

MODUL PRAKTIKUM

Revisi ke-5



ALGORITMA DAN PEMROGRAMAN KOMPUTER (Aplikasi Menggunakan Bahasa Pascal)

Disusun Oleh
Drs. Akmam, M.Si, dkk

**JURUSAN FISIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI PADANG
2017**

MODUL PRAKTIKUM I

A. Kompetensi yang diharapkan

Setelah kegiatan berlangsung diharapkan praktikan mampu:

1. Menggunakan fasilitas editor Turbo Pascal untuk penulisan *pseudocode* (penggunaan IDE)
2. Membuat *pseudocode* program sederhana yang user friendly dalam bahasa Turbo Pascal dan mudah ditelusuri
3. Memahami input/output (I/O) menggunakan *pseudocode* pada bahasa Turbo Pascal
4. Memahami kesalahan sintaks dan kesalahan logika pemrograman menggunakan *pseudocode* pada bahasa Turbo Pascal

B. Kegiatan Sebelum Praktikum

Sebelum melaksanakan praktikum pelajaryl Hand Out dan uraian berikut ini:

1. Menu dan Perintah Editor pada Turbo Pascal

Turbo Pascal dapat diaktifkan dengan meng-klik



Turbo Pascal 7.1.pif

untuk Pascal Under



TPW 1.5.lnk

Dos atau untuk yang Pascal under Windows terdapat pada deskop computer atau dengan :

1. Masuk editor DOS, dimana pada monitor akan muncul **C:\Windows>**
2. Kemudian tulis perintah **CD**, kemudian Enter, setelah muncul prompt **C:\>**
3. Tulis **CD TP\BIN>** kemudian tekan Enter
4. kemudian tulis **C:\TP7\BIN>turbo**

Pada saat turbo pasca diaktifkan, dilayar monitor akan muncul tampilan sebagai berikut:



Tampilan awal **Turbo Pascal**, seperti gambar diatas kita bisa lihat di Turbo Pascal memiliki 7 buah menu utama yang terdiri dari : **File, Edit, Run, Compile, Options, Debug**, dan **Break/Watch**. Untuk memilih salah satu dari menu itu anda bisa menekan tombol Alt + F (untuk

memanggil menu file), atau menekan Alt + [huruf pertama dari menu]. Jika anda ingin kembali ke menu utama anda bisa saja menekan tombol **F10**, untuk pindah dari menu satu ke menu yang lain anda juga bisa menekan panah kanan atau panah kiri. Contohnya ketika anda berada di menu **File** maka untuk pindah ke menu **Compile** anda cukup menggunakan tombol panah kanan atau panah kiri, dan anda juga bisa langsung menekan huruf **C**. Untuk menutup menu anda bisa menggunakan tombol **Esc**. Sebetulnya akan lebih mudah jika memanggil sebuah menu dengan menggunakan hotkey, table dibawah ini adalah hotkey yang digunakan dalam turbo pascal, **agar memudahkan dalam bekerja hafallah perintah berikut ini:**

| Kunci | Fungsi | Ekivalen menu |
|-----------|-----------------------------------------------------|---------------------|
| F1 | Mengaktifkan jendela pertolongan. | |
| F2 | Menyimpan berkas yang ada pada editor. | File / Save |
| F3 | Memanggil program kedalam editor. | File / Load |
| F4 | Mengeksekusi program sampai posisi kursor. | Run / Go to cursor |
| F5 | Memperbesar atau memperkecil jendela yang aktif. | |
| F6 | Mengganti jendela yang aktif. | |
| F7 | Melacak kedalam subrutin. | Run / Trace into |
| F8 | Melompati pemanggil subrutin. | |
| F9 | Mengaktifkan perintah 'Make'. | Compile / Make |
| F10 | Berpindah dari menu-menu ke jendela yang aktif. | |
| Alt + F1 | Memanggil jendela pertolongan yang terakhir dibaca. | |
| Alt + F3 | Memilih berkas untuk dimuat. | File / Pick |
| Alt + F5 | Memperlihatkan hasil eksekusi. | File / user screen |
| Alt + F6 | Mengganti isi jendela yang aktif. | |
| Alt + F9 | Mengkompilasi program. | Compile |
| Alt + B | Mengaktifkan menu Break / Watch. | |
| Alt + C | Mengaktifkan menu compile. | |
| Alt + D | Mengaktifkan menu debug. | |
| Alt + E | Mengaktifkan editor. | |
| Alt + F | Mengaktifkan menu file. | |
| Alt + O | Mengaktifkan menu option. | |
| Alt + R | Mengaktifkan menu run. | |
| Alt + X | Keluar dari Turbo Pascal dan kembali ke DOS. | |
| Ctrl + F1 | Menampilkan menu pertolongan bahasa. | |
| Ctrl + F2 | Menghentikan pembetulan. | Run / Program reset |

| | | |
|------------|----------------------------------------------------|-----------------------|
| Ctrl + F3 | Menampilkan isi tumpukan. | Debug / Call Stack |
| Ctrl + F4 | Melakukan penghitungan atau mengubah nilai peubah. | Debug / Evaluate |
| Ctrl + F7 | Menambahkan ungkapan pada jendela watch. | B / Add Watch |
| Ctrl + F8 | Toggles Breakpoint. | B / Toggle breakpoint |
| Ctrl + F9 | Menjalankan program. | Run / Run |
| Ctrl + F10 | Menampilkan versi layar monitor | |

Tabel di atas menunjukkan beberapa hotkey yang ada pada Turbo Pascal, sekarang kita langsung saja mempelajari menu di Turbo Pascal ini.

Menu File

Didalam menu File banyak submenu yang bisa dipilih misalkan saja perintah Load / F3 yang berfungsi untuk memanggil berkas yang sudah tersimpan, dan banyak lainnya.

Submenu Load

Pilihan ini digunakan untuk memanggil program dari suatu folder atau directory ke dalam editor. Didalam submenu ini anda juga bisa memilih submenu yang lain dengan hanya menekan tombol huruf depan dari submenu yang dituju, contohnya anda mau ke submenu **New**, maka anda tinggal menekan huruf **N**.

Submenu Pick

Pilihan yang satu ini digunakan untuk mengambil salah satu dari beberapa buah program yang sebelumnya telah dimuat ke dalam jendela **Edit**. Hal ini ditujukan agar kita tidak perlu repot-repot lagi mencarinya di submenu load.

Submenu New

Program yang ada di editor akan dihapus dan program yang akan anda tulis dianggap sebagai program yang baru dengan nama NONAME.PAS, nama ini bisa anda rubah ketika anda mau menyimpannya.

Submenu Save [F2]

Pilihan ini digunakan untuk menyimpan suatu program yang sudah di edit ke dalam cakram atau folder, jika nama program itu masih NONAME.PAS maka Turbo Pascal akan menanyakan pada anda, nama apa yang akan anda berikan pada program yang akan disimpan tersebut. Atau anda juga bisa menekan tombol F2.

Submenu Write to

pilihan ini digunakan untuk menggantikan program lama dengan program yang baru (istilah kerennya *Overwrite...* 😊). Jika nama program anda sudah ada pada cakram atau folder maka ada verifikasi dari Turbo Pascal untuk penggantian nama.

Submenu Directory

pilihan ini digunakan untuk menampilkan direktori dan nama-nama file yang anda inginkan. Anda juga bisa menggunakan pilihan **Load**.

Submenu Change Dir

Pilihan yang satu ini digunakan untuk menampilkan direktori yang sedang digunakan dan juga bisa digunakan untuk mengganti nama direktori yang baru.

Submenu OS shell

Perintah ini digunakan bagi anda yang ingin ke tampilan DOS tetapi tanpa mematikan Turbo Pascalnya, untuk kembali lagi ke Turbo Pascal, anda tinggal mengetikkan “exit”, tetapi biasanya ketika anda sedang membuat suatu program atau mengedit suatu program perintah ini tidak dapat dijalankan karena kurangnya kapasitas memory.

Submenu Quit [Alt + x]

Dari namanya sudah jelas, perintah ini digunakan untuk keluar dari Turbo Pascal dan kembali ke prompt DOS.

Menu Edit

Perintah edit ini digunakan untuk mengaktifkan editor Turbo Pascal.

Menu Run [Ctrl + F9]

Perintah ini digunakan untuk mengeksekusi program yang sudah anda buat. Anda juga bisa menekan Ctrl + F9 untuk mengeksekusi program anda.

Submenu Program Reset [Ctrl + F2]

Pilihan ini digunakan untuk memberitahukan Turbo Pascal bahwa anda selesai dengan pembetulan-pembetulan dan menginisialisasi debugger untuk operasi yang lain. Pilihan ini sekaligus membebaskan pengingat yang telah dialokasikan dan menutup semua berkas, tetapi tidak merubah nilai-nilai perubah. Pilihan ini sangat berguna jika anda ingin menjalankan pilihan

File / OS shell.

Submenu Go to Cursor [F4]

Pilihan ini digunakan untuk memulai / melanjutkan eksekusi program dimulai dari posisi saat eksekusi dimulai sampai tempat kursor berada. Jika kursor berada pada statement yang bersifat executable, seperti spasi atau baris komentar, maka eksekusi akan dilanjutkan sampai statement yang bersifat executable.

