

LAMPIRAN KODE PROGRAM

```
{ $M 65520,0,650000 }
{ *****
  PROGRAM UNTUK MENYELESAIKAN MASALAH PERKALIAN RANGKAIAN MATRIKS
  DISUSUN OLEH : HELMIE ARIF WIBAWA (J 2A0 96 027)
  ***** }
program MATRIKS;
uses crt,dos;
type Pmat = ^mat;
mat = record
  baris : byte;
  kolom : byte;
  nilai : integer;
  kanan,
  bawah : Pmat
end;
matrixs = record
  baris, kolom : byte;
  nilai : array[1..50,1..50] of integer;
end;
DataMatrik = record
  byk : byte;
  pl,p2 : bitel;
  matrix : array [1..10] of matrixs;
end;
RecordMenu = record
  PosisiMenu : array[1..6] of byte;
  KarakterMenu : array[1..6] of char;
  PosisiKarakterMenu : array[1..6] of byte;
  IsiMenu : array[1..6] of string23;
end;
RecordMenuEdit = record
  PosisiMenu : array[1..4] of byte;
  KarakterMenu : array[1..4] of char;
  PosisiKarakterMenu : array[1..4] of byte;
  IsiMenu : array[1..4] of string23;
end;
RecordMenuProses = record
  PosisiMenu : array[1..3] of byte;
  KarakterMenu : array[1..3] of char;
  PosisiKarakterMenu : array[1..3] of byte;
  IsiMenu : array[1..3] of string23;
end;
pixel = record
  karakter : char;
  atribut : byte;
end;
matrik = array [1..10] of Pmat;
tabel = array [1..2] of Pmat;
bite = array[1..50,1..50] of byte;
bitel = array[0..50] of byte;
string3 = string[3];
string23 = string[23];
mskalar = array[1..50,1..50] of longint;
bufferlayar = array[1..25,1..80] of pixel;
```

```

const DataMenu : RecordMenu =
  (PosisiMenu : (9,10,11,12,13,14);
   KarakterMenu : 'IMBSPe';
   PosisiKarakterMenu : (28,28,28,28,28,29);
   IsiMenu : (' Informasi Menu      ', ' Masukan Data      ',
             ' Buka File          ', ' Simpan ke File    ',
             ' Proses                ', ' Keluar            '));

DataMenuProses : RecordMenuProses =
  (PosisiMenu : (9,10,11);
   KarakterMenu : 'TDe';
   PosisiKarakterMenu : (28,28,29);
   IsiMenu : (' Tanpa Algoritma DP ', ' Dengan Algoritma DP ',
             ' Ke Menu Utama          '));

var c,mat_kali : Pmat;
    h          : matrik;
    jawab      : char;
    bar,kol,i,ii,jj,n : byte;
    nla        : integer;
    r,r1,r2    : bitel;
    tt         : bite;
    byk        : byte;
    benar      : boolean;
    Layar : BufferLayar absolute $B800:$0000;
    LayarMenuUtama,LayarMasukan,LayarMenuProses : BufferLayar;
    h1,h2,m1,m2,s1,s2,hund1,hund2 :word;

{prosedur untuk membuat simpul baru dari matriks }
procedure buat_simpul(var Baru : Pmat; b,k : byte; n : integer);
begin
  new(Baru);
  with Baru^ do
    begin
      baris := b;
      kolom := k;
      nilai := n;
      kanan := nil;
      bawah := nil;
    end
  end;
end;

{prosedur untuk mencari informasi apakah suatu baris terdapat dalam matriks}
procedure ada_baris(var nb : Pmat; var ada : boolean; awal : Pmat; b : byte);
begin
  ada := false;
  nb := awal^.bawah;
  repeat
    if nb^.baris = b then
      ada := true
    else
      nb := nb^.bawah
    until ada or (nb = awal) or (nb^.baris > b)
  end;
end;

```

```

{prosedur untuk mencari informasi apakah suatu kolom terdapat dalam matriks}
procedure ada_kolom(var nk : Pmat; var ada : boolean; awal_ : Pmat; k : byte);
begin
  ada := false;
  nk := awal^.kanan;
  repeat
    if nk^.kolom = k then
      ada := true
    else
      nk := nk^.kanan
  until ada or (nk = awal) or (nk^.kolom > k)
end;

```

```

{prosedur untuk memberikan informasi tentang langkah yang harus dilakukan
pengguna pada saat mengoperasikan sistem}
procedure Petunjuk(P:string);
var x,y,n:integer;
begin
  textcolor(15);
  textbackground(8);
  gotoxy(1,24);write(P);
  x:=wherex;
  y:=wherey;
  n:=81-x;
  gotoxy(x,y);write(' ':n);
  textcolor(0);
  textbackground(15);
end;

```

```

{prosedur untuk menampilkan informasi pada pengguna tentang kesalahan yang
terjadi dan saran pembetulannya}
procedure Peringatan(P:string);
var x,y,n:integer;
begin
  textcolor(15);
  textbackground(4);write(#7);
  gotoxy(1,24);write(P);
  x:=wherex;
  y:=wherey;
  n:=81-x;
  gotoxy(x,y);write(' ':n);
  textbackground(15);
  textcolor(0);
end;

```

```

{prosedur untuk menempatkan suatu elemen matriks pada suatu matriks sesuai
dengan baris dan kolomnya}
procedure sisip_element(var awal : Pmat;b,k : byte;n : integer);
var Baru,sb,sk,n_baris,n_kolom,bb : Pmat;
  ada : boolean;
begin
  buat_simpul(Baru,b,k,n);
  ada_baris(n_baris,ada,awal,b);
  if not (ada) then
    begin
      buat_simpul(sb,b,0,0);

```

```

    sb^.kanan := sb;
    sb^.bawah := sb;
    bb := awal;
    while (bb^.bawah^.baris < b) and(bb^.bawah <> awal) do
        bb := bb^.bawah;
        if bb^.bawah = awal then
            sb^.bawah := awal
        else
            sb^.bawah := bb^.bawah;
            bb^.bawah := sb;
            n_baris := sb;
        end;
    ada_kolom(n_kolom, ada, awal, k);
    if not (ada) then
        begin
            buat_simpul(sk, 0, k, 0);
            sk^.kanan := sk;
            sk^.bawah := sk;
            bb := awal;
            while (bb^.kanan^.kolom < k) and(bb^.kanan <> awal) do
                bb := bb^.kanan;
                if bb^.kanan = awal then
                    sk^.kanan := awal
                else
                    sk^.kanan := bb^.kanan;
                    bb^.kanan := sk;
                    n_kolom := sk;
                end;
            sb := n_baris;
            while (sb^.kanan^.kolom < k) and(sb^.kanan^.kolom <> 0) do
                sb := sb^.kanan;
                if sb^.kanan^.kolom = 0 then
                    Baru^.kanan := n_baris
                else
                    Baru^.kanan := sb^.kanan;
                    sb^.kanan := Baru;
                    sk := n_kolom;
                    while (sk^.bawah^.baris < b) and(sk^.bawah^.baris <> 0) do
                        sk := sk^.bawah;
                        if sk^.bawah^.baris = 0 then
                            Baru^.bawah := n_kolom
                        else
                            Baru^.bawah := sk^.bawah;
                            sk^.bawah := Baru
                        end;
                    end;

```

{prosedur untuk mencari elemen matriks tertentu pada suatu matriks dalam rangkaian dan menempatkan alamat simpul pada posisi tersebut}

```
function CariMatrik(M : Pmat; i, j : byte):Pmat;
```

```
var Caribrs, cariklm : Pmat;
```

```
begin
```

```
  if M <> nil then
```

```
    begin
```

```
      caribrs := M^.bawah;
```

```
      if caribrs <> nil then
```

```
        begin
```

```

while (caribrs <> nil) and (caribrs^.baris <> i) do
  caribrs := caribrs^.bawah;
  cariklm := caribrs^.kanan;
  while (cariklm <> nil) and (cariklm^.kolom <> j) do
    cariklm := cariklm^.kanan;
  CariMatrik := cariklm
end
else
  CariMatrik := caribrs;
end
else
  CariMatrik := M;
end;

