

## Lampiran Listing Program Splay\_Tree

```
{*****}
*
* NAMA PROGRAM      : SPLAY_TREE.PAS
* TANGGAL PEMBUATAN : SELASA, 3 OKTOBER 2000
* PEMBUAT PROGRAM   : APIP ICHSAN CAHYANA-J101951179
* KETERANGAN PROGRAM : PROSES PENYAJIAN POHON SPLAY
*
*****}
program splaytree;

uses crt, graph;

const
    { * Pendefinisian warna pada program * }
    Hitam      = 0;      Hitam_t   = 8;
    Biru       = 1;      Biru_t    = 9;
    Hijau      = 2;      Hijau_t   = 10;
    Cyan       = 3;      Cyan_t    = 11;
    Merah      = 4;      Merah_t   = 12;
    Magenta    = 5;      Magenta_t = 13;
    coklat     = 6;      Coklat_t  = 14;
    Putih      = 7;      Putih_t   = 15;
    Kedip      = 128;

    { * Pendefinisian istilah dan Keterangan * }
    ps0 = 'Tekan sebarang tombol untuk memulai proses...!';
    ps1 = 'Tekan < Enter > untuk melanjutkan proses....!';
    ps2 = '** PERINGATAN **';
    ps3 = '** KETERANGAN **';
    bel = ^G;

type tree = ^simpul;
    simpul = record
        info      : char;
        kiri, kanan : tree;
    end;

var akar      : tree;
    huruf,p   : char;
    i,x,y     : integer;
    lagi      : char;
    pilihan   : byte;
    ada       : boolean;
    Selesai   : boolean;
    HasilIO   : integer;
```

```

{*****
 *          <<<< Bagian Aksesoris Program >>>>          *
*****}

{*** Procedure Untuk Membunyikan Nada Peringatan ***}

Procedure Beep(noot,dly:integer);

begin
  sound(noot);delay(dly);nosound;
end; {* akhir procedure Beep *}

{*** Procedure Untuk Membunyikan Nada Musik Program ***}

Procedure Musik;

const
  frk : array[1..13] of integer =
    (0,0,213,227,256,288,320,341,384,427,453,512,576);

  noot: array[0..29] of byte =
    (29,6,0,8,6,0,6,9,6,5,6,0,10,6,0,6,11,10,8,6,10,13,6,5,0,5,3,7,6,0);

  jd  : array[0..29] of byte =
    (60,4,4,6,2,2,2,4,4,4,4,4,6,2,2,2,4,4,4,4,4,4,2,2,2,2,4,4,12,24);

var
  jmlnot,point,base,I : integer;
  note,jeda            : array[0..230] of byte;
  frek                : array[1..25] of integer;

begin
  move(frk,frek,26);
  move(noot,note,30);
  move(jd,jeda,30);
  jmlnot:=note[0];
  base:=jeda[0];
  for I:=1 to jmlnot do
    if note[I] = 0 then delay(jeda[I]*base)
    else
      begin
        point:=note[I];
        sound(frek[point]);delay(jeda[I]*base);nosound;
      end
    end; {* akhir dari procedure musik *}

```

```

{*****
 * <<<< Bagian Operasi Proses Program Pohon Splay >>>> *
*****}

{*** Procedure Untuk Membentuk Pohon ***}

function baru(hrf: char) : tree;

var b: tree;

begin
  new(b);
  b^.info := hrf;
  b^.kanan := nil;
  b^.kiri := nil;
  baru := b;
end;  {* akhir fungsi baru *}

{*** Procedure Untuk Menyisipkan Pohon ***}

procedure sisip_pohon(var pohon: tree; hrf: char);
begin
  if pohon = nil then
    pohon := baru(hrf)
  else
    if pohon^.info > hrf then
      sisip_pohon(pohon^.kiri, hrf)
    else
      if pohon^.info < hrf then
        sisip_pohon(pohon^.kanan, hrf)
      else
        begin
          textcolor(merah_t+kedip);writeln(' ':20,bel,ps2);
          textbackground(hitam);textcolor(magenta_t);
          write(' Terjadi duplikasi, Simpul "',hrf);
          writeln('" sudah ada dalam pohon.');

```

```

begin
  textcolor(merah_t+kedip);writeln(' ':20,bel,ps2);
  textcolor(magenta_t);
  write(' ':6,'Terjadi duplikasi, Simpul "',hrf);
  writeln('" sudah ada dalam pohon. ');
  writeln('          << Pilihan Proses >>');
  writeln('          [1]. Menyisipkan simpul');
  writeln('          [0]. Keluar dari proses');
  pilihan:=9;
  while (pilihan<0) or (pilihan>3) do
    begin
      textcolor(merah_t);
      write(' ':6,'Tentukan pilihan anda (0 - 1) : ');
      readln(pilihan);writeln;
    end;
  case pilihan of
    1 : begin
        textcolor(putih_t);
        write('   b. Masukan simpul yang akan disisipkan : ');
        while not eoln do
          begin
            read(huruf);sisip_simpul(akar,huruf);
          end
        end;
      end;
    0 : begin
        writeln('   Tekan <enter> untuk keluar dari proses ini...!');
        readln;
        exit;
      end
  end
end
end
end; (* akhir procedure sisip_simpul *)

(***) Procedure Untuk Mencari Simpul Pohon (***)

procedure cari_simpul(var Ada:boolean; var pohon: tree; hrf: char);
begin
  Ada:=true;
  if pohon = nil then Ada:=false
  else
    if pohon^.info > hrf then
      cari_simpul(ada, pohon^.kiri, hrf)
    else
      if pohon^.info < hrf then
        cari_simpul(ada, pohon^.kanan, hrf)
      else
        writeln('   b. Simpul "',hrf,'" terdapat dalam pohon. ');
      end;
  end; (* akhir procedure cari_simpul *)

```

```

{*** Procedure Untuk Menghapus Simpul ***}

procedure hapus_simpul(var posisi: tree);
var bantu: tree;
begin
  if posisi = nil then
  begin
    write('  Tidak bisa menghapus simpul yang tidak ada');
    exit;
  end
  else
  if posisi^.kanan = nil then
  begin
    bantu := posisi;
    posisi := posisi^.kiri;
    dispose(bantu);
  end
  else
  if posisi^.kiri = nil then
  begin
    bantu := posisi;
    posisi := posisi^.kanan;
    dispose(bantu);
  end
  else
  begin
    bantu := posisi^.kanan;
    while bantu^.kiri <> nil do
      bantu := bantu^.kiri;
    bantu^.kiri := posisi^.kiri;
    bantu := posisi;
    posisi := posisi^.kanan;
    dispose(bantu);
  end;
end; {* akhir procedure hapus_simpul *}

{*** Procedure Untuk Menghapus Simpul kunci ***}

procedure hapus_kunci(var pohon: tree; target: char);

var Ada : boolean;
    lagi : char;
    hrf: char;

begin
  if pohon = nil then
  begin
    textcolor(merah_t+kedip);writeln(' ':20,bel,ps2);
    textcolor(magenta_t);
    writeln('      Simpul "',target,'"tidak ada dalam pohon');
    writeln('      << Pilihan Proses >>');
    writeln('      [1]. Menghapus simpul lain');
    writeln('      [0]. Keluar dari proses');
    pilihan:=9;
    while (pilihan<0) or (pilihan>3) do