Submenu Trace Into [F7]

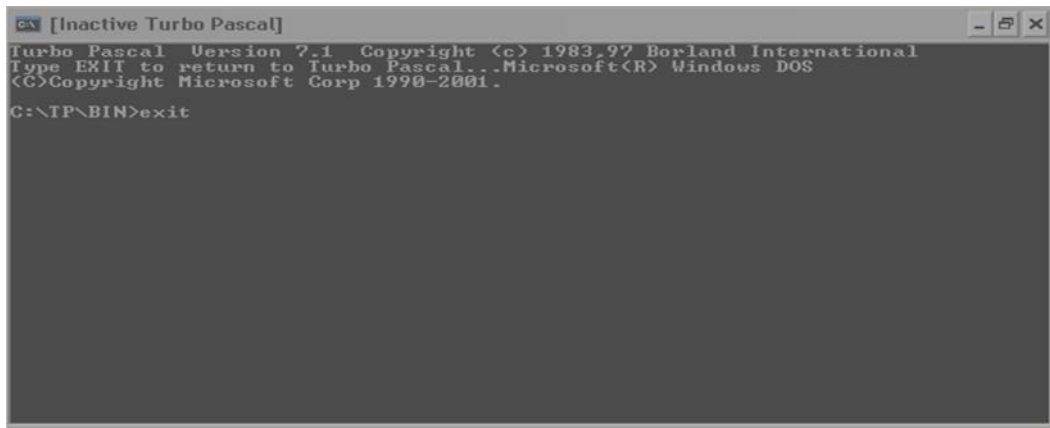
Perintah ini digunakan untuk mengeksekusi baris berikutnya, jika ada pemanggilan ke suatu subrutin, maka pelacakan akan dimulai dari statement pertama dari subrutin tersebut. Sekaligus juga akan memuat berkas-berkas atau unit **Include** jika dimungkinkan.

Submenu Step Over

Step over ini memiliki cara kerja yang hampir sama dengan Trace Into, hanya saja jika statement adalah sebuah prosedur atau fungsi, maka keseluruhan subrutin akan dikerjakan sekaligus dan debugger akan berhenti pada statement sesudah pemanggilan ke subrutin.

Submenu User screen [Alt + F5]

Pada umumnya perintah ini digunakan untuk melihat hasil dari program yang telah compile dan kita run. Pada tampilan ini IDE [Integrated Debugger Environment] berpindah-pindah pada saat anda menjalankan dan mendebug program. Perintah ini juga digunakan oleh File / OS shell.



Menu Compile [Alt + F9]

Perintah ini digunakan untuk mengkompilasi program atau bisa juga kita bilang kalau program yang telah kita buat tadi dengan bahasa pascal di terjemahkan kedalam bahasa komputer agar si komputer ini mengerti apa-apa saja perintah kita.

Submenu Make

Pilihan ini digunakan untuk mengaktifkan perintah make, jika file yang primer telah diberi nama, maka file ini akan dikompilasi, jika tidak maka file yang terakhir yang sedang dimuat di editor akan dikompilasi.. Pilihan ini sangat berguna jika terdapat beberapa program yang secara fisik terpisah tetapi secara logikanya merupakan suatu kesatuan.

Submenu Build

Perintah ini mirip dengan perintah make, hanya saja perintah build ini sifatnya bebas, sedangkan perintah make hanya akan mengkompilasi file-file selain file yang digunakan.

Submenu Destination [Memory]

Perintah ini secara umumnya hanya berfungsi untuk memilih apakah hasil kompilasi akan disimpan dalam cakram atau folder sebagai file yang bertipe .EXE atau hanya di pengingat utama yang akan hilang pada saat anda keluar dari Turbo Pascal.

Submenu Find Error

Dari namanya anda pasti sudah tahu, yah perintah ini digunakan untuk menentukan lokasi kesalahan pada saat program dijalankan [run time error]. Lokasi kesalahan dinyatakan dengan format seg:off, misalnya 2BE0:FFD4. jika anda kembali ke IDE, turbo pascal akan menempatkan kursor pada baris dimana terjadi kesalahan. Jika anda terlanjur memindahkan kursor sembarangan maka anda bisa memindahkan kembali kursor ke tempat kesalahan dengan menekan Ctrl + Q + W.

Submenu Primary File

Perintah ini digunakan untuk menyatakan bahwa file .PAS akan dikompilasi pada saat make [F9] atau build [Alt + C + B] diaktifkan.

Submenu Get info

Tidak salah lagi, perintah ini digunakan untuk menampilkan jendela informasi dari program .PAS yang sedang anda gunakan, termasuk ukuran kode sumber [dalam sumber dan baris], ukuran file .EXE atau .TPU dalam byte dan data

Menu Options

Pilihan pada menu yang satu ini berfungsi mengkonfigurasi untuk mengoptimalkan penggunaan fasilitas-fasilitas pada turbo pascal.

Submenu Compiler

Pilihan ini didalamnya pun masih banyak perintah yang lainnya seperti pengecekan batas, pengecekan tumpukan dll, seperti yang anda lihat pada gambar di atas. contohnya pengecekan batas [Range checking] adalah sama dengan {\$R}. sekarang saya akan coba jelaskan satu persatu agar lebih jelas dari masing-masing penggunaan ini.

Submenu Environment

Pilihan ini digunakan untuk memberitahukan Turbo Pascal dimana bisa ditemukan berkas yang akan dikompilasi, di link dengan dilengkapi dengan menu-menu pertolongan.

Submenu Parameters

Digunakan untuk memberikan parameter baris perintah [Command Line parameters] pada program-program yang akan dijalankan.

Submenu Save Option

Digunakan untuk menyimpan semua pengaturan dalam menu Compiler, Environment, dan Directories dalam berkas konfigurasi [standarnya adalah TURBO.TP]

Digunakan untuk menggerakkan kursor ke posisi breakpoint berikutnya.

Perintah Dasar untuk Menggerakkan Kursor dan hafalkanlah !

| Tombol | Fungsi |
|----------------------------------|---------------------------------|
| Ctrl + S atau panah kiri | Bergerak ke kiri satu karakter |
| Ctrl + D atau panah kanan | Bergerak ke kanan satu karakter |
| Ctrl + A atau Ctrl + panah kiri | Bergerak ke kiri satu kata |
| Ctrl + F atau Ctrl + panah kanan | Bergerak ke kanan satu kata |
| Ctrl + E atau panah naik | Bergerak ke atas satu baris |
| Ctrl + X atau panah turun | Bergerak ke bawah satu baris |
| Ctrl + W | Memutar layar ke atas |
| Ctrl + Z | Memutar layar kebawah |
| Ctrl + R atau PgUp | Kehalaman sebelumnya |
| Ctrl + C atau PgDown | Kehalaman berikutnya |

Submenu Retrieve Options

Digunakan untuk memuat berkas konfigurasi yang sebelumnya disimpan dengan pilihan save.

Perintah Kursor Tambahan

| Tombol | Fungsi |
|--------------------------------|-------------------------------------------|
| Ctrl + U | Membatalkan kegiatan |
| Ctrl + O + I atau Ctrl + Q + I | Meng – On atau meng – Off -kan autoindent |
| Ctrl + P | Awal karakter kendali |
| Ctrl + K + D atau Ctrl + K + Q | Keluar dari editor tanpa menyimpan |
| Ctrl + Q + F | Mencari kata |
| Ctrl + Q + A | Mencari kata dan menggantinya |
| Ctrl + Q + N | Mencari penanda tempat |
| Ctrl + Q + W | Menuju ke posisi kesalahan |
| Ctrl + O + O | Menyisipkan penunjuk kompiler |
| F10 | Mengaktifkan menu utama |
| Ctrl + F1 | Pertolongan bahasa |
| F3 | Memuat berkas |
| Ctrl + O + F | Mengisi On / Off secara optimal |
| Ctrl + Q + [| Pasangan tanda kurung kedepan |
| Ctrl + Q +] | Pasangan tanda kurung kebelakang |
| Ctrl + L | Mengulang pencarian terakhir |
| Ctrl + Q + L | Menyimpan kembali baris |
| Ctrl + K + S atau F2 | Menyimpan dan meneruskan |
| Ctrl + Kn | Mengatur penanda tempat |
| Ctrl + I atau Tab | Tab |
| Ctrl + O + T atau Ctrl + Q + T | Mode Tab |
| Ctrl + O + U | Unindent On / Off |

Tombol-Tombol Kendali Operasi Blok

| Tombol | Fungsi |
|--------------|------------------------------------------|
| Ctrl + T | Menghapus satu kata sebelah kanan kursor |
| Ctrl + Y | Menghapus kata dalam satu baris |
| Ctrl + K + Y | Menghapus blok |

| | |
|--------------|--------------------------------------------------|
| Ctrl + K + V | Memindah blok ke tempat lain |
| Ctrl + K + C | Membuat duplikat blok |
| Ctrl + K + W | Meyimpan blok ke dalam cakram |
| Ctrl + K + H | Menyembunyikan atau menampilkan blok |
| Ctrl + K + R | Membaca blok dari cakram untuk digabung di merge |
| Ctrl + K + T | Menandai satu kata |
| Ctrl + K + P | Mencetak satu blok |
| Ctrl + K + I | Mengident blok |
| Ctrl + K + U | Mengunident blok |
| Ctrl+ K+ B | Menandai awal teks yang akan di Blok (ditandai) |
| Ctrl+ K + K | Menandai akhir teks yang akan di Blok (ditandai) |
| Ctrl + K + V | Memindahkan teks yang di blok (ditandai) |
| Ctrl + K + C | Mem copy teks yang diblok (ditandai) |
| Ctrl+K + H. | Menghilangkan tanda blok |

2. PERINTAH MASUKAN DAN KELUARAN

a. Masukan

Ada beberapa fungsi untuk mendapatkan masukan dari pengguna, diantaranya:

- read
- readln

Perintah digunakan untuk membaca data dari default input - keyboard.