```

{prosedur untuk mengalikan dua matriks x dan matriks y dan menghasilkan matriks hasil yaitu kali yang akan dikembalikan ke sistem}

```

procedure kali_matriks(x,y : Pmat; var kali : Pmat);

```

```

var N,S,M : tabel;

```

```

  I,J,k,brs,klm : byte ;

```

```

  jml : integer;

```

```

begin

```

```

  buat_simpul(kali,0,0,0);

```

```

  kali^.kanan := kali;

```

```

  kali^.bawah := kali;

```

```

  N[1] := x^.bawah;

```

```

  N[2] := y^.kanan;

```

```

  repeat

```

```

    repeat;

```

```

      jml := 0;

```

```

      S[1] := N[1]^kanan;

```

```

      S[2] := N[2]^bawah;

```

```

      repeat

```

```

        if S[1]^kolom <> S[2]^baris then

```

```

          begin

```

```

            jml := jml;

```

```

            if S[1]^kolom > S[2]^baris then

```

```

              S[2] := S[2]^bawah

```

```

            else

```

```

              S[1] := S[1]^kanan;

```

```

            end

```

```

          else

```

```

            begin

```

```

              jml := jml + S[1]^nilai * S[2]^nilai;

```

```

              S[2] := S[2]^bawah;

```

```

              S[1] := S[1]^kanan;

```

```

            end;

```

```

          until (S[1] = N[1]) or (S[2] = N[2]);

```

```

          brs := S[1]^baris;

```

```

          klm := S[2]^kolom;

```

```

          if (brs <> 0) and (klm <> 0) then

```

```

            sisip_element(kali,brs,klm,jml);

```

```

          N[2] := N[2]^kanan;

```

```

          until N[2] = y;

```

```

          N[1] := N[1]^bawah;

```

```

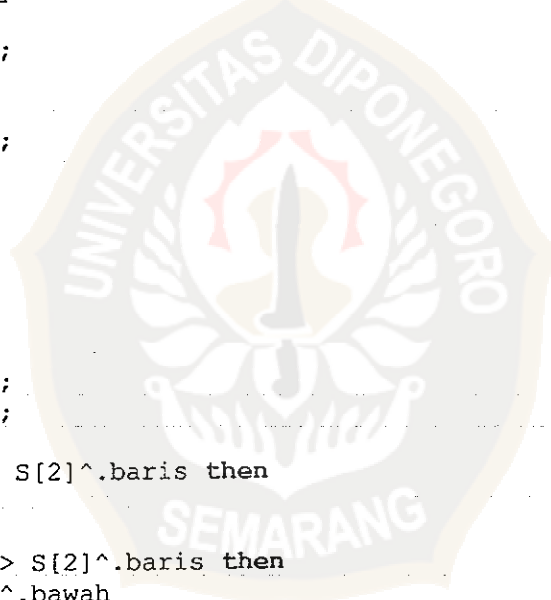
        until N[1] = x;

```

```

      end;

```



```

{prosedur untuk mencetak elemen matriks dalam bentuk matriks (segi empat)}
procedure cetak_matriks(awal : Pmat);
var bb,bb1 : Pmat;
    jk,jb,i,j,n_bar,n_kol,uk,baris : integer;
begin
    baris := 0;
    bb := awal^.kanan;
    while bb^.kanan^.kolom <> 0 do
        bb := bb^.kanan;
    jk := bb^.kolom;
    bb := awal^.bawah;
    while bb^.bawah^.baris <> 0 do
        bb := bb^.bawah;
    jb := bb^.baris;
    uk := jb;
    if (jb <> jk) then
        if jb < jk then
            uk := jk
        else
            uk := jb;
    bb := awal^.bawah;
    n_bar := 0;
    repeat
        if (bb^.baris - N_bar) <> 1 then
            for i := n_bar + 1 to bb^.baris - 1 do
                begin
                    for j := 1 to uk do
                        write(0:8);
                        writeln
                    end;
                bbl := bb^.kanan;
                if (bbl^.kolom - bb^.kolom) <> 1 then
                    for i := 1 to bbl^.kolom - 1 do
                        write(0:8);
                    repeat
                        baris := baris + 1;
                        write('Elemen baris ',bbl^.baris,' kolom ',bbl^.kolom,' = ');
                        writeln(bbl^.nilai:8);
                        if baris = 20 then
                            begin
                                baris := 0;readln;
                                clrscr;
                            end;
                        n_kol := bbl^.kolom;
                        bbl := bbl^.kanan;
                        if bbl^.kolom <> 0 then
                            if (bbl^.kolom - n_kol) <> 1 then
                                for i := 1 to bbl^.kolom-n_kol-1 do
                                    write(0:8);
                                until bbl^.kolom = 0;
                                writeln;
                                n_bar := bb^.baris;
                                bb := bb^.bawah
                            until bb^.baris = 0;
                    end;

```

```

{prosedur untuk mencetak elemen matriks tidak dalam bentuk matriks}
procedure cetak_matriks2(awal : Pmat);
var bb, bbl : Pmat;
    jk, jb, i, j, n_bar, n_kol, uk, baris : integer;
begin
    baris := 0;
    bb := awal^.kanan;
    while bb^.kanan^.kolom <> 0 do
        bb := bb^.kanan;
    jk := bb^.kolom;
    bb := awal^.bawah;
    while bb^.bawah^.baris <> 0 do
        bb := bb^.bawah;
    jb := bb^.baris;
    uk := jb;
    if (jb <> jk) then
        if jb < jk then
            uk := jk
        else
            uk := jb;
    bb := awal^.bawah;
    n_bar := 0;
    repeat
        if (bb^.baris - N_bar) <> 1 then
            for i := n_bar + 1 to bb^.baris - 1 do
                begin
                    for j := 1 to uk do
                        write(0:8);
                    writeln
                end;
            bbl := bb^.kanan;
            if (bbl^.kolom - bb^.kolom) <> 1 then
                for i := 1 to bbl^.kolom - 1 do
                    write(0:8);
                repeat
                    write(bbl^.nilai:8);
                    n_kol := bbl^.kolom;
                    bbl := bbl^.kanan;
                    if bbl^.kolom <> 0 then
                        if (bbl^.kolom - n_kol) <> 1 then
                            for i := 1 to bbl^.kolom-n_kol-1 do
                                write(0:8);
                            until bbl^.kolom = 0;
                            writeln;
                            n_bar := bb^.baris;
                            bb := bb^.bawah
                        until bb^.baris = 0;
                    end;

{prosedur untuk delay sampai waktu tunggu selesai atau sampai suatu tombol
ditekan}
procedure TungguTekan(Tunda : integer);
var i, j, x : integer;
    T, tekan : char;
begin
    i := 0;

```

```

x := Tunda*100;
repeat
  i := i+1;
  delay(10);
until (keypressed) or (i > x);
if keypressed then
begin
  tekan := readkey;
  write(#7);
end;
end;