```

```

begin
  textcolor(merah_t);
  write(' ':6,'Tentukan pilihan anda (0 - 1) : ');
  readln(pilihan);writeln;
end;
case pilihan of
1 : begin
  textcolor(putih_t);
  write('  b. Masukkan simpul yang akan dihapus : ');
  while not eoln do
    begin
      read(target);hapus_kunci(akar,target);
    end;
  end;
0 : begin
  writeln('  Tekan <enter> untuk keluar dari proses ini...!');
  readln;
  exit;
end
end
end
else
if pohon^.info = target then
begin
  writeln('  Simpul "',target,'" telah dihapus dari pohon');
  hapus_simpul(pohon);
end
else
if target < pohon^.info then
  hapus_kunci(pohon^.kiri,target)
else
  hapus_kunci(pohon^.kanan,target)
end;  {* akhir procedure hapus_kunci *}

{*****
* Procedure Untuk Membaca Kunjungan Pohon *
*****}

{*** Procedure Untuk Membaca Kunjungan PreOrder ***}

procedure r_pre(pohon: tree);
begin
  if pohon <> nil then
  begin
    write(pohon^.info,' ');
    r_pre(pohon^.kiri);
    r_pre(pohon^.kanan);
  end;
end;  {* akhir procedure r_pre *}

```

```
{** Procedure Untuk Membaca Kunjungan InOrder **}
```

```
procedure r_in(pohon: tree);
begin
  if pohon <> nil then
    begin
      r_in(pohon^.kiri);
      write(pohon^.info, ' ');
      r_in(pohon^.kanan);
    end;
  end;
end; (* akhir procedure r_in *)
```

```
{** Procedure Untuk Membaca Kunjungan PostOrder **}
```

```
procedure r_pos(pohon: tree);
begin
  if pohon <> nil then
    begin
      r_pos(pohon^.kiri);
      r_pos(pohon^.kanan);
      write(pohon^.info, ' ');
    end;
  end;
end; (* akhir procedure r_pos *)
```

```
{*****
 * Procedure Operasi Splaying Pohon Splay *
*****}
```

```
procedure treesplay(var pohon: tree; target: char);
var bantu, anak, lastsmall, firstlarge, null: tree;
```

```
procedure linkkiri;
begin
  lastsmall^.kanan := bantu;
  lastsmall      := bantu;
  bantu          := bantu^.kanan;
end; (* akhir procedure linkkiri *)
```

```
procedure linkkanan;
begin
  firstlarge^.kiri := bantu;
  firstlarge      := bantu;
  bantu          := bantu^.kiri;
end; (* akhir procedure linkkanan *)
```

```
procedure rotasikiri;
var anakkanan : tree;
begin
  anakkanan      := bantu^.kanan;
  bantu^.kanan   := anakkanan^.kiri;
  anakkanan^.kiri := bantu;
  bantu          := anakkanan;
end; (* akhir procedure rotasikiri *)
```

```

procedure rotasikanan;
var anakkiri : tree;
begin
    anakkiri      := bantu^.kiri;
    bantu^.kiri   := anakkiri^.kanan;
    anakkiri^.kanan := bantu;
    bantu        := anakkiri;
end;  (* akhir procedure rotasikanan *)

{***** Bagian Operasi Splaying *****}

begin
    null^.info := target;
    bantu      := pohon;
    lastsmall  := null;
    firstlarge := null;
    if bantu^.info = target then
    begin
        textcolor(merah_t+kedip);writeln(' ':20,bel,ps2);
        textcolor(magenta_t);write(' ':6,'Simpul "',target,'" yang ');
        writeln('diakses sudah merupakan akar ');
        writeln(' ':6,'sehingga pohon splay yang dihasilkan akan sama');
        writeln(' ':6,'dengan pohon splay sebelumnya');
        exit;
        readln;
    end;
    while bantu^.info <> target do
    if bantu^.info < target then
    begin
        anak := bantu^.kanan;
        if target = anak^.info then
            linkkiri
        else
            if target > anak^.info then
            begin
                rotasikiri;
                linkkiri
            end
            else
            begin
                linkkiri;
                linkkanan
            end
        end
    end
    else
    begin
        anak := bantu^.kiri;
        if target = anak^.info then
            linkkanan
        else
            if target < anak^.info then
            begin
                rotasikanan;
                linkkanan
            end
        end
    end
end

```



```

begin
    linkkanan;
    linkkiri
end
end;

if bantu = null then
begin
    writeln('    c. Target telah disisipkan sebagai akar. ');
    pohon := baru(target);
    pohon^.kiri := null;
    pohon^.kanan := null;
    bantu := pohon
end
else
    writeln('    c. Target ditemukan dan menjadi akar. ');
    lastsmall^.kanan := bantu^.kiri;
    firstlarge^.kiri := bantu^.kanan;
    bantu^.kanan := null^.kiri;
    bantu^.kiri := null^.kanan;
    pohon := bantu;
    null^.info := Chr(0);
    null^.kiri := null;
    null^.kanan := null
end;  { * akhir procedure TreeSplay * }

{*****
 *      Bagian Menu Operasi Pohon Splay
 *      *****}

{***** Procedure Penyajian Menu Splay *****}

Procedure menu_splay(var pohon:tree);

begin
    clrscr;
    gotoxy(20,1);textcolor(hijau_t);
    writeln('<<< PROSES PENYAJIAN POHON SPLAY >>>');
    gotoxy(20,2);writeln('( Dengan preorder, inorder, postorder )');
    gotoxy(20,3);writeln('=====');
    writeln;textcolor(cyan_t);
    writeln('1. Proses pemasukan data');
    write('  Masukkan untaian simpul : ');
    while not eoln do
    begin
        read(huruf);
        sisip_pohon(akar,huruf);
    end;
    readln;
    textcolor(cyan_t);
    writeln('2. Proses Pembacaan Simpul Pohon Biner Sebelum Splaying');
    textcolor(biru_t);
    write('  a. Preorder  : ');r_pre(akar);readln;
    write('  b. Inorder   : ');r_in(akar);readln;
    write('  c. Postorder : ');r_pos(akar);readln;writeln;

```

```

lagi:='Y';
repeat
textcolor(cyan_t);
writeln('3. Proses pemasukan Data yang diakses');
write('  a. Masukan simpul yang diakses : ');
read(huruf);cari_simpul(ada,akar,huruf);
if not(ada) then
begin
writeln;textcolor(merah_t+kedip);writeln(' ':18,bel,ps2);
textcolor(biru_t);
writeln('      Simpul "',huruf,'"tidak ada dalam pohon');
writeln('      << Pilihan Proses >>');
writeln('      [1]. Menyisipkan simpul');
writeln('      [0]. Keluar dari proses');
pilihan:=9;
while (pilihan<0) or (pilihan>3) do
begin
textcolor(merah_t);
write(' ':7,'Tentukan pilihan anda (0 - 1) : ');
readln(pilihan);writeln;
end;
case pilihan of
1 : begin
textcolor(putih_t);
write('  b. Masukan simpul yang akan disisipkan : ');
while not eoln do
begin
read(huruf);sisip_simpul(akar,huruf);
end
end;
0 : begin
writeln('  Tekan <enter> untuk keluar dari proses ini....!');
readln;
exit;
end
end
end;
treesplay(akar,huruf);
textcolor(cyan_t);writeln;
writeln('4. Proses pembacaan simpul setelah operasi splaying');
textcolor(biru_t);
write('  a. Preorder  : ');r_pre(akar);readln;writeln;
write('  b. Inorder   : ');r_in(akar);readln;
write('  c. Postorder : ');r_pos(akar);readln;
writeln;
textcolor(putih_t);
write(' ':15,'Apakah akan mengakses simpul ? [Y]a/[T]idak : ');
lagi:=upcase(readkey);clrscr;
until lagi <>'Y';

end;  {* akhir procedure menu_splay *}