SYNTAX read(x); atau readln(x);

program luas_PersegiPanjang;

{*Program untuk menghitung luas empat persegi panjang*}

Uses wincrt;

var

panjang, lebar, luas : integer;

{*Mendeklarasikan variable panjang, lebar dan luas persegi panjang sebagai integer*}

BEGIN {*Program MULAI*}

panjang := 25;

lebar := 20;

luas := panjang * lebar;

END. {*Program SELESAI*}

Contoh:

```

program luas_PersegiPanjang;
{ Program untuk menghitung luas bangun empat persegi panjang }
uses WinCRT;
var
    panjang, lebar, luas :integer;
{ Mendeklarasikan variable panjang, lebar dan luas persegi panjang sebagai integer}
BEGIN { Program MULAI }
{ Membaca dari data panjang dan lebar persegi panjang, default input unit (papan ketik
dan default format tanpa ganti baris}
    read(panjang);
    readln(lebar);
    (pengisian variabel pertama dan kedua dipisahkan dengan spasi)
    luas:=panjang*lebar;{ Menghitung luas persegi panjang}
END. { Program SELESAI }

```

b. Keluaran

Ada beberapa fungsi untuk menampilkan keluaran di layar, diantaranya:

- write
- writeln

| | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| write('TES'); | Perintah ini akan menampilkan teks string TES ke layar monitor pada posisi kursor sebelum perintah write dilaksanakan. Setelah menyajikan ke layar, kursor pindah ke ujung teks TES pada baris yang sama. |
| writeln('TES'); | Perintah ini akan menampilkan teks TES ke layar monitor pada posisi kursor sebelum perintah write dilaksanakan. Setelah menyajikan ke layar, kursor pindah ke awal baris berikutnya. |
| writeln(Luas); | Perintah ini akan menampilkan nilai Luas hasil proses ke layar monitor pada posisi kursor sebelum perintah write dilaksanakan. Setelah menyajikan ke layar, kursor pindah ke awal baris berikutnya. |
| <pre> program luas_PersegiPanjang; {Program untuk menghitung luas empat persegi panjang. input dibaca dari keyboard dan output ditampilkan ke layar monitor } uses WinCRT; var panjang, lebar, luas : integer; BEGIN { Membaca dari data panjang dan lebar persegi panjang; default input unit dan default format dengan memberikan petunjuk ke pemakai tanpa ganti baris. Setelah membaca data kursor pindah ke baris baru. } write('Panjang = '); readln(panjang); write('Lebar = '); readln(lebar); </pre> | |

```
luas := panjang * lebar;  
{ Menyajikan hasil hitungan luas persegi panjang ke unit default output dan default  
  format.Setelah menyajikan kursor pindah ke baris baru. }  
  writeln('Luas = ', luas);  
END.
```

Catatan:

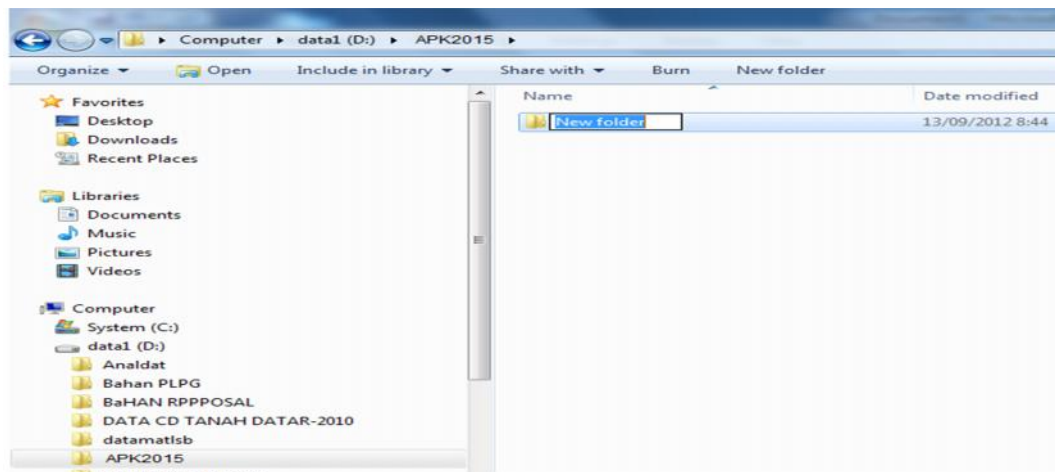
Argumen perintah **writeln** boleh berupa ekspresi.

Contoh :

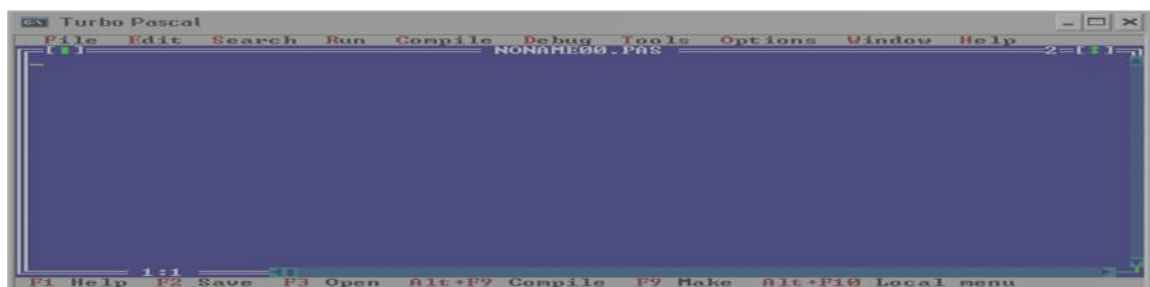
perintah penyajian **writeln('Luas = ', luas);** bisa dikerjakan dengan perintah:
writeln('Luas = ', panjang * lebar);

C. KEGIATAN PRAKTIKUM

1. Pelajari Hand Out Minggu II Perkuliahan ke-3
2. Buat folder di drive D:\APK2015 dengan nama folder Nama anda dengan cara:
 - a. Pada Start menu ambil perintah FILE, seperti pada gambar berikut:

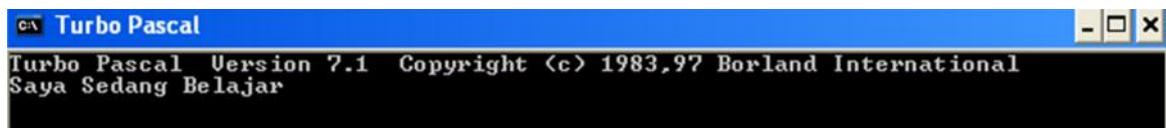


- b. Klik kanan dan tulislah nama anda maksimum dengan 8 huruf, kemudian tekan *Enter*
 - c. Pindahkan direktori kerja ke direktori/folder yang baru anda buat
3. Seterusnya masuklah ke Editor Pascal, seperti penjelasan **B1**, akan tampil seperti berikut:



4. Disini sdr. telah menjalankan editor Turbo Pascal. Pengoperasian editor ini secara umum sama dengan pengoperasian word start.
5. Dengan dasar editing yang telah sdr. ketahui, salinlah program berikut dengan

| | |
|----------------------------------|---------------------------|
| Program Latihan1; | { Bagian Judul Program } |
| Uses WinCRT; | { bagian Deklarasi Unit } |
| Begin | { Awal program } |
| Writeln ('Saya Sedang Belajar'); | { Badan program } |
| End. | { Akhir program } |
6. Running program dengan menekan tombol <Ctrl> + <F9>, perhatikan apa yang muncul. Dilayar tidak ada perubahan, untuk melihat hasil eksekusi tekan tombol <ALT>+<F5>, dilayar akan tampil tulisan



atau



Jika pekerjaan Sdr. Benar, kalau tidak cek kesalahan sdr. dalam menyalin program dan lakukan kembali langkah no.5. Setelah tampil tulisan seperti di atas, tekan kembali Enter

7. Selanjutnya, tambahkan tulis Readln; sebelum tulisan End. ,kemudian running program dengan menekan tombol <Ctrl> + <F9>. Perhatikan apa yang tampil.
8. Selanjutnya, tambahkan Clrscr; sesudah kata Begin , kemudian running program dengan menekan tombol <Ctrl> + <F9>. Perhatikan apa yang tampil.
9. Simpanlah program ke directory sdr. dengan nama Prakti1a.pas , dengan cara Caranya:
 - a. Tekan tombol <Alt><F>
 - b. Kemudian sorot kalimat **Save As ...**, dengan menggunakan tombol panah ke bawah, kemudian tekan **Enter**, maka akan muncul dilayar:
 - c. Ketik dalam kotak yang tersedia: **D:\APK2015\nama anda\Prakt01a**, kemudian **Enter**. Maka program sdr. tersimpan pada directory dengan nama anda dan nama filenya Prakti1a.pas. (seperti contoh).



10. Selanjutnya salinlah program berikut ini (perhatian teori dasar di atas):

```
program luas_PersegiPanjang;  
Uses crt;  
var  
    panjang, lebar, luas : integer;  
BEGIN {Program MULAI}  
    panjang := 25;  
    lebar := 20;  
    luas := panjang * lebar;  
END.
```

11. Running program, perhatikan apa yang muncul, lihat hasil dengan menekan tombol <ALT> + <F5>. Hasilnya belum ada, karena belum ada perintah output. Sekarang tambahkan tulisan pada baris sebelum kata END.