```

{prosedur untuk menampilkan isi informasi tentang fungsi dari bagian-bagian yang terdapat pada menu utama}

```
procedure InformasiMenuUtama;
```

```
var i : integer;
```

```
begin
```

```
  textbackground(15);clrscr;
```

```
  textcolor(4);gotoxy(1,2);
```

```
  write('INFORMASI MENU');
```

```
  textcolor(1);
```

```
  gotoxy(15,2);write('=====');
```

```
  gotoxy(1,4);write(DataMenu.IsiMenu[2]);
```

```
  gotoxy(1,7);write(DataMenu.IsiMenu[3]);
```

```
  gotoxy(1,12);write(DataMenu.IsiMenu[4]);
```

```
  gotoxy(1,16);write(DataMenu.IsiMenu[5]);
```

```
  gotoxy(1,20);write(DataMenu.IsiMenu[6]);
```

```
  textcolor(0);
```

```
  gotoxy(3,5);write('Memberi masukan matriks dari papan ketik');
```

```
  gotoxy(3,8);write('Memberi masukan matriks dengan membuka file yang menyimpan data matriks');
```

```
  gotoxy(3,9);write('nama file dan posisi awal serta akhir rangkaian diketikkan pada tempat ');
```

```
  gotoxy(3,10);write('yang disediakan');
```

```
  gotoxy(3,13);write('Menyimpan rangkaian matriks yang ada di memori ke media penyimpanan');
```

```
  gotoxy(3,14);write('disket atau hard disk');
```

```
  gotoxy(3,17);write('Melakukan proses dimana dapat dipilih menggunakan Algoritma Pemrograman');
```

```
  gotoxy(3,18);write('Dinamik atau tidak dan menampilkan hasilnya beserta lama eksekusinya');
```

```
  gotoxy(3,21);write('Kembali ke DOS');
```

```
  gotoxy(3,22);write('data rangkaian matriks otomatis hilang jika tidak disimpan ke file ');
```

```
  gotoxy(1,24);textcolor(1);
```

```
  write('=====');
```

```
  gotoxy(10,25);textcolor(4);write('Tekan <enter>');readln;
```

```
end;
```

{prosedur untuk menampilkan kover program pada layar}

```
procedure Cover;
```

```
begin
```

```
  textbackground(15);
```

```
  textcolor(0);clrscr;
```

```
  gotoxy(1,4);
```

```
  writeln('':8,'=====');
```



```

writeln('':8,' ');
writeln('':8,' Aplikasi Penyelesaian Masalah Perkalian Rangkaian Matriks ');
writeln('':8,' ');
writeln('':8,' Disusun oleh : ');
writeln('':8,' Helmie ARif Wibawa ');
writeln('':8,' ( J2A 096 060 ) ');
writeln('':8,' ');
writeln('':8,' Jurusan Matematika ');
writeln('':8,' Fakultas Matematika dan Ilmu Pengetahuan Alam ');
writeln('':8,' Universitas Diponegoro ');
writeln('':8,' Semarang ');
writeln('':8,' 2001 ');
writeln('':8,' ');
TungguTekan(5);
end;

```

{prosedur untuk mendapatkan masukan nama file yang akan digunakan dan drive tempat penyimpanan}

```

procedure FormBSFile(var NamaFile:string);

```

```

begin

```

```

  textbackground(0);

```

```

  textcolor(15);

```

```

  gotoxy(20,13);write(' ');

```

```

  gotoxy(20,14);write(' Masukan nama file yang akan di gunakan ');

```

```

  gotoxy(20,15);write(' dengan nama drive tempat penyimpanan ');

```

```

  gotoxy(20,16);write(' Nama file : ');

```

```

  gotoxy(20,17);write(' ');

```

```

  gotoxy(38,16);readln(NamaFile);

```

```

  gotoxy(38,16);write(' ');

```

```

  textbackground(15);

```

```

end;

```

{prosedur untuk menampilkan efek sorotan pada salah satu pilihan yang terdapat pada layar menu}

```

procedure ChAtributLayar(x1,y1,x2,y2,Atribut : byte);

```

```

var posisi : word;

```

```

    i,j : byte;

```

```

begin

```

```

  for i := y1 to y2 do

```

```

    begin

```

```

      for j := x1 to x2 do

```

```

        begin

```

```

          posisi := (((i-1)*80+j)-1)*2;

```

```

          Mem[$B800:$0000+posisi+1] := Atribut;

```

```

        end;

```

```

      end;

```

```

    end;

```

{prosedur untuk mengembalikan layar sesuai dengan layar yang diinginkan pada suatu posisi tertentu}

```

procedure KembalikanLayar(x1,y1,x2,y2:byte;var LayarSimpan : BufferLayar);

```

```

var posisi : word;

```

```

    i,j : byte;

```

```

begin

```

```

  for i := y1 to y2 do

```

```

begin
  for j := x1 to x2 do
    with LayarSimpan[i,j] do
      begin
        posisi := (((i-1)*80+j)-1)*2;
        Mem[$B800:$0000+posisi] := ord(karakter);
        Mem[$B800:$0000+posisi+1] := Atribut;
      end;
    end;
  end;

{prosedur untuk menyimpan data ke dalam file dengan nama dan drive yang
dimasukkan}
procedure SimpanKeFile(var SS:Matrik; p1,p2 :bitel; n:byte);
var Berkas      : text;
    NamaFile    : string;
    S           : Matrik;
    i           : byte;
    SudahAda    : boolean;
    F,G         : Pmat;
begin
  S:=SS;
  FormBSFile(NamaFile);
  Assign(Berkas,NamaFile);
  {$I-}
  reset(Berkas);
  {$I+}
  SudahAda := (IOResult=0);
  if SudahAda then
  begin
    jawab := ' ';
    while Not((jawab='Y') or (jawab='T')) do
      begin
        KembalikanLayar(20,13,85,17,LayarMenuUtama);
        write(chr(7));textcolor(4);
        gotoxy(18,20);write('File sudah ada, Tumpangi !!! (Y / T) ? ');
        readln(jawab);textcolor(0);
        jawab := upcase(jawab);
      end;
    if jawab = 'T' then exit;
  end;
  rewrite(Berkas);
  writeln(berkas,n:3);
  for i := 1 to n do
  begin
    writeln(berkas,p1[i]:3,p2[i]:3);
    G := S[i]^bawah;
    repeat
      F := G^.kanan;
    repeat
      textbackground(0);textcolor(15);
      gotoxy(20,14);write(' ');
      gotoxy(20,15);write(' Tunggu sedang menyimpan ..... ');
      gotoxy(20,16);write(' ');
      textbackground(15);textcolor(0);
      writeln(berkas,F^.nilai:5);

```

```

    F := F^.kanan;
  until F = G;
  G := G^.bawah;
  until G = S[i];
end;
close(Berkas);
end;

```

{prosedur untuk menampilkan form untuk memberikan masukan melalui papan ketik pada sistem}

```

procedure FormMasukan;
var i:byte;
begin
  textbackground(15);clrscr;
  gotoxy(33,1);write('MASUKAN DATA');
  gotoxy(3,2);write(' ');
  gotoxy(3,3);write(' Banyak matrik dalam rangkaian : ');
  gotoxy(3,4);write(' ');
  gotoxy(3,7);write(' — UKURAN MATRIKS KE- ');
  gotoxy(3,8);write(' Banyaknya baris : ');
  gotoxy(3,9);write(' Banyaknya kolom : ');
  gotoxy(3,10);write(' ');
  gotoxy(3,12);write(' — ELEMEN MATRIKS KE- ');
  gotoxy(3,13);write(' ');
  gotoxy(3,14);write(' Baris ke : ');
  gotoxy(3,15);write(' Kolom ke : ');
  gotoxy(3,16);write(' Nilai elemen : ');
  gotoxy(3,17);write(' ');
  gotoxy(3,19);write(' — AWAL DAN AKHIR ');
  gotoxy(3,20);write(' Awal rangkaian terletak pada matriks ke : ');
  gotoxy(3,21);write(' Akhir rangkaian terletak pada matriks ke : ');
  gotoxy(3,22);write(' ');
  move(Layar,LayarMasukan,4000);
end;

```

{prosedur untuk mendapatkan masukan berupa banyaknya matriks yang terdapat dalam rangkaian}

```

procedure MasukanBanyakMatriks(var n: byte);
var IOn : byte;
begin
  petunjuk('Masukan banyaknya matriks yang ada dalam rangkaian ');
  {$I-}
  repeat
    gotoxy(38,3);readln(n);
    IOn := IOresult;
    if IOn <> 0 then
      begin
        peringatan(' tipe data tidak valid');
        KembaliLayar(38,3,40,3,LayarMasukan);
      end;
  until IOn = 0;
  {$I+}
end;