```

```

{***** Procedure Penyajian Menu Sisip Pohon Splay *****)

Procedure menu_sisip(var pohon:tree);

begin
  clrscr;
  gotoxy(19,1);textcolor(hijau_t);
  writeln('<<< PROSES PENYISIPAN SIMPUL POHON SPLAY >>>');
  gotoxy(23,2);writeln('( Dengan preorder,inorder,postorder )');
  gotoxy(23,3);writeln('=====');
  writeln;textcolor(cyan_t);
  textcolor(cyan_t);
  writeln('1. Pembacaan Simpul Pohon Splay Sebelum Penyisipan');
  textcolor(biru_t);
  write('  a. Preorder  : ');r_pre(akar);readln;
  write('  b. Inorder   : ');r_in(akar);readln;
  write('  c. Postorder  : ');r_pos(akar);readln;writeln;
  textcolor(cyan_t);writeln('2. Proses Penyisipan Simpul Pohon
Splay');
  write('  a. Masukkan simpul yang akan disisipkan : ');
  while not eoln do
  begin
    read(huruf);
    sisip_simpul(akar,huruf);
  end;
  readln;
  textcolor(cyan_t);
  writeln('3. Pembacaan Simpul Pohon Splay Sesudah Penyisipan');
  textcolor(biru_t);
  write('  a. Preorder  : ');r_pre(akar);readln;
  write('  b. Inorder   : ');r_in(akar);readln;
  write('  c. Postorder  : ');r_pos(akar);readln;writeln;
  lagi:='Y';
  repeat
  textcolor(cyan_t);
  writeln('4. Proses Pemasukan Data yang Diakses');
  write('  a. Masukan simpul yang diakses : ');
  read(huruf);cari_simpul(ada,akar,huruf);
  if not(ada) then
  begin
    writeln;textcolor(merah_t+kedip);writeln('  ':18,bel,ps2);
    textcolor(biru_t);
    writeln('      Simpul "',huruf,'"tidak ada dalam pohon');
    writeln('      << Pilihan Proses >>');
    writeln('      [1]. Menyisipkan simpul');
    writeln('      [0]. Keluar dari proses');
    pilihan:=9;
    while (pilihan<0) or (pilihan>3) do
    begin
      textcolor(merah_t);
      write('  ':7,'Tentukan pilihan anda (0 - 1) : ');
      readln(pilihan);writeln;
    end;
  end;
end;

```

```

case pilihan of
1 : begin
    textcolor(putih_t);
    write('    b. Masukan simpul yang akan disisipkan : ');
    while not eoln do
        begin
            read(huruf);sisip_simpul(akar,huruf);
        end
    end;
0 : begin
    writeln('    Tekan <enter> untuk keluar dari proses ini...!');
    readln;
    exit;
end
end;
treesplay(akar,huruf);
textcolor(cyan_t);writeln;
writeln('5. Proses Pembacaan Simpul Setelah Operasi Splaying');
textcolor(biru_t);
write('    a. Preorder   : ');r_pre(akar);readln;writeln;
write('    b. Inorder     : ');r_in(akar);readln;
write('    c. Postorder    : ');r_pos(akar);readln;
writeln;
textcolor(putih_t);
write(' ':15,'Apakah akan mengakses simpul ? [Y]a/[T]idak : ');
lagi:=upcase(readkey);clrscr;
until lagi <>'Y';

end;  {* akhir procedure menu sisip *}

{***** Procedure Penyajian Menu Hapus Pohon Splay *****)

Procedure menu_hapus(var pohon:tree);

begin
    clrscr;
    gotoxy(19,1);textcolor(hijau_t);
    writeln('<<< PROSES PENGHAPUSAN SIMPUL POHON SPLAY >>>');
    gotoxy(23,2);writeln('( Dengan preorder,inorder,postorder )');
    gotoxy(23,3);writeln('=====');
    writeln;writeln;textcolor(cyan_t);
    writeln('1. Pembacaan Simpul Pohon Biner Sebelum Penghapusan');
    textcolor(biru_t);
    write('    a. Preorder   : ');r_pre(akar);readln;
    write('    b. Inorder     : ');r_in(akar);readln;
    write('    c. Postorder    : ');r_pos(akar);readln;writeln;
    textcolor(cyan_t);writeln('2. Proses Penghapusan Simpul Pohon
    Splay');

    write('    a. Masukkan simpul yang akan dihapus : ');
    while not eoln do
        begin
            read(huruf);
            hapus_kunci(akar,huruf);
        end;
    readln;

```

```

textcolor(cyan_t);writeln;
writeln('3. Pembacaan Simpul Pohon Splay Sesudah Penghapusan');
textcolor(biru_t);
write('    a. Preorder  : ');r_pre(akar);readln;
write('    b. Inorder   : ');r_in(akar);readln;
write('    c. Postorder  : ');r_pos(akar);readln;writeln;
lagi:='Y';
repeat
textcolor(cyan_t);
writeln('4. Proses Pemasukan Data yang Diakses');
write('    a. Masukan simpul yang diakses : ');
read(huruf);cari_simpul(ada,akar,huruf);
if not(ada) then
begin
writeln;textcolor(merah_t+kedip);writeln(' ':18,bel,ps2);
textcolor(biru_t);
writeln('        Simpul "',huruf,'"tidak ada dalam pohon');
writeln('        << Pilihan Proses >>');
writeln('        [1]. Menyisipkan simpul');
writeln('        [0]. Keluar dari proses');
pilihan:=9;
while (pilihan<0) or (pilihan>3) do
begin
textcolor(merah_t);
write(' ':7,'Tentukan pilihan anda (0 - 1) : ');
readln(pilihan);writeln;
end;
case pilihan of
1 : begin
textcolor(putih_t);
write('    b. Masukan simpul yang akan disisipkan : ');
while not eoln do
begin
read(huruf);sisip_simpul(akar,huruf);
end
end;
0 : begin
writeln('    Tekan <enter> untuk keluar dari proses ini...!');
readln;
exit;
end
end
end;
treesplay(akar,huruf);
textcolor(cyan_t);writeln;
writeln('5. Proses Pembacaan Simpul Setelah Operasi Splaying');
textcolor(biru_t);
write('    a. Preorder  : ');r_pre(akar);readln;writeln;
write('    b. Inorder   : ');r_in(akar);readln;
write('    c. Postorder  : ');r_pos(akar);readln;
writeln;
textcolor(putih_t);
write(' ':15,'Apakah akan mengakses simpul ? [Y]a/[T]idak : ');
lagi:=upcase(readkey);clrscr;
until lagi <>'Y';
end;