Writeln (luas);

Kemudian running program, perhatikan apa yang muncul, cek apakah hasilnya sudah benar.

Agar bersifat User Friendly, **Ganti Writeln(luas);** dengan
Writeln(' luas dari 25 x 20 adalah ',luas);
Readln;

Sekarang ganti angka 25 dengan 50, apakah hasilnya tatap benar ?, kenapa demikian ?

Simpan program dengan nama prakt01b., dengan seperti yang dijelaskan pada kegiatan ke - 8.

12. Salinlah program berikut ini:

```
program luas_PersegiPanjang;  
uses crt;  
var  
    panjang, lebar, luas : integer;  
BEGIN  
    Clrscr;  
    write('masukan harga Panjang = '); readln(panjang);  
    write('masukkan harga Lebar = '); readln(lebar);  
    luas := panjang * lebar;  
    Writeln('Luas = ',luas);  
    Readln;  
END.
```

Running program di atas , kemudian pelajari bagaimana hasilnya. Simpan program ini dengan nama prakt01c.

13. Dengan dasar editing yang telah sdr. ketahui, ketiklah program berikut ini:

| | |
|----------------------|--------------------------|
| Program Gerak_Balok; | { Bagian Judul Program} |
| Uses CRT; | { bagian Deklarasi Unit} |
| Const | |

| | |
|--------------------------------------|--------------------------------|
| Grafitasi = 10 | { bagian deklarasi Constanta } |
| Var | { bagian deklarasi variabel } |
| F,us,m,percepatan | : real; |
| Begin | { Awal program } |
| Clrscr; | { Badan program } |
| Gaya Tarik := 50; | { Bagian masukan } |
| massa := 5; | { Bagian masukan } |
| Percepatan := (F - m*G*us)/m; | { Proses perhitungan } |
| Writeln(' Percepatan: ',percepatan); | { Bagian Pencetakan } |
| Koeff_Gesekan := 0.3; | { Bagian masukan } |
| End. | { Akhir program } |

14. Simpanlah program ke directory sdr. dengan nama Prakt01d,
15. Compile program di atas dengan menekan tombol F9, bila keluar pesan berwarna merah pada monitor bagian atas disebut dengan kesalahan sintaks.
16. Tambahkan tanda titik koma (;) di belakang angka 10 (pada bagian konstanta, dan kemudian tekan lagi tombol (F9). Lihat bagaimana hasilnya ?
17. Disini terjadi lagi kesalahan sintaks pada bagian deklarasi dan benarkan program di atas dengan cara mengganti kata-kata:

| Yang telah ditulis pada pseudocode | Diganti dengan |
|------------------------------------|----------------|
| <i>Gaya_Tarik</i> | <i>F</i> |
| <i>Koeff_Gesekan</i> | <i>us</i> |
| <i>massa</i> | <i>M</i> |
| <i>Grafitasi</i> | <i>G</i> |

18. Kemudian tekan lagi tombol F9. Bila masih keluar pesan kesalahan sintaks, chek ulang program sdr. dan bila semua sesuai dengan instruksi yang diberikan dalam petunjuk ini tanyakan kepada pembimbing sdr. kenapa masih ada kesalahan.
19. RUN program ini dengan menekan tombol <Ctrl><F9> sekaligus. Amatilah apa yang terjadi.
20. Untuk melihat bekas tampilan tekan tombol <Alt><F5>.
21. Periksa hasil perhitungan keluaran program. Hasilnya akan salah, kesalahan ini disebut **kesalahan logika**. Benarkan program ini dengan memindahkan
 $us := 0.3$ ke atas perintah $Percepatan := (F - m * G * us) / m;$
22. RUN program seperti yang dilakukan pada langkah ke-6.
23. Konsultasikan pada pembimbing sdr. tentang kebenaran program yang sudah dibuat. Bila selesai tekan tombol F2 untuk menyimpan pseudocode yang telah benar tersebut.
24. Salinlah program **berikut**:

| | |
|----------------------|--------------------------------|
| Program Gerak_Balok; | { Bagian Judul Program } |
| Uses Wincrt; | { bagian Deklarasi Unit } |
| Const | |
| G =10; | { bagian deklarasi Constanta } |
| Var | { bagian deklarasi variabel } |

```

F,us,m,percepatan,s, usaha: real;
Begin                                     { Awal program}
  Clrscr;                               { Badan program}
  Writeln(`Masukan Angka Anda, Kemudian Enter ! `);
  Write(`Gaya Tarik :      ');readln(F );      {Bagian masukan }
  Write(`Massa Benda : '); readln(m);           {Bagian masukan }
  Write(`Koefesien Gesekkan : '); readln(us);   {Bagian masukan }
  Write(`Perpindahan      : '); readln(s);       {Bagian masukan }
  Percepatan := (F - m*g*us)/m;                  {Proses perhitungan}
  Usaha := (F - m*g*us)*s/m;                     {Proses perhitungan}
  Writeln(`Percepatan      : `,percepatan);      {Bagian Pencetakan}
  Writeln(`Usaha oleh gaya F : `,usaha);          {Bagian Pencetakan}
  Readln;
End.

```

25. Simpan program ini (Save As..), dengan prakt01e.pas. Kemudian RUN program tersebut, masukkan angka sesuai dengan variabel yang diminta, amati apa yang terjadi. Ambil kesimpulan tentang fungsi dan cara pemakaian perintah read, readln, write dan writeln dan write.
26. Buatlah program yang dapat digunakan untuk menghitung volume dan luas selimut benda, dimana badannya berbentuk selinder, salah satu ujungnya berbentuk bola dan pada ujung lainnya berbentuk kerucut. (ingat tugas pendahuluan no. 6). Simpan program ini dengan nama Prakt01f.pas.
27. Anda masih punya waktu, chek kebenaran tugas awal sdr. no. 7.

D. TUGAS AWAL (Dikumpulkan Sebelum Praktikum).

1. Apa perbedaan antara kesalahan sintaks dengan kesalahan logika.
2. Jelaskan yang dimaksud dengan pemograman terstruktur
3. Secara garis besar, struktur pemograman dengan Pascal terdiri dari 3 bagian utarna, jelaskan bagian-bagian tersebut dengan contoh.
4. Apa fungsi perintah Read, readln, write, writeln.
5. Penamaan suatu variabel harus mengikuti peraturan tertentu, sebutkan aturan tersebut.
6. Buatlah diagram alir (flowchart) dan program untuk kegiatan praktikum no.26
7. Buatlah diagram alir dan pseudocode dalam bahasa Pascal untuk menghitung volume balok. Simpan dengan nama **Balok**

MODUL PRAKTIKUM II

A. Kompetensi yang diharapkan

Setelah kegiatan berlangsung diharapkan Praktikan mampu:

1. Membuat program sederhana yang user friendly dan mudah ditelusuri
2. Memahami konsep pemrograman menggunakan *pseudocode* Turbo Pascal
3. Memahami pemakaian operator aritmatika pada bahasa Pascal
4. Memahami pengulangan proses dengan perintah
 - a. For .. To .. Do,
 - b. For ... downto .. do,
 - c. While .. do
 - d. Repeat Until.

B. Kegiatan Sebelum Praktikum

Pelajari Bahan ajar perkuliahan Minggu II dan III. Pada Praktikum ini Praktikan akan diperkenalkan pada lingkungan kerja pengembangan program terpadu serta operasi aritmatika dan perulangan.

1. Bekerja pada lingkungan kerja FreePascal

- a. Membuat berkas baru
 - **Alt+F -> N**
- b. Berpindah direktori kerja
 - **Alt+F -> C**
- c. Menyimpan berkas
 - **Alt+F -> S**
 - **F2**
- d. Menyusun program (compiling)
 - **Alt+C -> C**
 - **Alt+F9**

2. Struktur kode program PASCAL

- a. Kepala
Deklarasi variabel, konstanta dan fungsi/prosedur

Pernyataan peubah (variable)

```
var
panjang : integer;
Lebar   : integer;
luas, keliling : integer;
```

Pernyataan nilai tetap (constants)

```
const
KONSTANTA1 = 'abcd';
angkaku    = 3.14;
```

- b. Badan

```
Begin
...
... { KODE PROGRAM LAINNYA }
...
```

- c. End.

3. Operator Aritmatika

Operator-operator aritmatika dapat dilihat pada Tabel di bawah ini:

| Operator | Operasi |
|----------|-------------|
| * | Perkalian |
| / | Pembagian |
| + | Penjumlahan |
| - | Pengurangan |
| mod | Sisa bagi |
| div | Hasil bagi |

4. Urutan evaluasi operator dapat dilihat pada tabel dibawah ini. Operator pada urutan evaluasi yang sama dievaluasi dari kiri ke kanan. Untuk mendahulukan evaluasi, beri tanda kurung () pada evaluasi yang ingin didahulukan.

| Operator |
|----------|
| *, / |
| +, - |
| mod, div |

5. Perulangan

- a. Perintah for .. to .. do

Ada beberapa macam perulangan (looping) dalam bahasa PASCAL, salah satunya adalah for-to-do.

