizontal
          then begin EE (Ap); IAp := Horizontal; end
          else if Ap^.Esq^.BitD = Horizontal
            then begin ED (Ap); IAp := Horizontal; end;
          end
          else Fim := true;
        end
      else
        if x.Chave > Ap^.Reg.Chave
        then begin
          IInsere (x, Ap^.Dir, Ap^.BitD, Fim);
          if not Fim
          then if Ap^.BitD = Horizontal
            then begin
              if Ap^.Dir^.BitD = Horizontal
              then begin DD (Ap); IAp := Horizontal; end
              else if Ap^.Dir^.BitE = Horizontal
                then begin DE (Ap); IAp := Horizontal; end;
              end
              else Fim := true;
            end
          end
        end
      begin { IInsere }
    begin { Insere }
      IInsere (x, Ap, IAp, Fim);
    end; { Insere }

```