```

```

{*****
 *   Bagian Informasi Penggunaan Operasi Splaying   *
*****}

{***** Procedure Aturan Pembacaan Kunjungan *****}

procedure Kunjungan;

begin
  clrscr;textbackground(biru);clrscr;
  textcolor(hijau_t);writeln;writeln;
  writeln(' ':15,'PENJELASAN PROSES KUNJUNGAN POHON BINER');
  writeln(' ':15,' ( Preorder, Inorder, dan Postorder ) ');
  writeln(' ':15,'-----');
  textcolor(putih_t);writeln;
  writeln(' 1. Urutan Proses kunjungan Preorder ');
  writeln('   - Cetak isi simpul yang dikunjungi');
  writeln('   - Kunjungi cabang kiri');
  writeln('   - Kunjungi cabang kanan');readln;
  writeln(' 2. Urutan proses kunjungan Inorder ');
  writeln('   - Kunjungi cabang kiri');
  writeln('   - Cetak isi simpul yang dikunjungi');
  writeln('   - Kunjungi cabang kanan');readln;
  writeln(' 3. Urutan proses kunjungan Postorder');
  writeln('   - Kunjungi cabang kiri');
  writeln('   - Kunjungi cabang kanan');
  writeln('   - Cetak isi simpul yang dikunjungi');readln;
end;  (* akhir procedure kunjungan *)

{***** Procedure Informasi Menu Utama *****}

procedure Komentar01;

begin
  clrscr;textbackground(biru);clrscr;
  gotoxy(33,1);textcolor(merah+kedip);textbackground(putih_t);
  Beep(500,100);writeln(ps3);
  Writeln('( Informasi Menu Utama )');writeln;writeln;
  textcolor(putih_t);textbackground(biru);
  writeln(' [1]. Proses Pohon Splay ');
  writeln('     Menyajikan proses pembentukan pohon splay');
  writeln('     dengan pemasukan untaian karakter');
  writeln;
  writeln(' [2]. Proses Penyisipan Pohon Splay');
  writeln('     Menyajikan proses penyisipan simpul pohon splay');
  writeln;
  writeln(' [3]. Proses Penghapusan Simpul');
  writeln('     Menyajikan proses penghapusan simpul pohon splay');
  writeln;
  writeln(' [0]. Selesai');
  writeln('     Kembali ke DOS');
  writeln;writeln;textcolor(putih_t+kedip);
  write(' Tekan Sebarang Tombol Untuk Kembali Menu Petunjuk...! ');
  textbackground(hitam);Beep(500,100);
  p:=readkey; clrscr;
end;  (* akhir procedure komentar01 *)

```



```

{***** Procedure Informasi Menu Splay *****}

Procedure Komentar02;

begin
  clrscr;textbackground(biru);clrscr;
  gotoxy(33,1);textcolor(merah+kedip);textbackground(putih_t);
  beep(500,100);writeln(ps3);
  writeln('( Informasi Menu Proses Splaying )');
  textcolor(putih_t);textbackground(biru);
  writeln(' 1. Untaian simpul yang dapat diterima berupa karakter,');
  writeln('      baik huruf besar maupun kecil semua atau kedua-
      duanya');
  writeln;
  writeln(' 2. Aturan pembentukan pohon biner :');
  writeln('      Karakter pertama dari untaian simpul akan menjadi');
  writeln('      akar pohon biner. Simpul yang lebih besar urutannya');
  writeln('      dari simpul ayah akan ditempatkan sebagai cabang ');
  writeln('      kanan dan simpul yang lebih kecil urutannya sebagai');
  writeln('      cabang kiri');writeln;
  writeln(' 3. Jika simpul akar diakses kembali dalam proses
      operasi');
  writeln('      splaying maka akan didapatkan pohon splay yang sama ');
  writeln('      dengan pohon splay sebelumnya');writeln;
  writeln(' 4. Karakter dengan huruf besar urutannya lebih kecil ');
  writeln('      dibandingkan dengan huruf kecil (berdasarkan ASCII)');
  writeln;
  writeln(' 5. Pembacaan simpul pohon sebelum maupun sesudah adalah');
  writeln('      dengan operasi kunjungan preorder, inorder, dan
      postorder');
  writeln;textcolor(putih_t+kedip);
  write('      Tekan < Enter > untuk penjelasan no. 5 ....!');
  readln;Kunjungan;textcolor(putih_t+kedip);
  write('      Tekan Sebarang Tombol Untuk Kembali Menu Petunjuk...! ');
  textbackground(hitam);Beep(500,100);
  p:=readkey;
  clrscr;
end; (* akhir procedure komentar02 *)

{***** Procedure Informasi Menu Sisip simpul *****}

Procedure Komentar03;

begin
  clrscr;textbackground(biru);clrscr;
  gotoxy(33,1);textcolor(merah+kedip);textbackground(putih_t);
  Beep(500,100);writeln(ps3);
  Writeln('( Informasi Menu Proses Sisip Simpul )');writeln;writeln
  textcolor(putih_t);textbackground(biru);
  writeln(' 1. Simpul yang disisipkan minimal satu karakter, ');
  writeln('      baik huruf besar maupun huruf kecil');writeln;
  writeln(' 2. Jika terjadi duplikasi, harap mengganti simpul yang');
  writeln('      akan disisipkan atau keluar dari proses ini');writeln;
  writeln(' 3. Jika simpul akar diakses kembali dalam proses
      operasi');
  writeln('      splaying maka akan didapatkan pohon splay yang sama ');

```

```

writeln('    dengan pohon splay sebelumnya');writeln;
writeln(' 4. Pembacaan simpul pohon sebelum maupun sesudah adalah');
writeln('    dengan operasi kunjungan preorder, inorder, dan
        postorder');
writeln;writeln;textcolor(putih_t+kedip);
writeln;write('    Tekan < Enter > untuk penjelasan no.4 ...!');
readln;kunjungan;writeln;
textcolor(putih_t+kedip);
write('    Tekan Sebarang Tombol Untuk Kembali Menu Petunjuk...! ');
textbackground(hitam);Beep(500,100);
p:=readkey;
clrscr;

end;  (* akhir procedure komentar03 *)

{***** Procedure Informasi Menu Hapus Simpul *****)

procedure Komentar04;

Begin
  clrscr;textbackground(biru);clrscr;
  gotoxy(33,1);textcolor(merah_t+kedip);textbackground(putih);
  Beep(500,100);writeln(ps3);
  Writeln('( Informasi Menu Proses Hapus Simpul )');writeln;writeln;
  textcolor(putih_t);textbackground(biru);
  writeln(' 1. Simpul yang dihapus minimal satu karakter, ');
  writeln('    baik huruf besar maupun huruf kecil');
  writeln(' 2. Jika tidak ada simpulnya, harap mengganti simpul');
  writeln('    yang akan dihapus atau keluar dari proses ini');
  writeln;
  writeln(' 3. Jika simpul akar diakses kembali dalam proses
    operasi');
  writeln('    splaying maka akan didapatkan pohon splay yang sama ');
  writeln('    dengan pohon splay sebelumnya');writeln;
  writeln(' 4. Pembacaan simpul pohon sebelum maupun sesudah');
  writeln('    dengan operasi kunjungan preorder, inorder, dan
    postorder');
  writeln;writeln;textcolor(putih_t+kedip);
  write('    Tekan < Enter > untuk penjelasan no.4 ...! ');
  readln;kunjungan;writeln;
  textcolor(putih_t+kedip);
  write('    Tekan Sebarang Tombol Untuk Kembali Menu Petunjuk...! ');
  textbackground(hitam);Beep(500,100);
  p:=readkey;
  clrscr;

end;  (* akhir procedure komentar04 *)