Contoh kalimat :

```
Program fortodo;
Uses crt;
var
    i : integer;
Begin
    for i := 1 to 5 do
        Begin
            writeln('ini yang ke ', i, '. Aku bangga jadi Mahasiswa Fisika.');
```

```
        End;
    End.
```

- b. Perintah for .. downto .. do

```
Program fordowtodo;
Uses crt;
var
    i : integer;
Begin
    for i := 5 to 1 do
        Begin
            writeln('ini yang ke ', i, '. Aku bangga jadi Mahasiswa Fisika.');
```

```
        End;
    End.
```

c. Perintah While .. do

Bagaimana kalau mau pakai while? Perhatikan contoh:

```
Program whiledo;  
Uses Crt;  
var i: Integer;  
Begin  
  Clrscr;  
  i:=0;  
  while i < 5 do  
  begin  
    writeln('ini yang ke ', i, ' Aku bangga jadi Mahasiswa Fisika.');    i:=i+1;  
  end;  
End.
```

Apa perbedaan mendasar while dengan for? While dibutuhkan counter diawal perintah (pada program ini adalah i). Jadi, baris ke-9 dan 10 akan dieksekusi (dijalankan) apabila nilai $i < 5$. Coba lihat baris ke 8, di sana ada "begin" dan "end;" sebagai pasangannya (dibaris ke 11).

d. Perintah Repeat ... Until

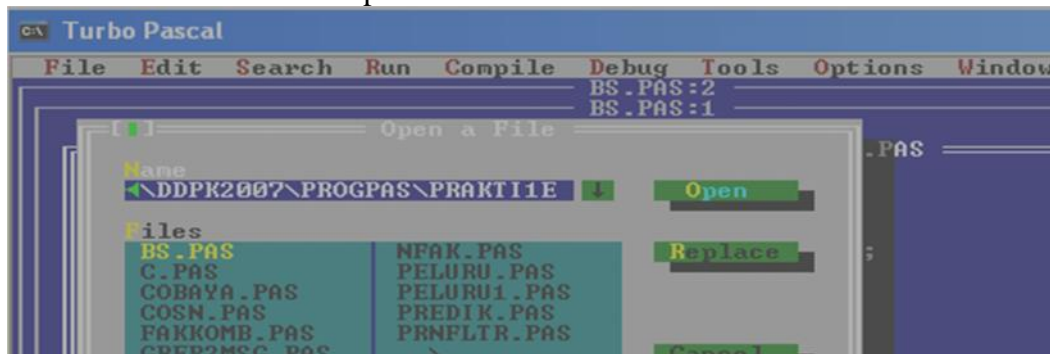
Repeat ... until disebut juga dengan invinitive loop. Sekarang kita coba dengan repeat.

```
Program Repeat_Until;  
Uses Crt;  
var pilih: char;  
Begin  
  i:=0;  
  repeat  
    i:=i+1;  
    writeln('ini yang ke ', i, ' Aku bangga jadi Mahasiswa Fisika.');  until i < 5;  
End.
```

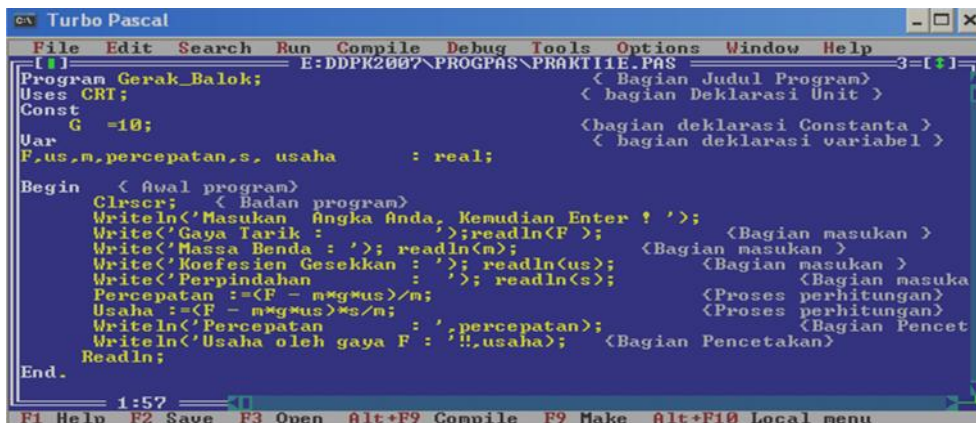
Apa outputnya? Hanya 1, "ini yang ke 1, Aku bangga jadi Mahasiswa Fisika ". Kenapa bisa, perhatikan pada baris ke 9 tidak memenuhi persyaratan $i < 5$? Memang tidak memenuhi persyaratan itu, tapi ini adalah keistimewaan repeat. Ia akan mengeksekusi sekali dulu baru dicek persyaratan yang ada dibawah. Agar program mengeksekusi lebih dari 1, gantilah, persyaratan $i < 5$ dengan $(i > 5)$ ” **ini yang ke**”, **Aku bangga jadi Mahasiswa Fisika**” akan dieksekusi sebanyak 6 kali. Untuk aplikatifnya, praktekan pengetahuan di atas, dalam kegiatan Praktikum berikut ini.

C. Kegiatan Praktikum

1. Masuklah ke editor Pascal seperti yang telah dijelaskan pada Praktikum 1
2. Sebagai ulangan coba ketika program berikut ini:
Program Ulangan;
Uses crt;
var nama: string;
Begin
Write('Masukkan nama anda: '); readln(nama);
Clrscr;
Write('Hi ', nama, 'apa kabar?');
End.
3. Running program di atas, perhatikan dan pelajari cara pemakaian perintah input (read) dan output (write) secara bersamaan. Pada baris ke-3 dideklarasikan variabel nama dengan tipe string. Karena berupa string, variabel nama bisa menampung input dari user berupa kata dan angka. Simpan program dengan nama Prakt02a.
4. Panggil kembali pseudocode **Pascal Prakt01E.pas**, dengan cara
 - a. Tekan Tombol **F10**, kemudian tekan huruf **F**, atau **Tombol F3**, maka pada monitor akan muncul seperti berikut:



- b. Pada kotak yang tersedia tulis:
Sebagai contoh tulis **D:\APK2015>Nama_Anda\Prakt01E.pas**, (perhatikan sesuaikan dengan komputer dan nama direktori atau folder yang sdr. buat saat Praktikum I), kemudian tekan Enter.
5. Setelah itu akan muncul tampilan seperti berikut:



4. Perbaiki format keluaran dengan mengganti perintah **Writeln('Percepatan : ' ,
percepatan);**, dengan **Writeln('Percepatan : ',percepatan:8:4);** kemudian kemudian
simpan dengan nama **Prakt02b**, dengan menggunakan perintah **SAVE AS**. Seperti
telah dilaksanakan pada Praktikum I
5. Running (<Ctrl><F9>) dan perhatikan perbedaan tampilan hasil running sebelum ini
atau Running program **Prakt01E** dengan Running **Prakt02b**.
6. Format yang juga dapat dilakukan pada **Writeln('Usaha oleh Gaya F : ',usaha);**
dengan menggantinya dengan **Writeln('Usaha oleh Gaya F : ',Usaha: 8:4);**.
Format ini hanya berlaku untuk Tipe Data bilangan pecahan (real).
7. Aturlah letak tampilan masukkan dengan perintah **GotoXY(X,Y) ;**
Contoh

```
gotoXY(5,11); write('Gaya Tarik = '); read(F);
gotoXY(5,12); write('Massa Benda = '); read(m);
gotoXY(5,13); write('Koefisien Gesekan = '); read(us);
```

8. Simpan program ini dengan nama **Prakt02c.**, kemudian Running, perhatikan
tampilannya
9. Kemudian atur pula tampilan keluaran dengan perintah:
gotoXY(5,14); Writeln('Percepatan : ',percepatan:8:4);
gotoXY(5,15); Writeln('Usaha oleh Gaya F : ',Usaha: 8:4);
(Tarik kesimpulan fungsi perintah **GOTOXY(x,y)**).
10. Salinlah program berikut:
Program fortodo;
Uses crt;
var
 i : integer;
Begin
 for i := 1 to 5 do
 Begin
 writeln('ini yang ke ', i , '. Aku bangga jadi Mahasiswa Fisika.');
11. Runninglah, program di atas, perhatikan hasilnya. Apabila ada hasilnya yaitu
munculnya tulisan
 "ini yang ke 1. Aku bangga jadi Mahasiswa Fisika, sebanyak 5 kali, simpan
program di atas dengan nama **Prakt02d**.
12. Gantilah perintah **For .. to .. do**, dengan perintah **For ... downto .. do**, running
program, pelajar perbedaannya.
13. Lakukan juga pergantian perintah pengulangan yang lainnya yaitu **While ... do**
14. Lakukan juga pergantian perintah **repeat ... until**. (Perhatikan teori di atas).
Simpan dengan nama file **Prakt02E**.
15. Berdasarkan diagram alir berikut ini, buat psedocode dengan bahasa Pascalnya

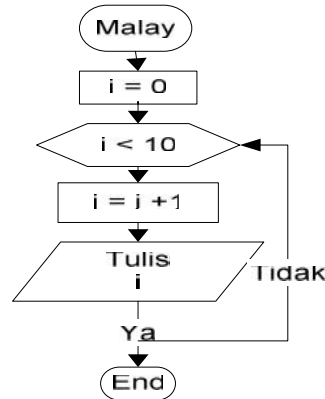


Diagram alir Prakt02F.

Simpan program di atas dengan nama **Prakt02F**. Sebaiknya program dibuat dengan menggunakan perintah pengulangan While ... do.