```

{prosedur untuk mendapatkan masukan berupa elemen suatu matriks dalam rangkaian sesuai dengan baris dan kolom yang dimasukkan}

```

procedure MasukanElemenMatriks(var nilai :integer;i,j: byte);
var IOnilai : byte;
begin
  petunjuk('Masukan nilai elemen untuk baris dan kolom yang sesuai ');
  gotoxy(16,14);write(i);
  gotoxy(16,15);write(j);
  {$I-}
  repeat
    gotoxy(20,16);readln(nilai);
    IOnilai := IOresult;
    if IOnilai <> 0 then
      begin
        KembalikanLayar(20,16,25,16,LayarMasukan);
        peringatan(' Tipe nilai yang dimasukkan salah, seharusnya bertipe
          integer');
      end;
    until IOnilai = 0;
  {$I+}
end;

```

{prosedur untuk mendapatkan masukan berupa posisi awal rangkaian yaitu posisi matriks pertama dalam rangkaian}

```

procedure MasukanAwal(var i : byte);
var IOawal : byte;
begin
  petunjuk('Masukan posisi matriks yang berada di awal rangkaian ');
  {$I-}
  repeat
    gotoxy(48,20);readln(i);
    IOawal := IOresult;
    if IOawal <> 0 then
      begin
        KembalikanLayar(48,20,50,20,LayarMasukan);
        Peringatan(' Tipe data awal rangkaian salah, seharusnya bertipe integer');
      end;
    until IOawal = 0;
  {$I+}
end;

```

{prosedur untuk mendapatkan masukan berupa posisi akhir rangkaian yaitu berupa posisi matriks terakhir dalam rangkaian}

```

procedure MasukanAkhir(var j : byte);
var IOakhir : byte;
begin
  petunjuk('Masukan posisi matriks yang berada di akhir rangkaian ');
  {$I-}
  repeat
    gotoxy(48,21);readln(j);
    IOakhir := IOresult;
    if IOakhir <> 0 then
      begin
        KembalikanLayar(48,21,50,21,LayarMasukan);
        Peringatan(' Tipe data akhir rangkaian salah, harusnya bertipe integer');
      end;

```

```

until IOakhir = 0;
  {$I+}
end;

{prosedur untuk mendapatkan masukan berupa banyaknya baris suatu matriks}
procedure MasukanBaris(var i : byte);
var IObaris : byte;
begin
  petunjuk(' Masukkan banyaknya baris pada matriks !');
  {$I-}
  repeat
    gotoxy(23,8);readln(i);
    IObaris := IOresult;
    if IObaris <> 0 then
      begin
        KembalikanLayar(23,8,25,8,LayarMasukan);
        Peringatan(' Tipe data banyak baris salah, harusnya bertipe integer');
      end;
  until IObaris = 0;
  {$I+}
end;

{prosedur untuk mendapatkan masukan berupa banyaknya kolom suatu matriks}
procedure MasukanKolom(var j : byte);
var IOkolom : byte;
begin
  petunjuk(' Masukkan banyaknya kolom pada matriks ! ');
  {$I-}
  repeat
    gotoxy(23,9);readln(j);
    IOkolom := IOresult;
    if IOkolom <> 0 then
      begin
        KembalikanLayar(23,9,25,9,LayarMasukan);
        Peringatan(' Tipe data banyak kolom salah, harusnya bertipe integer');
      end;
  until IOkolom = 0;
  {$I+}
end;

{prosedure untuk memasukkan posisi baris suatu elemen matriks pada saat akan
dilakukan pembetulan elemen matriks}
procedure Gantibarís(var i : byte);
var IObaris : byte;
begin
  {$I-}
  repeat
    textbackground(1);textcolor(15);
    gotoxy(19,20);write(' ');
    gotoxy(19,20);readln(i);
    IObaris := IOresult;
    if IObaris <> 0 then
      begin
        Peringatan(' Tipe data baris salah, harusnya bertipe integer');
      end;
  until IObaris = 0;

```

```

    {$I+}
    textbackground(15);
    textcolor(0);
end;

```

{prosedure untuk memasukkan posisi kolom suatu elemen matriks yang akan diganti pada saat dilakukan pembetulan elemen matriks}

```

procedure Gantikolom(var j : byte);
var IOkolom : byte;
begin
    {$I-}
    repeat
    textbackground(1);
    textcolor(15);
    gotoxy(44,20);write('      ');
    gotoxy(44,20);readln(j);
    IOkolom := IOresult;
    if IOkolom <> 0 then
    begin
        Peringatan(' Tipe data kolom salah, harusnya bertipe integer');
    end;
    until IOkolom = 0;
    {$I+}
    textbackground(15);textcolor(0)
end;

```

{prosedur untuk memasukkan elemen pengganti pada saat akan dilakukan pembetulan elemen matriks}

```

procedure GantiNilai(var i : integer);
var IOnilai : byte;
begin
    {$I-}
    repeat
    textbackground(1);textcolor(15);
    gotoxy(29,21);readln(i);
    IOnilai := IOresult;
    if IOnilai <> 0 then
    begin
        gotoxy(29,21);write('      ');
        Peringatan(' Tipe data untuk elemen salah, harusnya bertipe integer');
    end;
    until IOnilai = 0;
    {$I+}
    textbackground(15);textcolor(0);
end;

```

{Prosedur untuk mencetak nama-nama file pada suatu direktori ke layar pada saat dilakukan proses buka file}

```

procedure TulisFile;
var DirCari : string[139];
    hasil : SearchRec;
begin
    clrscr;
    gotoxy(3,3);write('File : ');
    gotoxy(10,3); readln(DirCari);writeln;
    FindFirst(DirCari,Archive,hasil);

```

```

while DosError = 0 do
begin
write(hasil.name:16);
FindNext(Hasil);
end;
end;

```

{prosedure untuk memberikan pesan kesalahan yang terjadi pada proses pemasukan posisi awal rangkaian}

```
procedure AwalSalah(i:byte);
```

```
begin
```

```
textbackground(4);
```

```
textcolor(7);
```

```
gotoxy(10,15);write(' ');
```

```
gotoxy(10,16);write(' POSISI MATRIKS PERTAMA SALAH ');
```

```
gotoxy(10,17);write(' SEHARUSNYA BERADA DI ANTARA 1 DAN ',i,' ');
```

```
gotoxy(10,18);write(' BETULKAN DAHULU ! ');
```

```
gotoxy(10,19);write(' ');
```

```
textcolor(0);
```

```
textbackground(7);
```

```
readln;
```

```
end;
```

{prosedur untuk menampilkan pesan kesalahan pada saat proses memasukkan posisi akhir rangkaian}

```
procedure AkhirSalah(j:byte);
```

```
begin
```

```
textbackground(4);
```

```
textcolor(7);
```

```
gotoxy(10,15);write(' ');
```

```
gotoxy(10,16);write(' POSISI MATRIKS TERAKHIR SALAH ');
```

```
gotoxy(10,17);write(' TIDAK BOLEH LEBIH DARI ',j,' ');
```

```
gotoxy(10,18);write(' BETULKAN DAHULU ! ');
```

```
gotoxy(10,19);write(' ');
```

```
textcolor(0);
```

```
textbackground(15);
```

```
readln;
```

```
end;
```