```



```

{*****
*      Bagian Proses Menu Utama Operasi pohon Splay      *
*****}

{***** Procedure Penyajian Menu Utama Pohon Splay *****}

Procedure Menu_utama(var pohon:tree);

{***** Procedure Penyajian Menu Informasi Pohon Splay *****}

Procedure Komentar;

begin
repeat
  clrscr;textbackground(hitam);clrscr;
  textbackground(biru);textcolor(putih_t);
  gotoxy(20,5) gotoxy(20,6) ;
  gotoxy(20,7) ;writeln('°   INFORMASI PENGGUNAAN PROGRAM SPLAY   °');
  gotoxy(20,8) ;writeln('°');
  gotoxy(20,9) ;writeln('° [1]. Informasi Menu Utama °');
  gotoxy(20,10);writeln('° [2]. Informasi Menu Proses Splaying °');
  gotoxy(20,11);writeln('° [3]. Informasi Menu Proses Sisip Simpul °');
  gotoxy(20,12);writeln('° [4]. Informasi Menu Proses Hapus Simpul °');
  gotoxy(20,13);writeln('° [0]. Kembali Ke Menu Utama °');
  gotoxy(20,14);writeln('°');
  gotoxy(20,20);textcolor(putih_t);textbackground(biru);
  write(' Tentukan pilihan anda ( 0 .. 4 ) : ');
  repeat
    {$I-}
    y:=20;
    readln(pilihan);textbackground(hitam);
    Beep(500,100);
    {$I+}
    hasilIO:=IOresult;
    if hasilIO <> 0 then
      textcolor(merah_t+kedip);writeln;
      writeln(' ':30,bel,ps2);textcolor(magenta);
      writeln(' ':20,'Pilihan harus bilangan bulat (0..4), ulangi...!');
      gotoxy(20,y+5);textcolor(putih_t);textbackground(biru);
      write(' Tentukan pilihan anda ( 0 .. 4 ) : ');
      textbackground(hitam);
    until hasilIO = 0;
  clrscr;
  Selesai:=False;
  case pilihan of
    1 : komentar01;
    2 : komentar02;
    3 : komentar03;
    4 : komentar04;
    0 : Selesai:=True
  else

```

```

Begin
  clrscr;Textbackground(cyan_t);clrscr;
  writeln;writeln;gotoxy(30,12);
  textcolor(putih_t+kedip);textbackground(merah);
  Beep(500,100);writeln(bel,' SALAH PILIH,EUY.....! ');
  textbackground(cyan_t);textcolor(putih_t);
  gotoxy(25,14);writeln('Pilihan Menu Hanya ( 0 .. 4 )...!');
  gotoxy(25,16);write(psl);
  readln;
end;
end;  (* akhir case *)
until Selesai;
menu_utama(akar);
end;  (* akhir procedure komentar *)

{***** Procedure Penyajian Menu Tambahan Pohon Splay *****)

Procedure Menu_tambahan(var pohon:tree);

begin
repeat
clrscr;textbackground(hitam);clrscr;
textbackground(biru);textcolor(putih_t);
gotoxy(20,7);
gotoxy(20,8); writeln('°          <<<  PERINGATAN  >>>          °');
gotoxy(20,9); writeln('°          °          °');
gotoxy(20,10);writeln('°          [1]. Kembali Ke Menu Utama          °');
gotoxy(20,11);writeln('°          [0]. Keluar Dari Program Splay Tree °');
gotoxy(20,12);writeln('°          °          °');
Musik;
gotoxy(20,16);textcolor(putih_t);textbackground(biru);
writeln(' Apakah Anda Ingin Keluar dari Proses ini ? ');
textbackground(hitam);
gotoxy(20,20);textcolor(putih_t);textbackground(biru);
write(' Tentukan Pilihan Anda ( 0 .. 1 ) : ');
repeat
{$I-}
y:=20;
readln(pilihan);textbackground(hitam);
Beep(500,100);
{$I+}
hasilIO:=IOresult;
if hasilIO <> 0 then
textcolor(merah_t+kedip);writeln;
writeln(' ':30,bel,ps2);textcolor(magenta);
writeln(' ':20,'Pilihan harus bilangan bulat (0..1),
ulangi....!');
gotoxy(20,y+5);textcolor(putih_t);textbackground(biru);
write(' Tentukan Pilihan Anda ? ( 0 .. 1 ) : ');
textbackground(hitam);
until hasilIO = 0;
clrscr;
Selesai:=false;
case pilihan of
1 : Menu_utama(akar);
0 : Selesai:=True;

```

```

else
  Begin
    clrscr;Textbackground(cyan_t);clrscr;
    writeln;writeln;gotoxy(30,12);
    textcolor(putih_t+kedip);textbackground(merah_t);
    Beep(500,100);writeln(bel,' SALAH PILIH, EUY.....! ');
    textbackground(cyan_t);textcolor(putih_t);
    gotoxy(25,14);writeln('Pilihan Menu Hanya ( 0 .. 1 )...!');
    gotoxy(25,16);write(ps1);
    readln;
  end
end;
until selesai;
exit;

end; (* akhir procedure menu_tambahan *)

{***** Procedure Penyajian Proses Menu Utama Splay *****}

begin
repeat
clrscr;textbackground(hitam);clrscr;
gotoxy(30,2);textcolor(hijau_t);
writeln('PROSES PENYAJIAN POHON SPLAY!');
writeln(' ':25,'DENGAN OPERASI KUNJUNGAN POHON BINER');
writeln(' ':25,'( PreOrder, InOrder, dan PostOrder )');
textbackground(biru);textcolor(putih_t);
gotoxy(20,7);
gotoxy(20,8); writeln('° <<< MENU UTAMA >>> °');
gotoxy(20,9); writeln('° °');
gotoxy(20,10);writeln('° [1]. Informasi Menu °');
gotoxy(20,11);writeln('° [2]. Proses Pohon Splay °');
gotoxy(20,12);writeln('° [3]. Proses Penyisipan Simpul °');
gotoxy(20,13);writeln('° [4]. Proses Penghapusan Simpul °');
gotoxy(20,14);writeln('° [0]. Selesai °');
gotoxy(20,15);writeln('° °');
gotoxy(20,20);textcolor(putih_t);textbackground(biru);
write(' Tentukan pilihan anda ( 0 .. 4 ) : ');
repeat
  {$I-}
  y:=20;
  readln(pilihan);textbackground(hitam);
  Beep(500,100);
  {$I+}
  hasilIO:=IOresult;
  if hasilIO <> 0 then
    textcolor(merah_t+kedip);writeln;
    writeln(' ':30,bel,ps2);textcolor(magenta);
    writeln(' ':20,'Pilihan harus bilangan bulat (0..4),
ulangi.....!');
    gotoxy(20,y+5);textcolor(putih_t);textbackground(biru);
    write(' Tentukan pilihan anda ( 0 .. 4 ) : ');
    textbackground(hitam);
  until hasilIO = 0;
  clrscr;
Selesai:=False;