16. Sebuah peluru bermassa **m** ditembakkan pada titik awal (0,0) dengan kecepatan awal **vo** dan elevasi **theta**. Buatlah program untuk menghitung posisi, kecepatan arah sumbu-x (**Vx**) dan sumbu-y (**Vy**), serta momentum setiap saat dalam bentuk tabel, dengan ketentuan sebagai berikut:
 - a. Kecepatan awal (**vo**), sudut elevasi (**theta**), dan massa (**m**) dimasukkan melalui keyboard setelah program di RUN
 - b. Keluaran dibuar dalam bentuk tabel
 - c. Judul program “Peluru”
 - d. Simpan file program dengan nama **Prakt02G**
 - e. Ingat eksekusi pengulangan dengan kounter waktu mulai dari 0 sampai dengan waktu peluru sampai ke tanah atau ketinggian peluru sama dengan nol.

D. Tugas Awal (dikumpulkan sebelum Praktikum)

1. Jelaskan fungsi perintah dan sintaks penggunaan write, writeln, read, readln, gotoxy
2. Jelaskan fungsi perintah dan sintaks penggunaan wherex dan wherey
3. Jelaskan dengan menggunakan diagram alir fungsi struktur control
 - a. For .. to .. do
 - b. For .. downto .. do
 - c. Repeat ... until
 - d. While .. do
4. Buat beberapa program untuk menjumlahkan angka dari 1 sampai 10 menggunakan perulangan While ... do , berdasarkan diagram alir **Prakt02F**
5. Buatlah algoritma, diagram alir dan pseudocode dalam bahasa Pascal untuk Praktikum langkah 16 atau Prakt02G

MODUL PRAKT0KUM III

A. Kompetensi yang diharapkan

Setelah kegiatan berlangsung diharapkan Prakt0kan mampu:

1. Membuat program sederhana yang user friendly dan mudah ditelusuri
2. Memahami Konsep Pemrograman menggunakan pseudocode pada bahasa Turbo Pascal
3. Memahami pemakaian ekspresi boolean
4. Memahami pemilihan kondisi dengan perintah If .. Then
5. Memahami pemilihan kondisi dengan perintah Case .. Of
6. Memahami pemilihan kondisi dengan perintah Case .. Of

B. Kegiatan Sebelum Prakt0kum

Perlajarilah bahan perkuliahan pada pertemuan ke-5 dan ringkasan materi berikut secara menyeluruh

1. IF..THEN.. ELSE

Bentuk umum :

```
if (kondisi) then aksi ;  
    { Jika kondisi benar maka aksi akan dilaksanakan dan sebaliknya }  
if (kondisi) then aksi1 else aksi2 ;  
    { Jika kondisi benar maka aksi1 dilaksanakan, jika kondisi salah  
      maka aksi2 akan dilaksanakan }  
if (kondisi1) then aksi1 else  
    if (kondisi2) then aksi2 else  
    .....  
    if (kondisi-n) then aksi-n ;  
    { Jika kondisi1 benar maka aksi1 dilaksanakan tanpa membaca  
      kondisi2 dan seterusnya. Jika kondisi1 salah maka aksi2  
      dilaksanakan tanpa membaca aksi3 dan selanjutnya dan seterusnya }  
if (kondisi1) then aksi1 ;  
    if (kondisi2) then aksi2 ;  
    .....  
    if (kondisi-n) then aksi-n ;  
    { Masing-masing kondisi akan dibaca dan aksi akan tetap dilaksa-  
      kan. Jadi masing-masing struktur terpisah satu dengan yang lain }
```

Bentuk struktur umum If – Then adalah sebagai berikut :

Contoh 1;

Contoh Program :

```
Uses Crt;
Var
Nilai : Real;
Begin
    Write ('Jumlah Nilai :'); Readln (nilai); { Pemasukan data }
    If nilai > 60 Then { Variabel nilai }
        Writeln('Lulus') { jika nilai besar dari 60 }
    Else
        Writeln('Tidak lulus'); { nilai kecil dari 60 }
End.
```

2. Kombinasi dari Ekspresi Boolean (AND, OR, dan NOT)

Kata kunci AND, OR dan NOT diperlukan apabila kita ingin mengeksekusi blok program yang memerlukan lebih dari satu kondisi.

a. Pernyataan AND

Pernyataan AND akan dieksekusi jika kedua kondisi dalam keadaan TRUE.

```
if <Ekspresi Bool_1> and <Ekspresi Bool_2> then
Begin
:
:
End
```

Contoh:

Bila ingin menyatakan harga $30 < x < 40$, dapat ditulis dengan pernyataan :

```
If (X > 30 ) AND X < 40) then
```

b. Pernyataan OR

Pernyataan OR akan dieksekusi bila salah satu kondisi dalam keadaan TRUE,

```
if <Ekspresi Bool_1> or <Ekspresi Bool_2> then
```

```
Begin
:
:
End
```

Contoh:

Bila ingin menyatakan harga $x < 30$ atau $x > 40$, dapat ditulis dengan pernyataan :

```
If (x < 30 ) OR x .> 40) then
```

c. Pernyataan NOT

NOT menggantikan TRUE ke FALSE dan sebaliknya (vice versa)

```
if not <Ekspresi Boolean> then
begin
:
:
End
```

Pernyataan ini digunakan untuk menyatakan kecuali atau memilih kondisi saja yaitu betul atau salah

Contoh : Bila tidak **Lulus** tentu **Gagal**

3. Case - Of

Bentuk umumnya :

| | | | | |
|-------------|---------------------------|-------------|-------------|--------------------------|
| <i>Case</i> | <i>var.pilih of</i> | <i>atau</i> | <i>Case</i> | <i>var.pilih of</i> |
| | <i>Pilih1 : aksi1 ;</i> | | | <i>pilih1 : aksi1 ;</i> |
| | <i>Pilih2 : aksi2 ;</i> | | | <i>pilih2 : aksi2 ;</i> |
| | <i>..... ;</i> | | | <i>..... ;</i> |
| | <i>pilih-n : aksi-n ;</i> | | | <i>pilih-n : aksi n;</i> |
| | | | | <i>else aksi-n+1</i> |
| | <i>end;</i> | | | <i>end;</i> |

C. Kegiatan Praktikum

1. Masuklah ke editor Pascal seperti yang telah dijelaskan pada Praktikum 1
2. Edit dan pelajari program dan hasil running contoh 1 di atas, telah selesai simpan dengan nama **Prakt03a**
3. Kerjakan dan uji tugas pendahuluan no. 1 yang sdr. Buat, simpan program dengan nama **Prakt03b**
4. Kerjakan dan uji tugas pendahuluan no. 2 yang sdr. Buat, simpan program dengan nama **Prakt03c**
5. Kerjakan dan uji tugas pendahuluan no. 3 yang sdr. Buat, simpan program dengan nama **Prakt03d**
6. Kerjakan dan uji tugas pendahuluan no. 4. dan simpan program dengan nama **Prakt03e**.

D. Tugas Pendahuluan

Buatlah diagram alir dan program (pseudocode Pascal Turbo) soal di bawah ini dengan tampilan sebagus mungkin (bila perlu gunakan perintah Window, Textcolor dll). Jumlah iterasi sesuai dengan input dari keyboard.

1. Buatlah diagram alir dan program untuk menyeleksi suatu bilangan dengan ketentuan sebagai berikut :
 $0 \leq \text{nilai} < 30$: Nilai rendah
 $30 \leq \text{nilai} < 60$: Nilai sedang
 $60 \leq \text{nilai} \leq 100$: Nilai tinggi
2. Buatlah diagram alir dan program menu yang dapat digunakan untuk menghitung luas kulit dan volume benda sebanyak **n** buah dengan benda berbentuk :
 - a. Persegi panjang
 - b. Lingkaran
 - c. Segitiga
3. Bila kita punya persamaan $ax^2 + bx + c = 0$, buatlah pseudocode dalam bahasa Pascal Turbo
4. Buatlah program untuk menentukan apakah bilangan yang ditemukan adalah bilangan ganjil atau genap. Bilangan yang dicari mulai dari 1 sampai 50, dilacak melalui suatu pengulangan atau looping.

MODUL PRAKTIKUM IV

A. Kompetensi yang diharapkan

Setelah kegiatan berlangsung diharapkan Praktikan mampu:

1. Membuat program sederhana yang user friendly dan mudah ditelusuri
2. Memahami konsep pemrograman menggunakan pseudocode pada bahasa Turbo Pascal
3. Memahami perulangan tersarang (Nested Loop) dan penyeleksian kondisi tersarang (Nested IF)
4. Memahami struktur control kombinasi (Nested Control Structure)

B. Kegiatan Sebelum Praktikum

Perajarilah bahan perkuliahan pada pertemuan IV dan V dan ringkasan materi berikut secara menyeluruh

Perulangan For tersarang adalah perulangan For yang berada pada perulangan yang lainnya. Perulangan yang lebih dalam akan diproses terlebih dahulu sampai habis (sampai kepada kondisi yang telah ditentukan), kemudian perulangan yang lebih luar baru akan bertambah kepada iterasi berikut sampai dengan batas kondisi yang telah ditentukan :

```
Program Contoh1;  
Uses WinCrt;  
Var  
a,b : integer;  
Begin  
  For a := 1 to 3 Do  
    Begin  
      For b := 1 to 4 Do Write (a :4,b:2);  
      Writeln;  
    End;  
  End.
```

1. **Perulangan While – Do tersarang** (nested While - Do) merupakan perulangan While – Do yang satu di dalam perulangan While – Do yang lainnya.

```
Program Contoh2;  
Uses Crt;  
Var a, b : Integer;  
Begin  
  ClrScr;  
  a:=1; b:=1;  
  While a < 6 Do { loop selama a masih lebih kecil dari 6 }  
    Begin  
      a := a+1;  
      While b < 4 Do { loop selama b masih lebih kecil dari 4 }  
        Begin  
          Write (a:3,b:2);  
          b:=b+1;
```

```

End;
End;
Readln;
End.