{prosedur untuk menampilkan pesan kesalahan pada saat terjadi kesalahan pada memasukkan banyaknya baris (baris ke i ≠ kolom ke i-1)}

```
procedure MasukanSalah(k,p :byte);
```

```
begin
```

```
textbackground(4);
```

```
textcolor(7);
```

```
gotoxy(10,15);write(' ');
```

```
gotoxy(10,16);write(' KOLOM MATRIKS ',k-1,' DAN BARIS MATRIKS ',k,' ');
```

```
gotoxy(10,17);write(' TIDAK SESUAI !!! ');
```

```
gotoxy(10,18);write(' SEHARUSNYA BANYAKNYA BARIS MATRIKS ',k,' ADALAH',p,' ');
```

```
gotoxy(10,19);write(' BETULKAN DAHULU ! ');
```

```
gotoxy(10,20);write(' ');
```

```
textcolor(0);textbackground(15);
```

```
readln;
```

```
end;
```

```

{prosedure induk untuk melakukan proses masukan data rangkaian matriks}
procedure input(var p,p1,p2:bitel; var matrix :matrik; var n,ii,jj : byte);
var i,j,k,bar,kol,brs,klm,bantu : byte;
    nla : integer;
    benar : boolean;
    edit,jawab : char;
    M,MM,MA,Mbawah,Mkolom : Pmat;
    IOn : integer;
begin
    KembalikanLayar(1,1,80,25,LayarMasukan);
    MasukanBanyakMatriks(n);
    bantu := 0;
    for k := 1 to n do
        begin
            buat_simpul(matrix[k],0,0,0);
            matrix[k]^bawah := matrix[k];
            matrix[k]^kanan := matrix[k];
            bantu := bantu + 1;
            gotoxy(26,7);write(k);
            gotoxy(26,12);write(k);
            MasukanBaris(p1[k]);
            repeat
                benar := true;
                petunjuk('Masukan banyaknya baris pada matriks ');
                if (k > 1) and (p1[k] <> p2[k-1]) then
                    begin
                        MasukanSalah(k,p2[k-1]);
                        KembalikanLayar(10,15,59,20,LayarMasukan);
                        KembalikanLayar(22,8,26,9,LayarMasukan);
                        MasukanBaris(p1[k]);
                        benar := false;
                    end;
            until benar;
            petunjuk('Masukkan banyaknya kolom pada matriks ');
            MasukanKolom(p2[k]);
            p[k] := p2[k];
            for i := 1 to p1[k] do
                begin
                    bar := i;
                    for j := 1 to p2[k] do
                        begin
                            kol := j;
                            MasukanElemenMatriks(nla,i,j);
                            sisip_element(matrix[k],bar,kol,nla);
                            if bantu <> n then
                                KembalikanLayar(15,14,20,15,LayarMasukan);
                            if (bantu <> n) and (k <> n) and (p2[k] <> p2[n]) then
                                KembalikanLayar(18,16,22,16,LayarMasukan);
                        end;
                    end;
                repeat
                    edit := ' ';
                    while not ((edit = 'Y') or (edit = 'T')) do
                        begin
                            clrscr;
                            gotoxy(5,5);writeln('matriks ke- ',k,' yang anda masukkan adalah

```



```

                sebagai berikut :');writeln;
cetak_matriks2(matrix[k]);
textcolor(1);
gotoxy(15,20);write(' Apakah ada data yang salah (Y/T) ? ');read(edit);
edit := upcase(edit); textcolor(0);
KembalikanLayar(15,20,40,20,LayarMasukan);
end;
if edit = 'Y' then
begin
    textbackground(1);
    textcolor(15);
    gotoxy(3,18);write(' ');
    gotoxy(3,19);write(' Masukkan posisi elemen dan nilai elemen yang akan
                        diganti ! ');
    gotoxy(3,20);write(' Posisi baris :           kolom : ');
    gotoxy(3,21);write(' Nilai elemen yang baru : ');
    gotoxy(3,22);write(' ');
    petunjuk('Masukan posisi baris dari elemen yang akan diganti ');
    repeat
        benar := true;
        GantiBaris(bar);
        if (1 > bar) or (bar > pl[k]) then
        begin
            peringatan(' Posisi baris untuk elemen harus disesuaikan ukuran
                        barisnya');
            benar := false;
        end;
    until benar;
    petunjuk('Masukan posisi kolom dari elemen yang akan diganti ');
    repeat
        benar := true;
        GantiKolom(kol);
        if (1 > kol) or (kol > p2[k]) then
        begin
            peringatan(' Posisi kolom untuk elemen yang akan diganti salah,
                        harus sesuai ukuran kolom');
            benar := false;
        end;
    until benar;
    petunjuk(' Masukkan elemen matriks yang baru ! ');
    GantiNilai(nla);
    M := CariMatrik(matrix[k],bar,kol);
    M^.nilai := nla;
    textcolor(0);textbackground(15);
end;
until edit = 'T';
KembalikanLayar(1,1,80,25,LayarMasukan);
if bantu <> n then
    KembalikanLayar(22,8,26,9,LayarMasukan);
end;
p[0] := pl[1];
repeat
    benar := true;
    MasukanAwal(ii);
    if (ii < 1) or (ii > n) then
        begin

```

```

    AwalSalah(n);
    KembalikanLayar(10,15,59,19,LayarMasukan);
    gotoxy(46,20);write(' ');
    benar := false;
end;
until benar;
repeat
    benar := true;
    MasukkanAkhir(jj);
    if (jj < 1) or (jj > n) then
        begin
            AkhirSalah(n);
            KembalikanLayar(10,15,65,19,LayarMasukan);
            gotoxy(46,21);write(' ');
            benar := false;
        end;
    until benar;
end;

```

{prosedure untuk menampilkan form pengisian posisi awal dan akhir rangkaian }

Procedure FormAwalAkhir(var i,j : byte;n :byte);

begin

textbackground(1);

textcolor(15);

gotoxy(3,19);write(' — MASUKAN POSISI AWAL DAN AKHIR RANGKAIAN — ');

gotoxy(3,20);write(' Terdapat ',n,' matriks dalam rangkaian ');

gotoxy(3,21);write(' Awal barisan terletak pada matriks ke : ');

gotoxy(3,22);write(' Akhir barisan terletak pada matriks ke : ');

gotoxy(3,23);write(' ');

gotoxy(46,21);readln(i);

gotoxy(46,22);readln(j);

textcolor(0);

textbackground(15);

end;

{prosedure untuk menampilkan form pertanyaan untuk masukan posisi pemecahan rangkaian pada proses penyelesaian tanpa menggunakan algoritma DP}

procedure FormTanya;

begin

textbackground(1);

textcolor(15);

gotoxy(3,19);write(' — LETAK PEMECAHAN BARISAN — ');

gotoxy(3,20);write(' Awal barisan : Akhir barisan : ');

gotoxy(3,21);write(' Pemecahan dilakukan pada posisi setelah matriks ke: ');

gotoxy(3,22);write(' ');

textcolor(0);

textbackground(15);

end;

{prosedure untuk membuka suatu file pada saat pemasukan melalui open file}

procedure BukaFile(var matrix: matrik; var p : bitel;var n : byte);

var p1,p2 : bitel;

brs,klm,i,k,j : byte;

nilai : integer;

NamaFile : string;

Berkas : text;

```

begin
  FormBSFile>NamaFile);
  Assign(Berkas,>NamaFile);
  {$I-}
  reset(Berkas);
  {$I+}
  if IOresult <> 0 then
  begin
    clrscr;
    Peringatan(' File tidak ditemukan ( tekan <enter> untuk ke menu utama ! ');
    textbackground(4); textcolor(15);
    gotoxy(20,13);write(' ');
    gotoxy(20,14);write(' File tidak ditemukan ... ');
    gotoxy(20,15);write(' ');
    gotoxy(48,14);readln;textbackground(15); textcolor(0);
  end
  else
  begin
    while not Eof(Berkas) do
    begin
      readln(Berkas,n);
      for k := 1 to n do
      begin
        buat_simpul(matrix[k],0,0,0);
        matrix[k]^bawah := matrix[k];
        matrix[k]^kanan := matrix[k];
        readln(Berkas,p1[k],p2[k]);
        for i := 1 to p1[k] do
          begin
            brs := i;
            for j := 1 to p2[k] do
              begin
                klm := j;
                readln(Berkas,nilai);
                sisip_element(matrix[k],brs,klm,nilai);
              end;
            end;
            p[k] := p2[k];
          end;
          p[0] := p1[1];
        end;
        close(Berkas);
        FormAwalAkhir(ii,jj,byk);
      end;
    end;
  end;

  {prosedure untuk memasukkan posisi pemecahan rangkaian}
  procedure tanya(var s : bite;i,j : byte);
  begin
    if j > i then
    begin
      gotoxy(20,20);write(i);
      gotoxy(56,20);write(j);
      gotoxy(58,21);readln(s[i,j]);
      tanya(s,i,s[i,j]);
      tanya(s,s[i,j]+1,j);
    end;
  end;