```

```

case pilihan of
  1 : Komentor;
  2 : menu_splay(akar);
  3 : menu_sisip(akar);
  4 : menu_hapus(akar);
  0 : Selesai:=True
else
  Begin
    clrscr;Textbackground(cyan_t);clrscr;
    writeln;writeln;gotoxy(30,12);
    textcolor(putih_t+kedip);textbackground(merah_t);
    Beep(500,100);writeln(bel,' SALAH PILIH, EUY.....! ');
    textbackground(cyan_t);textcolor(putih_t);
    gotoxy(25,14);writeln('Pilihan Menu Hanya ( 0 .. 4 )...!');
    gotoxy(25,16);write(ps1);
    readln;
  end
end;  (* akhir case *)
until Selesai;
menu_tambahan(akar)
end;  (* akhir procedure menu utama *)
[***** Procedure Penyajian Tampilan Program *****]

procedure Tampilan;

var p : char;

begin
  clrscr;textbackground(biru_t); clrscr;
  textcolor(cyan_t);gotoxy(36,9);writeln('Nama Program');
  textcolor(hijau_t);gotoxy(21,11);
  writeln('PROSES PENYAJIAN POHON SPLAY DENGAN OPERASI');
  writeln(' ':19,'KUNJUNGAN POHON BINER (Binary Tree Traversal)');
  writeln(' ':23,'< Preorder, Inorder, dan Postorder >');
  textcolor(cyan_t); gotoxy(40,16);writeln('Oleh');
  textcolor(hijau_t);gotoxy(24,19);
  writeln(' Nama      : APIP ICHSAN CAHYANA');
  writeln(' ':25,'Nim      : J 101 95 1179');
  writeln(' ':25,'Jurusan  : Matematika ');
  gotoxy(20,24);
  writeln('Fakultas Matematika dan Ilmu Pengetahuan Alam');
  writeln(' ':31,'Universitas Diponegoro');
  writeln(' ':38,'Semarang');writeln(' ':40,'2000');
  gotoxy(30,30);textcolor(cyan_t+kedip);writeln;writeln;writeln;
  musik;write(' Tekan Sebarang Tombol Untuk Memulai.....! ');
  p:=readkey;
  clrscr;
end;  (* akhir procedure tampilan *)

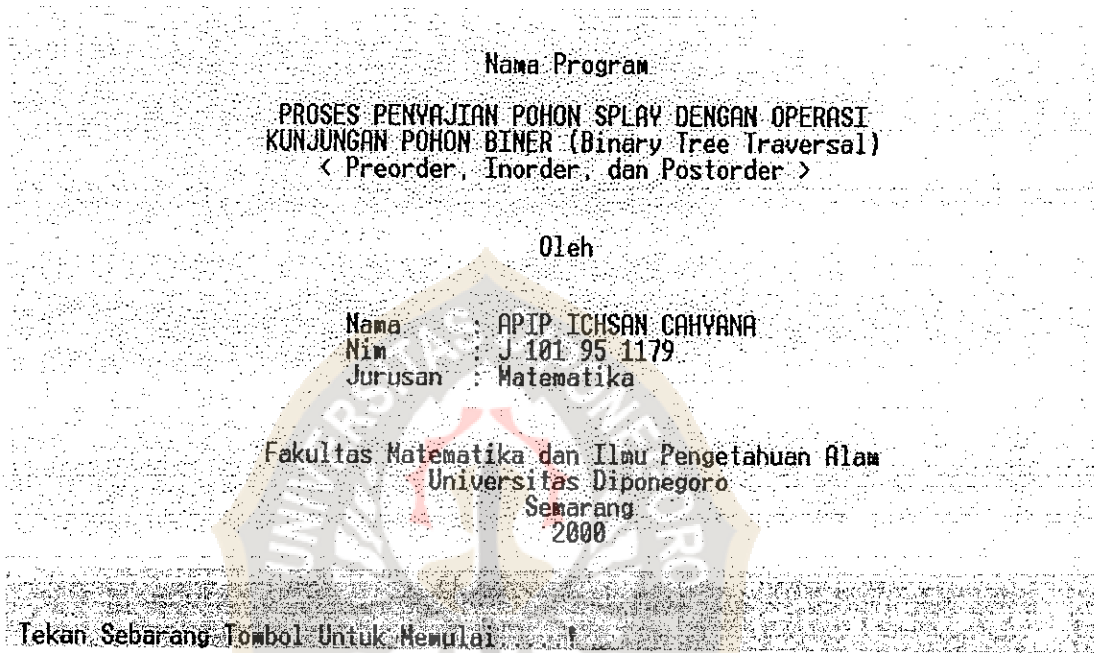
(* program utama *)

begin
  clrscr;
  tampilan;
  menu_utama(akar);
end.  (* akhir Program splaytree *)

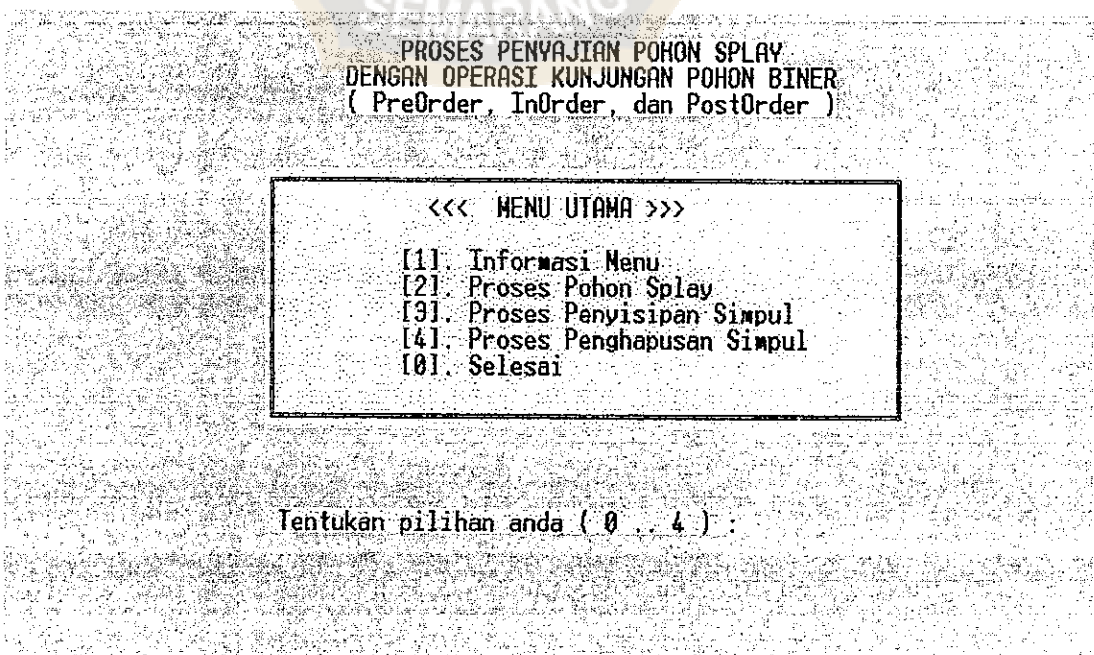
```

## LAMPIRAN OUTPUT PROGRAM

( Beberapa Contoh Tampilan Output Program Pohon Splay )

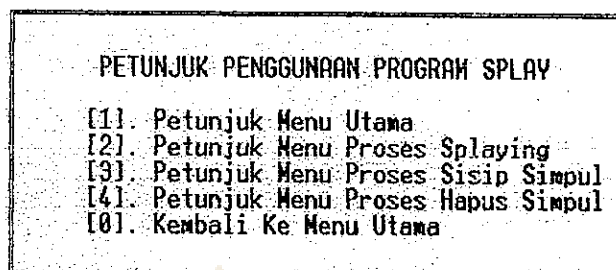


Gambar 1. Tampilan Cover Program



Gambar 2. Tampilan Menu Utama Program





Tentukan pilihan anda ( 0 .. 4 ) :

Gambar 3. Tampilan Petunjuk Penggunaan Pohon Splay

\*\* K E T E R A N G A N \*\*

- [1]. Proses Pohon Splay  
Menyajikan proses pembentukan pohon splay dengan pemasukan untaian karakter yang berupa huruf atau angka
- [2]. Proses Penyisipan Pohon Splay  
Menyajikan proses penyisipan simpul pohon splay
- [3]. Proses Penghapusan Simpul  
Menyajikan proses penghapusan simpul pohon splay
- [0]. Selesai  
Kembali ke DOS

Tekan Sebarang Tombol Untuk Kembali Menu Petunjuk...!