```

2. **If tersarang (nested If)** merupakan bentuk dari suatu statement If berada di dalam lingkungan statemen If yang lainnya. Bentuk statement If tersarang sebagai berikut :

| | | |
|----------------------------------------------------------------------------------|------|--------------------------------------------------------------------------------------------|
| <pre> If kondisi1 Then If kondisi2 Then statemen1 Else statemen2; </pre> | atau | <pre> If Kondisi1 Then Begin IF kondisi2 Then statemen1 Else statemen2 End; </pre> |
|----------------------------------------------------------------------------------|------|--------------------------------------------------------------------------------------------|

Contoh 3:

```

Program If_then_Else;
Uses
WinCrt;
var Nilai : Byte;
Begin
    clrscr;
        write('Baca nilai :');readln(Nilai);
    if nilai>=80 then write('Nilai = A') else
        if nilai>=65 then write('Nilai = B') else
            if nilai>=41 then write('Nilai = C') else
                if nilai>=26 then write('Nilai = D') else
                    write('Nilai = E');
        readln;
    End.

```

3. **Kombinasi pemakaian if, while dan case of.** Merupakan teknik pemakaian struktur kontrol secara bersamaan.. Pelajarilah contoh program berikut:

```

program Tebak;
Uses WinCrt;
var
    pilihan: char;
acak, angka, awal: integer;

begin
    randomize; { start pembangkit random }
    while upcase(pilihan) <> 'N' do { while kita akan mulai main }
        begin
            acak := random(100) + 1; { membangkit angka random untuk permainan }
            awal := 6; { set angka awal }

```

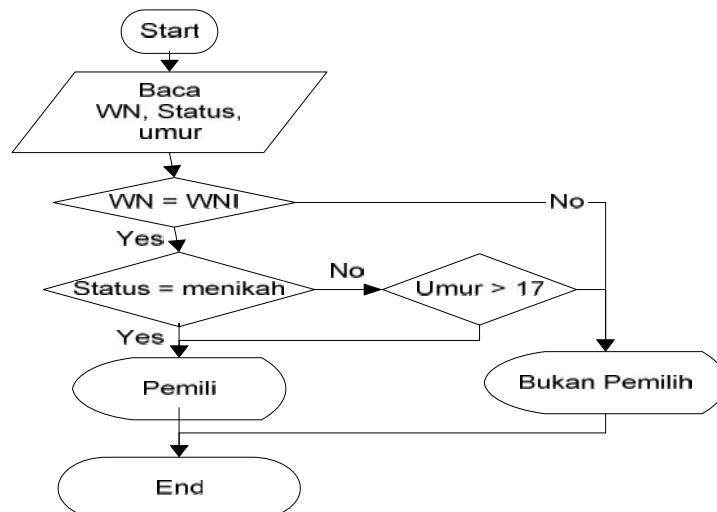
```

angka := 1000;
writeln('Saya Punya Bilangan 1 dan 100. Coba Tebak Bilangan itu ');
while (awal > 0) and (angka <> acak) do
begin
  readln(angka);
  if angka = acak then
    writeln('Selamat! Kamu mendapatkan angkanya !')
  else
    begin
      awal := awal - 1;
      if awal = 0 then
        writeln('Maaf, kesempatan habis ', ' Bilangan Saya adalah ', acak, '.')
      else
        if angka > acak then
          writeln('Kurang dari itu. (' , awal, ' kesempatan lagi)')
        else
          writeln('Lebih dari. (' , awal, ' kesempatan lagi)');
    end;
  end;
  writeln;
  writeln('Kamu mau main lagi? (Y/N)');
  readln(pilihan);
end;
end.

```

C. Kegiatan Praktikum

1. Salinlah program-program contoh 1, contoh 2 dan contoh 3 di atas, simpan dengan nama masing-masing Prakt04a, Prakt04b dan Prakt04c. Kemudian pelajari keluarnya.
2. Gambar berikut berikr adalah diagram alir utama cara penyeleksian calon pemilih dalam suatu pemilihan umum di Indonesia untuk seorang calon pemilih.



Berdasarkan diagram alir di atas buatlah program dalam bahasa Pascal untuk menyeleksi banyak pendaftar. Pemasukkan data selesai, apabila pada bagian nama pendaftar ditulis kata “habis”. Simpan file dengan nama **Prakt04d**

3. Berdasrakan contoh-contoh di atas, buatlah program untuk menghitung volume dan luas benda berupa piramida dan selinder dengan ketentuan keluaran sebagai berikut:

Nama file : Prakt04e

Nama program : Piramida_dan_Selinder

Masukannya paramida : Panjang, lebar alas, tinggi

Masukannya selinder : jari-jari alas dan panjang

Proses perhitungan untuk setiap benda akan berakhir apabila pada masukan data ditulis kata "selesai". Perlu diingat adalah ini merupakan kombinasi perintah Looping dan penyeleksian kondisi.

D. Tugas Pendahuluan

1. Jelaskan peranan algoritma dalam perancangan pemogram komputer
2. Buatlah diagram alir dan pseudocode dengan bahasa pascal tugas Praktikum ke-2
3. Buatlah diagram alir dan pseudocode dengan bahasa pascal tugas Praktikum ke-3
4. Buatlah diagram alir dan pseudocode dengan bahasa pascal tugas Praktikum ke-4

MODUL PRAKTIKUM V

A. Kompetensi yang diharapkan

Setelah kegiatan berlangsung diharapkan praktikan mampu:

1. Membuat program sederhana yang user friendly dan mudah ditelusuri
2. Memahami konsep pemrograman menggunakan pseudocode pada bahasa Turbo Pascal
3. Memahami pembuatan program secara procedural dengan menggunakan program pembantu dalam bentuk procedure
4. Memahami perbedaan Variabel global dan variable lokal
5. Memahami perbedaan parameter formal dan parameter aktual
6. Memahami perbedaan parameter nilai dan parameter variabel
7. Memahami Procedure memanggil procedure lain

B. Kegiatan Sebelum Praktikum

1. Pelajari Bahan Ajar yang relevan

Pemrograman modular adalah teknik yang digunakan untuk menulis program besar. Program ini dibagi menjadi bagian-bagian kecil. Setiap bagian disebut modul, dan melakukan satu tugas.

Contoh tugas modul mungkin untuk:

- a. menampilkan menu pilihan
- b. mencetak hasil
- c. menghitung nilai rata-rata
- d. memilah data ke dalam kelompok-kelompok

Modul ini dikenal melalui namanya, dan terdiri dari set bagian-bagian program dikelompokkan menggunakan kata kunci BEGIN dan END. Modul (kelompok pernyataan) yang dijalankan ketika Anda mengetikkan nama modul. Pascal menggunakan tiga jenis modul. Dua yang pertama disebut PROCEDURE, FUNCTION lain

- a. Prosedur sederhana tidak menerima argumen (nilai atau data) ketika prosedur dijalankan (disebut).
- b. prosedur Kompleks menerima nilai-nilai untuk bekerja dengan ketika mereka dieksekusi.
- c. Fungsi, ketika dijalankan, mengembalikan nilai (yaitu, menghitung jawaban yang dibuat tersedia untuk modul yang menginginkan jawaban). Prosedur membantu desain dukungan program terstruktur, dengan memungkinkan pengembangan independen dari modul. Prosedur dasarnya sub-program.

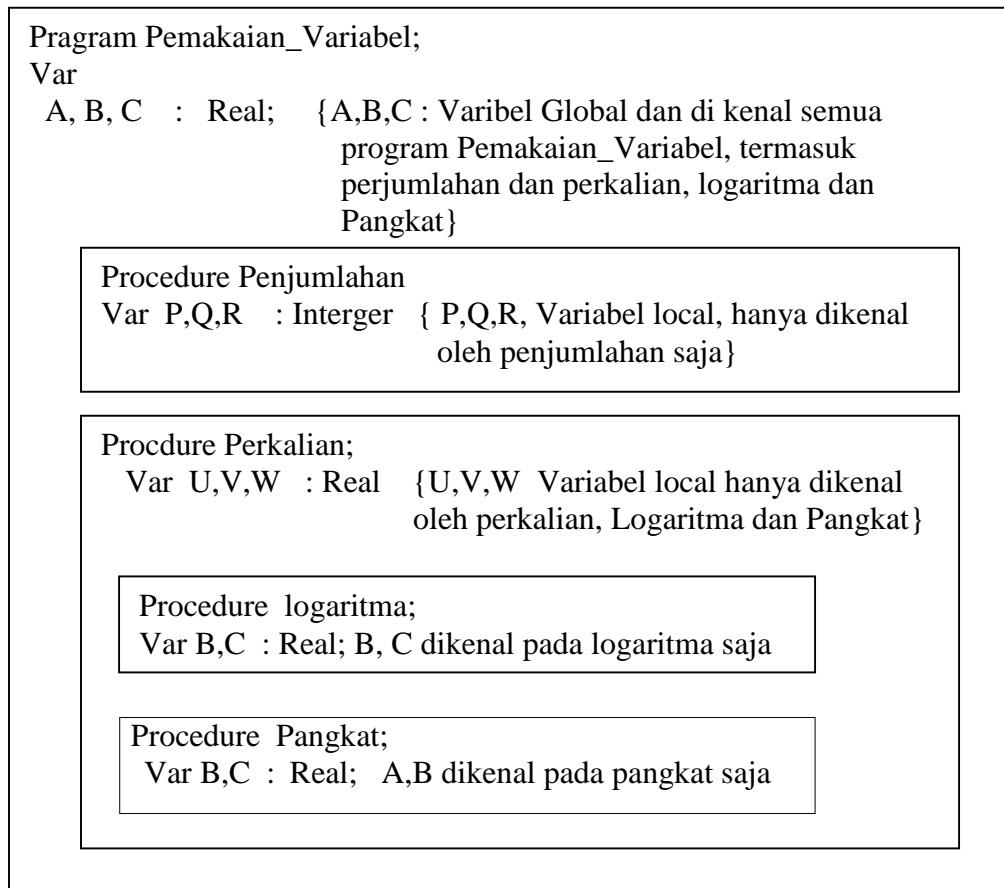
2. Perhatikan struktur procedure ini

```
procedure nama-prosedur [(deretan-parameter)];  
[bagian deklarasi konstanta, tipe, variabel, dan prosedur/fungsi]  
begin  
    [statemen-statemen dalam badan prosedur]  
end;
```