```

```

end;
end;

```

(prosedure untuk menentukan posisi pemecahan rangkaian dengan menggunakan algoritma DP)

```

procedure order( var s:bite;n :byte;p :bitel);

```

```

var i,j,k,l : byte;

```

```

    m:mskalar;

```

```

    q : longint;

```

```

begin

```

```

    for i:= 1 to n do

```

```

        m[i,i] :=0;

```

```

        for l := 2 to n do

```

```

            for i:= 1 to n-l+1 do

```

```

                begin

```

```

                    j:=i+l-1;

```

```

                    m[i,j]:= 2147483647;

```

```

                    for k:=i to j-1 do

```

```

                        begin

```

```

                            q:=m[i,k]+m[k+1,j]+p[i-1]*p[k]*p[j];

```

```

                            if (q > 0) and (q < m[i,j]) then

```

```

                                begin

```

```

                                    m[i,j]:= q;

```

```

                                    s[i,j]:= k;

```

```

                                end;

```

```

                            end;

```

```

                    end;

```

```

            end;

```

(prosedur untuk proses mengalikan rangkaian matriks)

```

procedure barisan(matrix: matrik; s: bite;i,j:byte ;var c:Pmat);

```

```

var x,y : Pmat;

```

```

begin

```

```

    buat_simpul(c,0,0,0);

```

```

    if j > i then

```

```

        begin

```

```

            barisan(matrix,s,i,s[i,j],x);

```

```

            barisan(matrix,s,s[i,j]+1,j,y);

```

```

            kali_matriks(x,y,c);

```

```

        end

```

```

    else

```

```

        c := h[i];

```

```

    end;

```

(prosedur untuk menampilkan rangkaian matriks lengkap dengan letak tanda kurungnya)

```

Procedure BentukPerkalian(i,j :byte;s:bite);

```

```

var k: byte;

```

```

begin

```

```

    if i = j then write('A',i)

```

```

    else

```

```

        begin

```

```

            k := s[i,j];

```

```

            write('(');

```

```

            BentukPerkalian(i,k,s);

```

```

            write('*');

```

```

    BentukPerkalian(k+1,j,s);
    write('');
end;
end;

{prosedur untuk menampilkan form menu utama}
procedure FormMenuUtama;
var i : byte;
begin
    textbackground(15);
    clrscr;
    Petunjuk('Gerakkan kursor '+chr(24)+' atau '+chr(25)+' atau tekan huruf dengan
              warna berbeda ');
    textcolor(1);
    gotoxy(12,2);write('APLIKASI PENYELESAIAN MASALAH PERKALIAN RANGKAIAN
                       Matriks');
    gotoxy(20,3);write('DENGAN ALGORITMA PEMROGRAMAN DINAMIK');
    textcolor(4);
    gotoxy(22,7);write('===== MENU Pengerjaan =====');
    gotoxy(22,8);write(' ');
    gotoxy(53,8);write(' ');
    for i:= 1 to 6 do
    begin
        textcolor(4);
        gotoxy(22,i+8);write(' ');
        gotoxy(53,i+8);write(' ');
        with DataMenu do
        begin
            textcolor(4);textbackground(15);
            gotoxy(27,posisiMenu[i]);textbackground(1);textcolor(7);write(IsiMenu[i]);
            gotoxy(posisiKarakterMenu[i],i+8);textcolor(14);write(KarakterMenu[i]);
        end;
        end;
        textcolor(4);gotoxy(22,i+10);write('=====');
        gotoxy(22,i+9);write(' ');
        gotoxy(53,i+9);write(' ');
        move(layar,LayarMenuUtama,4000);
    end;

{prosedure untuk menampilkan form menu proses}
procedure FormMenuProses;
var i : byte;
begin
    textbackground(15);textcolor(4);
    clrscr;
    Petunjuk('Gerakkan kursor '+chr(24)+' atau '+chr(25)+' atau tekan huruf dengan
              warna berbeda ');
    gotoxy(12,2);write('APLIKASI PENYELESAIAN MASALAH PERKALIAN RANGKAIAN
                       Matriks');
    gotoxy(22,7);write('===== PILIHAN =====');
    gotoxy(22,8);write(' ');
    gotoxy(52,8);write(' ');
    for i:= 1 to 3 do
    begin
        textcolor(4);textbackground(15);
        gotoxy(22,i+8);write(' ');

```

```

gotoxy(52,i+8);write(' ');
with DataMenuProses do
begin
  textcolor(0);
  gotoxy(27,posisiMenu[i]);textbackground(1);textcolor(7);write(IsiMenu[i]);
  gotoxy(posisiKarakterMenu[i],i+8);textcolor(14);write(KarakterMenu[i]);
end;
end;
textcolor(4);gotoxy(22,i+10);write(' ');
gotoxy(22,i+9);write(' ');
gotoxy(52,i+9);write(' ');
move(layar,LayarMenuProses,4000);
end;

```

{prosedur untuk melakukan proses perkalian (dengan algoritma DP atau tanpa DP),
mencetak hasilnya, dan mendapatkan lama eksekusi}

```

procedure Proses(var time : word);

```

```

begin
  GetTime(h1,m1,s1,hund1);
  barisan(h,tt,ii,jj,c);
  GetTime(h2,m2,s2,hund2);
  time := (h2-h1)*3600*100 + (m2-m1)*60*100 + (s2-s1)*100 + (hund2-hund1);
  cetak_matriks(c);
end;

```

{prosedur induk untuk melakukan proses perkalian rangkaian matriks}

```

procedure MenuProses;

```

```

var i,j,pilih,pilihprev,
    baris1      :byte;
    TekanMenu,selesai,t    : char;
    h3,h4,m3,m4,s3,s4,hund3,hund4,time,timel,waktu : word;

```

```

begin

```

```

  KembalikanLayar(1,1,80,25,LayarMenuProses);

```

```

  pilih:=1;

```

```

  baris1:=DataMenuProses.PosisiMenu[pilih];

```

```

  ChAtributLayar(27,baris1,47,baris1,$62);

```

```

  repeat

```

```

    TekanMenu := ' ';

```

```

    PilihPrev := Pilih;

```

```

    gotoxy(28,baris1);

```

```

    TekanMenu := readkey;

```

```

    if TekanMenu <> #0 then

```

```

      begin

```

```

        case upcase(TekanMenu) of

```

```

          'T': pilih:=1;

```

```

          'D': pilih:=2;

```

```

          'E': pilih:=3;

```

```

          #13: pilih:=PilihPrev;

```

```

        else

```

```

          write(#7);

```

```

      end;

```

```

    end;

```

```

    if TekanMenu = #0 then

```

```

      begin

```

```

        TekanMenu := readkey;