Gambar 4. Tampilan Petunjuk Menu Utama

**\*\* KETERANGAN \*\***

1. Untaian simpul yang dapat diterima berupa karakter, baik huruf besar maupun kecil semua atau kedua-duanya. Untaian angka dapat pula dimasukkan sebagai karakter
2. Aturan pembentukan pohon biner :  
Karakter pertama dari untaian simpul akan menjadi akar pohon biner. Simpul yang lebih besar urutannya dari simpul ayah akan ditempatkan sebagai cabang kanan dan simpul yang lebih kecil urutannya sebagai cabang kiri
3. Jika simpul akar diakses kembali dalam proses operasi splaying maka akan didapatkan pohon splay yang sama dengan pohon splay sebelumnya
4. Karakter dengan huruf besar urutannya lebih kecil dibandingkan dengan huruf kecil (berdasarkan ASCII)
5. Pembacaan simpul pohon sebelum maupun sesudah adalah dengan operasi kunjungan preorder, inorder, dan postorder

Tekan < Enter > untuk penjelasan no. 5 ... !

Gambar 5. Tampilan Petunjuk Menu Proses Pohon Splay

**\*\* KETERANGAN \*\***

1. Simpul yang disisipkan minimal satu karakter, baik huruf besar maupun huruf kecil, demikian pula berlaku untuk penyisipan angka
2. Jika terjadi duplikasi, harap mengganti simpul yang akan disisipkan atau keluar dari proses ini
3. Jika simpul akar diakses kembali dalam proses operasi splaying maka akan didapatkan pohon splay yang sama dengan pohon splay sebelumnya
4. Pembacaan simpul pohon sebelum maupun sesudah adalah dengan operasi kunjungan preorder, inorder, dan postorder

Tekan < Enter > untuk penjelasan no. 4 ... !

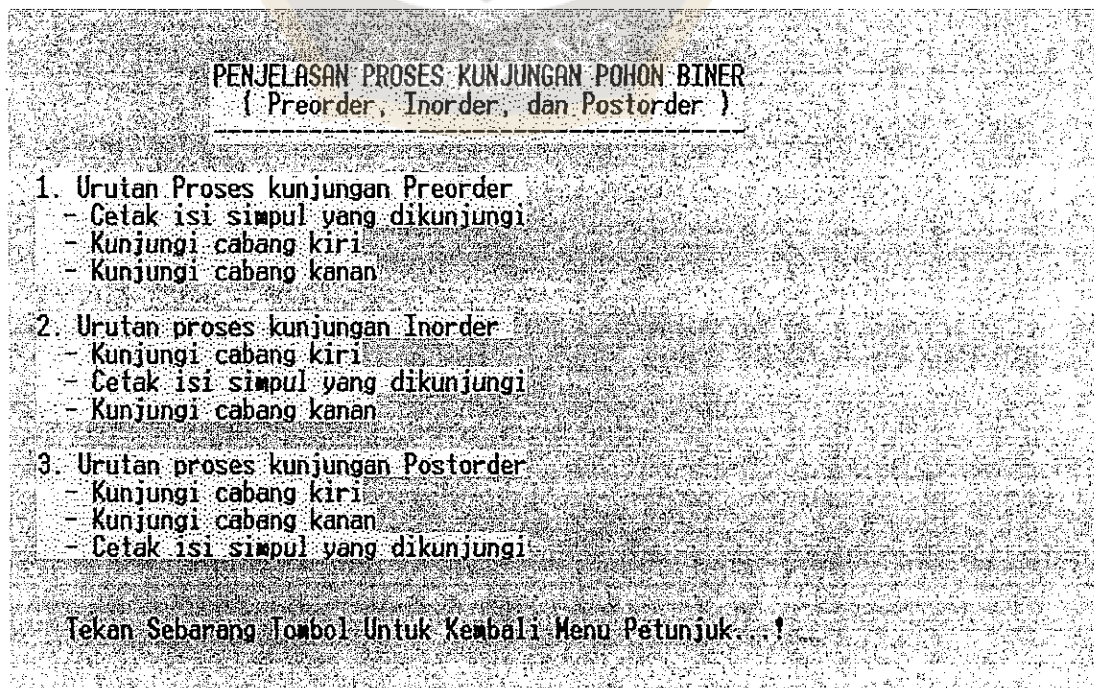
Gambar 6. Tampilan Petunjuk Menu Penyisipan Simpul

**KETERANGAN**

1. Simpul yang dihapus minimal satu karakter, baik huruf besar maupun huruf kecil, demikian pula berlaku untuk penghapusan angka
2. Jika tidak ada simpulnya, harap mengganti simpul yang akan dihapus atau keluar dari proses ini
3. Jika simpul akar diakses kembali dalam proses operasi splaying maka akan didapatkan pohon splay yang sama dengan pohon splay sebelumnya
4. Pembacaan simpul pohon sebelum maupun sesudah dengan operasi kunjungan preorder, inorder, dan postorder

Tekan < Enter > untuk penjelasan no.4 ...!

Gambar 7. Tampilan Petunjuk Menu Penghapusan Simpul



Gambar 8. Tampilan Keterangan Kunjungan Pohon Biner



**Proses Penyajian Pohon Splay  
Dengan Pemasukan Berupa Untaian Simpul Huruf**

<<< PROSES PENYAJIAN POHON SPLAY >>>  
( Dengan preorder, inorder, postorder )

1. Proses pemasukan data  
Masukkan untaian simpul : hkacbdjl
2. Proses Pembacaan Simpul Pohon Biner Sebelum Splaying
  - a. Preorder : h a c b d k j l
  - b. Inorder : a b c d h j k l
  - c. Postorder : b d c a j l k h
3. Proses pemasukan Data yang diakses
  - a. Masukan simpul yang diakses : b
  - b. Simpul "b" terdapat dalam pohon.
  - c. Target ditemukan dan menjadi akar.
4. Proses pembacaan simpul setelah operasi splaying
  - a. Preorder : b a h c d k j l
  - b. Inorder : a b c d h j k l
  - c. Postorder : a d c j l k h b

Apakah akan mengakses simpul ? [Y]/a/[T]idak : \_

Gambar 9. Tampilan Proses Penyajian Pohon Splay

<<< PROSES PENYISIPAN SIMPUL POHON SPLAY >>>  
( Dengan preorder, inorder, postorder )

1. Pembacaan Simpul Pohon Splay Sebelum Penyisipan
  - a. Preorder : b a h c d k j l
  - b. Inorder : a b c d h j k l
  - c. Postorder : a d c j l k h b
2. Proses Penyisipan Simpul Pohon Splay
  - a. Masukkan simpul yang akan disisipkan : f
3. Pembacaan Simpul Pohon Splay Sesudah Penyisipan
  - a. Preorder : b a h c d f k j l
  - b. Inorder : a b c d f h j k l
  - c. Postorder : a f d c j l k h b
4. Proses Pemasukan Data yang Diakses
  - a. Masukan simpul yang diakses : f
  - b. Simpul "f" terdapat dalam pohon.
  - c. Target ditemukan dan menjadi akar.
5. Proses Pembacaan Simpul Setelah Operasi Splaying
  - a. Preorder : f b a d c h k j l
  - b. Inorder : a b c d f h j k l
  - c. Postorder : a c d b j l k h f