```

```

        case TekanMenu of

```

```

#72 : begin
  if pilih = 1 then
    pilih := 3
  else
    pilih := pilih-1;
  end;
#80 : begin
  if pilih = 3 then
    pilih := 1;
  else
    pilih := pilih+1;
  end;
else
  write(#7);
end;
end;
if Pilih <> PilihPrev then
begin
  baris1 := DataMenuProses.PosisiMenu[PilihPrev];
  KembalikanLayar (27, baris1, 50, baris1, LayarMenuProses);
  baris1 := DataMenuProses.PosisiMenu[pilih];
  ChAtributLayar (27, baris1, 47, baris1, $62);
end;
if (upcase(TekanMenu) = 'E') or ((pilih = 3) and (TekanMenu = #13)) then
exit;
KembalikanLayar (30, 20, 65, 20, LayarMenuProses);
if TekanMenu = #13 then
begin
  if pilih = 1 then
  begin
    FormTanya;
    tanya(tt, ii, jj);
  end;
  if pilih = 2 then
  begin
    GetTime (h3, m3, s3, hund3);
    order(tt, byk, r);
    GetTime (h4, m4, s4, hund4);
    time := (h4-h3)*3600*100 + (m4-m3)*60*100 + (s4-s3)*100 + hund4 - hund3;
  end;
  clrscr;
  writeln('Rangkaian matrik akan dikalikan menurut peletakan tanda kurung :');
  writeln('BentukPerkalian(ii, jj, tt);
  writeln;writeln;
  writeln('Hasil akhir perkalian adalah sebagai berikut : ');
  writeln;
  Proses(time1);
  if pilih = 2 then
    waktu := time1 + time
  else
    waktu := time1;
  writeln('Waktu mulai : ', h1, ':', m1, ':', s1, '.', hund1);
  writeln('Waktu selesai : ', h2, ':', m2, ':', s2, '.', hund2);
  writeln('Lama eksekusi : ', waktu, ' per 100 detik');
  readln;
  KembalikanLayar (1, 1, 80, 25, LayarMenuProses);

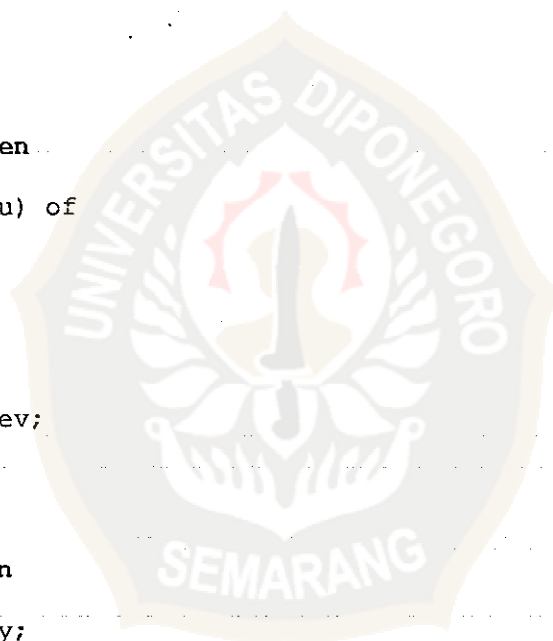
```

```

    end;
until false;
end;

{prosedur untuk melakukan proses menu utama}
procedure MenuUtama;
var i,j,pilih,pilihprev,
    baris1      :byte;
    TekanMenu,selesai,t      : char;
begin
    KembalikanLayar(1,1,80,25,LayarMenuUtama);
    pilih:=1;
    baris1:=DataMenu.PosisiMenu[pilih];
    ChAtributLayar(27,baris1,49,baris1,$62);
    repeat
        TekanMenu :=' ';
        PilihPrev := Pilih;
        gotoxy(28,baris1);
        TekanMenu := readkey;
        if TekanMenu <> #0 then
            begin
                case upcase(TekanMenu) of
                    'I': pilih:=1;
                    'M': pilih:=2;
                    'B': pilih:=3;
                    'S': pilih:=4;
                    'P': pilih:=5;
                    'E': pilih:=6;
                    #13: pilih:=PilihPrev;
                    else
                        write(#7);
                end;
            end;
        if TekanMenu = #0 then
            begin
                TekanMenu := readkey;
                case TekanMenu of
                    #72 : begin
                            if pilih = 1 then
                                pilih := 6
                            else
                                pilih := pilih-1;
                            end;
                        end;
                    #80 : begin
                            if pilih = 6 then
                                pilih := 1
                            else
                                pilih := pilih+1;
                            end;
                        end;
                    else
                        write(#7);
                end;
            end;
        if Pilih <> PilihPrev then
            begin
                baris1 := DataMenu.PosisiMenu[PilihPrev];

```

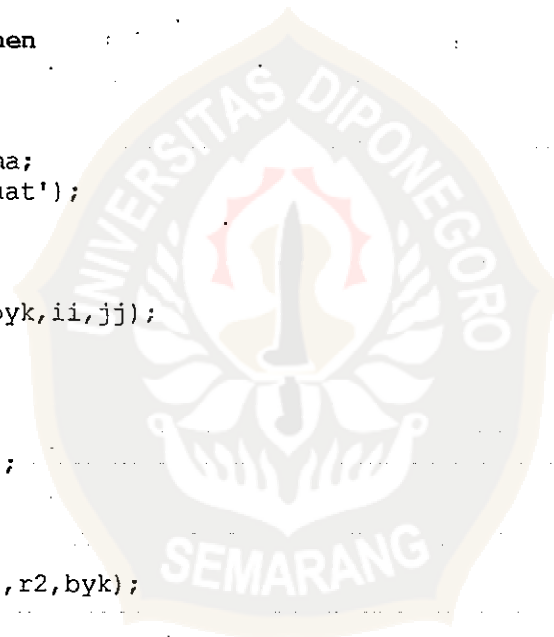



```

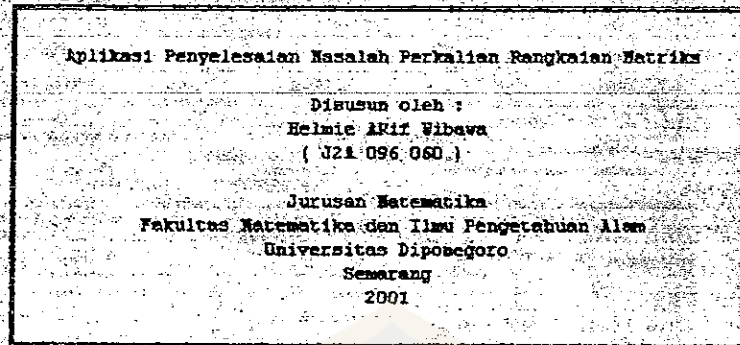
    KembalikanLayar(27,baris1,50,baris1,LayarMenuUtama);
    baris1 := DataMenu.PosisiMenu[pilih];
    ChAtributLayar(27,baris1,49,baris1,$62);
end;
if (upcase(TekanMenu) = 'E') or ((pilih = 6) and (TekanMenu = #13)) then
begin
    selesai := ' ';
    while not ((selesai = 'Y') or (selesai = 'T')) do
    begin
        textcolor(1);
        gotoxy(28,20); write(' Anda yakin (Y/T) ? '); read(selesai);
        selesai := upcase(selesai); textcolor(0);
    end;
    if selesai = 'Y' then halt;
    KembalikanLayar(30,20,65,20,LayarMenuUtama);
end;
if TekanMenu = #13 then
begin
    if pilih = 1 then
    begin
        InformasiMenuUtama;
        write('belom dibuat');
    end;
    if pilih = 2 then
    begin
        Input(r,r1,r2,h,byk,ii,jj);
    end;
    if pilih = 3 then
    begin
        TulisFile;
        BukaFile(h,r,byk);
    end;
    if pilih = 4 then
    begin
        SimpanKeFile(h,r1,r2,byk);
    end;
    if pilih = 5 then
    begin
        MenuProses;
    end;
    KembalikanLayar(1,1,80,25,LayarMenuUtama);
end;
until false;
end;

{program utama}
begin
    clrscr;
    FormMenuUtama;
    FormMenuProses;
    FormMasukan;
    Cover;
    MenuUtama;
end.

```

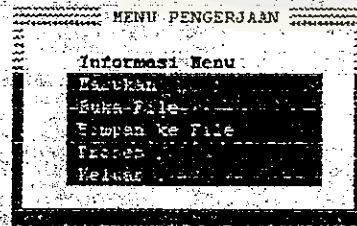


LAMPIRAN OUTPUT PROGRAM



Tampilan pembukaan program

APLIKASI PENYELESAIAN MASALAH PERKALIAN RANGKAIAN Matriks
DENGAN ALGORITMA PENROGRAMAN DINAMIK



Tampilan Menu Utama

INFORMASI MENU=====

Masukan Data
Memberi masukan matriks dari papan ketik

Buka File
Memberi masukan matriks dengan membuka file yang menyimpan data matriks
nama file dan posisi awal serta akhir rangkaian diketikkan pada tempat
yang disediakan

Simpan ke File
Menyimpan rangkaian matriks yang ada di memori ke media penyimpanan
disket atau hard disk

Proses
Melakukan proses dimana dapat dipilih menggunakan algoritma Pemrograman
Dinamik atau tidak dan menampilkan hasilnya beserta lama eksekusinya

Keluar
Kembali ke DOS
data rangkaian matriks otomatis hilang jika tidak disimpan ke file

=====

Tekan <enter>

Tampilan Informasi Menu

MASUKAN DATA

Banyak matrik dalam rangkaian :

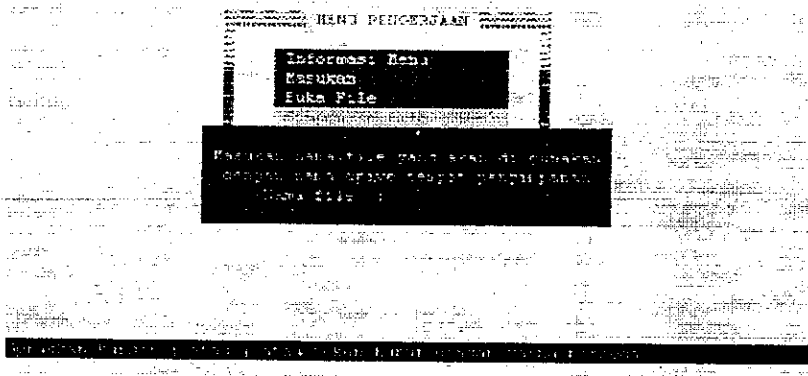
UKURAN MATRIKS KE-
Banyaknya baris :
Banyaknya kolom :

ELEMEN MATRIKS KE-
Baris ke :
Kolom ke :
Nilai elemen :

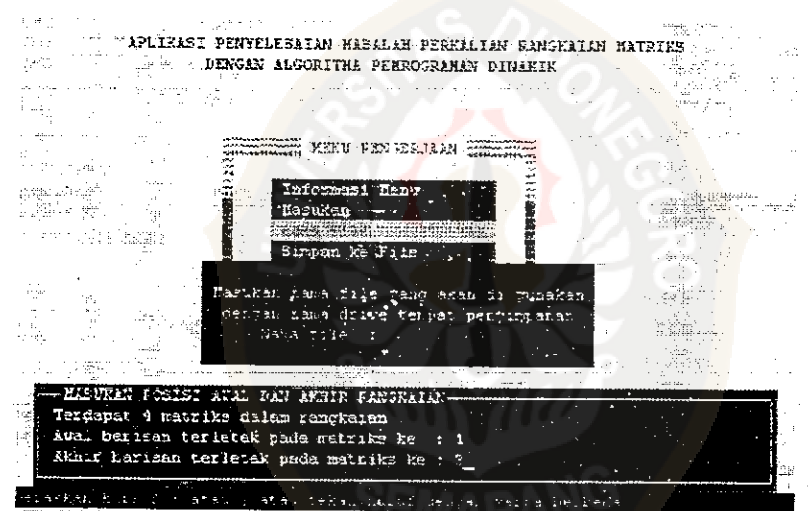
AWAL DAN AKHIR MATRIKS
Awal rangkaian terletak pada matriks ke :
Akhir rangkaian terletak pada matriks ke :

Tampilan Masukan

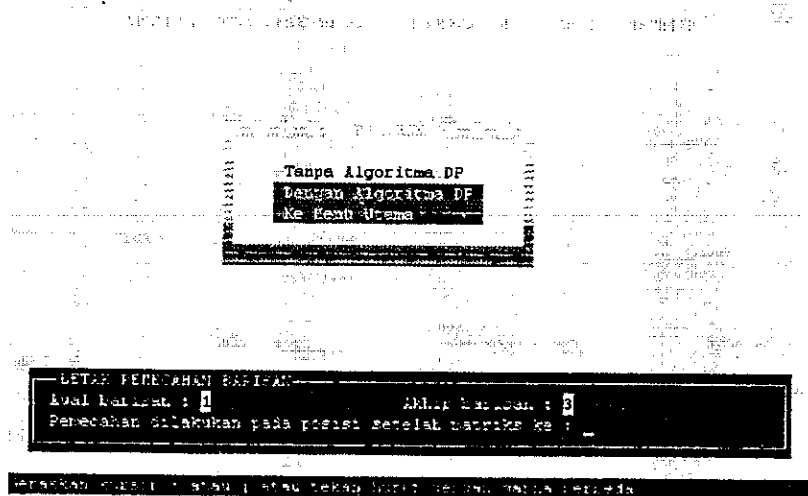
APLIKASI PENYELESAIAN MASALAH PERKALIAN BANGKAIAN MATRIKS
DENGAN ALGORITMA PEMROGRAMAN DINAMIK



Tampilan Simpan ke File



Tampilan Buka File



Tampilan Proses Tanpa DP

((A1*A2)*A3)*A4)

Hasil akhir perkalian adalah sebagai berikut :

Elemen baris 1 kolom 1 = 400
 Elemen baris 1 kolom 2 = 400
 Elemen baris 1 kolom 3 = 400
 Elemen baris 1 kolom 4 = 400
 Elemen baris 1 kolom 5 = 400
 Elemen baris 1 kolom 6 = 400
 Elemen baris 1 kolom 7 = 400
 Elemen baris 1 kolom 8 = 400
 Elemen baris 1 kolom 9 = 400
 Elemen baris 1 kolom 10 = 400
 Elemen baris 1 kolom 11 = 400
 Elemen baris 1 kolom 12 = 400
 Elemen baris 1 kolom 13 = 400
 Elemen baris 1 kolom 14 = 400
 Elemen baris 1 kolom 15 = 400
 Elemen baris 1 kolom 16 = 400
 Elemen baris 1 kolom 17 = 400
 Elemen baris 1 kolom 18 = 400
 Elemen baris 1 kolom 19 = 400
 Elemen baris 1 kolom 20 = 400

.....

Elemen baris 30 kolom 16 = 400
 Elemen baris 30 kolom 17 = 400
 Elemen baris 30 kolom 18 = 400
 Elemen baris 30 kolom 19 = 400
 Elemen baris 30 kolom 20 = 400
 Elemen baris 30 kolom 21 = 400
 Elemen baris 30 kolom 22 = 400
 Elemen baris 30 kolom 23 = 400
 Elemen baris 30 kolom 24 = 400
 Elemen baris 30 kolom 25 = 400

Waktu mulai : 5:16:51.80
 Waktu selesai : 5:16:51.91
 Lama eksekusi : 11 milidetik

Tampilan Sebagian Hasil Akhir

LAMPIRAN PERBANDINGAN WAKTU EKSEKUSI

Data :

Eksekusi dilakukan pada komputer dengan spesifikasi :

- Processor = pentium 133
- Ram = 16 MHz
- Harddisk = 1,2 GB

Banyaknya matriks dalam rangkaian adalah 4 matriks dengan elemen pada setiap baris dan kolom setiap matriks adalah 1.

Ukuran matriks

A_1	30×1	A_3	40×10
A_2	1×40	A_4	10×25

Hasil

- Letak tanda kurung : $(A_1.(A_2.(A_3.A_4)))$
Lama eksekusi : 5 milidetik
- Letak tanda kurung : $(A_1.((A_2.A_3).A_4))$
Lama eksekusi : 0 milidetik
- Letak tanda kurung : $((A_1.A_2).(A_3.A_4))$
Lama eksekusi : 11 milidetik
- Letak tanda kurung : $((A_1.(A_2.A_3)).A_4)$
Lama eksekusi : 6 milidetik
- Letak tanda kurung : $((((A_1.A_2).A_3).A_4))$
Lama eksekusi : 6 milidetik