Gambar 10. Tampilan Proses Penyisipan Simpul Pohon Splay

**<<< PROSES PENGHAPUSAN SIMPUL POHON SPLAY >>>**  
**( Dengan preorder, inorder, postorder )**  
 =====

1. Pembacaan Simpul Pohon Biner Sebelum Penghapusan
  - a. Preorder : f b a d c h k j l
  - b. Inorder : a b c d f h j k l
  - c. Postorder : a c d b j l k h f
2. Proses Penghapusan Simpul Pohon Splay
  - a. Masukkan simpul yang akan dihapus : f  
Simpul "f" telah dihapus dari pohon
3. Pembacaan Simpul Pohon Splay Sesudah Penghapusan
  - a. Preorder : h b a d c k j l
  - b. Inorder : a b c d h j k l
  - c. Postorder : a c d b j l k h
4. Proses Pemasukan Data yang Diakses
  - a. Masukan simpul yang diakses : b

Gambar 11a. Tampilan Proses Penghapusan Simpul Pohon Splay

1. Pembacaan Simpul Pohon Biner Sebelum Penghapusan
  - a. Preorder : f b a d c h k j l
  - b. Inorder : a b c d f h j k l
  - c. Postorder : a c d b j l k h f
2. Proses Penghapusan Simpul Pohon Splay
  - a. Masukkan simpul yang akan dihapus : f  
Simpul "f" telah dihapus dari pohon
3. Pembacaan Simpul Pohon Splay Sesudah Penghapusan
  - a. Preorder : h b a d c k j l
  - b. Inorder : a b c d h j k l
  - c. Postorder : a c d b j l k h
4. Proses Pemasukan Data yang Diakses
  - a. Masukan simpul yang diakses : b
  - b. Simpul "b" terdapat dalam pohon.
  - c. Target ditemukan dan menjadi akar.
5. Proses Pembacaan Simpul Setelah Operasi Splaying
  - a. Preorder : b a h d c k j l
  - b. Inorder : a b c d h j k l
  - c. Postorder : a c d j l k h b

Apakah akan mengakses simpul ? [Y]/a/[T]idak :

Gambar 11b. Tampilan Proses Penghapusan Simpul Pohon Splay

**Proses Penyajian Pohon Splay  
Dengan Pemasukan Berupa Untaian Simpul Angka**

<<< PROSES PENYAJIAN POHON SPLAY >>>  
( Dengan preorder, inorder, postorder )

1. Proses pemasukan data  
Masukkan untaian simpul : 198276945
2. Proses Pembacaan Simpul Pohon Biner Sebelum Splaying
  - a. Preorder : 1 9 8 2 7 6 3 4 5
  - b. Inorder : 1 2 3 4 5 6 7 8 9
  - c. Postorder : 5 4 3 6 7 2 8 9 1
3. Proses pemasukan Data yang diakses
  - a. Masukan simpul yang diakses : 5
  - b. Simpul "5" terdapat dalam pohon.
  - c. Target ditemukan dan menjadi akar.
4. Proses pembacaan simpul setelah operasi splaying
  - a. Preorder : 5 1 2 4 3 9 8 6 7
  - b. Inorder : 1 2 3 4 5 6 7 8 9
  - c. Postorder : 3 4 2 1 7 6 8 9 5

Apakah akan mengakses simpul ? IYA/TIDAK :

Gambar 12. Tampilan Proses Penyajian Pohon Splay

<<< PROSES PENYISIPAN SIMPUL POHON SPLAY >>>  
( Dengan preorder, inorder, postorder )

1. Pembacaan Simpul Pohon Splay Sebelum Penyisipan
  - a. Preorder : 5 1 2 4 3 9 8 6 7
  - b. Inorder : 1 2 3 4 5 6 7 8 9
  - c. Postorder : 3 4 2 1 7 6 8 9 5
2. Proses Penyisipan Simpul Pohon Splay
  - a. Masukan simpul yang akan disisipkan : 0
3. Pembacaan Simpul Pohon Splay Sesudah Penyisipan
  - a. Preorder : 5 1 0 2 4 3 9 8 6 7
  - b. Inorder : 0 1 2 3 4 5 6 7 8 9
  - c. Postorder : 0 3 4 2 1 7 6 8 9 5
4. Proses Pemasukan Data yang Diakses
  - a. Masukan simpul yang diakses : 0
  - b. Simpul "0" terdapat dalam pohon.
  - c. Target ditemukan dan menjadi akar.
5. Proses Pembacaan Simpul Setelah Operasi Splaying
  - a. Preorder : 0 1 5 2 4 3 9 8 6 7
  - b. Inorder : 0 1 2 3 4 5 6 7 8 9
  - c. Postorder : 3 4 2 7 6 8 9 5 1 0

Gambar 13. Tampilan Proses Penyisipan Simpul Pohon Splay

<<< PROSES PENGHAPUSAN SIMPUL POHON SPLAY >>>  
( Dengan preorder, inorder, postorder )

1. Pembacaan Simpul Pohon Biner Sebelum Penghapusan
  - a. Preorder : 0 1 5 2 4 3 9 8 6 7
  - b. Inorder : 0 1 2 3 4 5 6 7 8 9
  - c. Postorder : 3 4 2 7 6 8 9 5 1 0
2. Proses Penghapusan Simpul Pohon Splay
  - a. Masukkan simpul yang akan dihapus : 0  
Simpul "0" telah dihapus dari pohon
3. Pembacaan Simpul Pohon Splay Sesudah Penghapusan
  - a. Preorder : 1 5 2 4 3 9 8 6 7
  - b. Inorder : 1 2 3 4 5 6 7 8 9
  - c. Postorder : 3 4 2 7 6 8 9 5 1
4. Proses Pemasukan Data yang Diakses
  - a. Masukkan simpul yang diakses : 5

Gambar 14a.. Tampilan Proses Penghapusan Simpul Pohon Splay

1. Pembacaan Simpul Pohon Biner Sebelum Penghapusan
  - a. Preorder : 0 1 5 2 4 3 9 8 6 7
  - b. Inorder : 0 1 2 3 4 5 6 7 8 9
  - c. Postorder : 3 4 2 7 6 8 9 5 1 0
2. Proses Penghapusan Simpul Pohon Splay
  - a. Masukkan simpul yang akan dihapus : 0  
Simpul "0" telah dihapus dari pohon
3. Pembacaan Simpul Pohon Splay Sesudah Penghapusan
  - a. Preorder : 1 5 2 4 3 9 8 6 7
  - b. Inorder : 1 2 3 4 5 6 7 8 9
  - c. Postorder : 3 4 2 7 6 8 9 5 1
4. Proses Pemasukan Data yang Diakses
  - a. Masukkan simpul yang diakses : 5
  - b. Simpul "5" terdapat dalam pohon.
  - c. Target ditemukan dan menjadi akar.
5. Proses Pembacaan Simpul Setelah Operasi Splaying
  - a. Preorder : 5 1 2 4 3 9 8 6 7
  - b. Inorder : 1 2 3 4 5 6 7 8 9
  - c. Postorder : 3 4 2 1 7 6 8 9 5

Apakah akan mengakses simpul ? [Y]/[N]idak :

Gambar 14b.. Tampilan Proses Penghapusan Simpul Pohon Splay