3. Program variable global dan variable lokal, yang dapat dipetakan sebagai berikut:

Suatu prosedur dapat mendeklarasikan variabel itu sendiri untuk bekerja dengan. Variabel ini milik prosedur di mana mereka dinyatakan. Variabel dideklarasikan di dalam prosedur

yang dikenal sebagai lokal. Variabel lokal dapat diakses di mana saja antara begin dan kata kunci akhir pencocokan prosedur. Program berikut mengilustrasikan penggunaan dan ruang lingkup (di mana variabel yang terlihat atau dikenal) dari variabel lokal.



4. Program pembantu dalam Pascal, ditulis dalam dua bentuk yaitu procedure dan function. Penekanan perbedaan keduanya adalah suatu function mempunyai return value (nilai kembalian), sedangkan sebuah procedure tidak. Kedua jenis program pembantu ini mempunyai beberapa parameter, dengan data type yang ditentukan sebelumnya. Walaupun demikian, dalam kenyataannya, perbedaan antara function dan procedure sangatlah terbatas: Kita dapat memanggil suatu function untuk melakukan suatu tugas dan melewati hasil atau kita dapat memanggil suatu procedure yang menghasilkan nilai melalui parameter (reference parameter akan dibahas lebih lanjut pada bab ini).
5. Kata Procedure menunjukan awal dari deklarasi prosedur. Hal ini diikuti dengan identifier yang digunakan untuk procedure. Setelah nama procedure terdapat daftar parameter formal yang dimulai dengan tanda kurung terbuka (yaitu () dan berisi sejumlah deklarasi parameter dipisahkan oleh titik koma (;). Untuk deklarasi parameter nilai yang Anda akan melihat daftar identifier (identifier dipisahkan dengan tanda koma (,) diikuti oleh titik dua (:)) dan kemudian jenis parameter. Akhir daftar parameter formal diidentifikasi oleh tanda kurung tutup).

6. Berikut ini adalah definisi procedure dan dua macam versi fungsi yang sama, menggunakan cara penulisan yang sedikit berbeda:

Contoh 1;

```
procedure Hello;  
begin  
  ShowMessage ('Hello world!');  
end;  
Procedure Double (Nilai: Integer;  
begin  
  duakali := nilai * 2;  
end;
```

contoh lain:

• **Contoh 2:**

```
program pemakaian_procedure;  
uses crt;  
procedure input_data;  
begin  
    write ( " NIM = " ); readln ( nim ) ;  
    write ( " Nama = " ); readln ( nama ) ;  
end;  
var  
    nim : string [ 14 ] ;  
    nama : string [ 20 ] ;  
begin  
    clrscr ;  
    input_data;  
    writeln ( " Selesai ... " ) ;  
readln;  
end
```

7. Pelajari prosedur berikut ini perhatikan keluarnya pada contoh 3 berikut:

Contoh 3

```
Program contohsegitiga;  
Uses wincrt;  
var  
  n : integer;  
  
Procedure SegiTiga (m : integer ) ;  
  var i , j : integer ;  
  Begin  
    for i :=1 to m do  
      begin  
        for j :=1 to i do  
          write ( '*' ) ;
```

```

        writeln ;
    end ;
End;
Begin
write('jumlah bintang ');readln(n);
segitiga(n);
end.

```

Bila dimasukan nilai $n = 5$, maka hasilnya adalah sebagai berikut:

```

*
**
***
****
*****

```

8. Prosedur juga dapat menerima variabel (data) untuk bekerja dengan ketika mereka dipanggil. Cara mendeklarasikan variabel dalam prosedur
 - a. Variabel yang diterima oleh prosedur diletakan dalam kurung tertutup setelah namanya
 - b. Deklarasi variabel diterima diletakan dalam kurung setelah nama prosedur dan mengakhiri dengan titik koma.
9. Cara pemanggilan Prosedur dan variabel atau nilai yang dilewatkan:
 - a. Bila prosedur ini dipanggil, nama prosedur diikuti oleh satu set kurung.
 - b. Variabel yang akan dilewatkan ditulis dalam kurung.
 - c. Variabel ditulis dalam urutan yang sama seperti yang ditentukan dalam prosedur.
10. Parameter Nilai Pada program sebelumnya, ketika variabel yang dilewatkan ke prosedur, prosedur bekerja dengan menyalin variabel asli. Nilai dari variabel asli yang dilewatkan ke prosedur tidak berubah harganya. Ada dua jenis parameter formal, yaitu parameter nilai dan parameter variabel. Parameter nilai digunakan jika ia hanya diharapkan membawa data pendukung dari bagian pemanggil untuk diberikan kepada prosedur/fungsi. Isi dari parameter nilai tidak akan berubah, meskipun di dalam prosedur/fungsi ia dimodifikasi. Sebaliknya, selain dimaksudkan untuk membawa data ke prosedur/fungsi, parameter variabel diharapkan juga untuk membawa data hasil pemrosesan kembali ke bagian pemanggil. Konsekuensinya, isi parameter variabel dapat berubah selama terjadi pemrosesan di dalam prosedur/fungsi. Perubahan ini akan diterima oleh bagian pemanggil.
11. Salinan yang dibuat prosedur dapat diubah nilai, tetapi ini tidak mengubah nilai aslinya. Ketika prosedur bekerja dengan cara mengcopy variabel, mereka dikenal sebagai parameter nilai. Seperti contoh contoh kode berikut,

```

Program Variable_Parameters (output);
Procedure SWAP ( var value1, value2 : integer );
    var temp : integer;
begin
    temp := value1;
    value1 := value2; {value1 is actually number1}
    value2 := temp   {value2 is actually number2}
end;

```



```

var number1, number2 : integer;
BEGIN
    number1 := 10;
    number2 := 33;
    writeln( 'Number1 = ', number1, ' Number2 = ', number2 );
    SWAP( number1, number2 );
    writeln( 'Number1 = ', number1, ' Number2 = ', number2 )
END.

```

12. Parameter Variabel

Prosedur ini juga dapat diterapkan untuk mengubah nilai variabel asli yang diterima. Untuk menggambarkan hal ini, disini dibuat prosedur kecil yang disebut **swap**. Prosedur ini menerima dua nilai integer. Prosedur sebelumnya menerima parameter nilai tidak dapat melakukan hal ini, karena mereka hanya bekerja dengan salinan dari nilai-nilai asli. Untuk memaksa prosedur untuk menggunakan parameter variabel, terlebih dahulu dideklarasikan variabel (dalam tanda kurung setelah nama fungsi) dengan kata kunci var. Hal ini memiliki efek terhadap penggunaan variabel asli, bukan salinannya.

13. Parameter yang digunakan dalam mendefinisikan prosedur atau fungsi disebut parameter formal, dan parameter yang digunakan untuk memanggil prosedur/fungsi disebut parameter aktual. Keduanya bisa sama, bisa juga berbeda.

C. Kegiatan Praktikum

1. Ketik program contoh 2 di atas, perhatikan serta pelajarylal hasilnnya.
simpan sebagai **Prakt05a**
2. Ketikan program berikut dan pelajarylal hasilnnya dan kemudian simpan sebagai **Prakt05b**

```

Program ContohProsedur_Tanpa_Parameter ;
Uses Crt;
Var
    a, b          : integer ;
Procedure Cetak_Bilangan ;
Begin
    writeln ( ' Hasil pertama = ' , a ) ;
    writeln ( ' Hasil kedua = ' , b ) ;
end ;
BEGIN
    write ( 'Masukkan bilangan pertama : ' ) ; readln ( a ) ;
    write ( 'Masukkan bilang an kedua : ' ) ; readln ( b ) ;
    Cetak_Bilangan ;
    a := a + 10;
    b := b + 5;
    Cetak_Bilangan ;
END.

```

Perhatikan hasilnya, kenapa demikian ????, coba ganti nilai variabel input a dan b, perhatikan hasilnya

3. Ketikkan program berikut dan pelajarilah hasilnya dan kemudian simpan sebagai **Prakt05c**

```

Program Contoh_Procedur_dengan_Parameter;
Uses Crt;
Var
  a,b,maks : integer;
Procedure Besar( x : integer; y : integer; Var Max : Integer); {x dan y parameter nilai }
begin
  if x > y then Max := x
  else Max := y ;
end ;
Procedure Tukar (var x:integer; var y :integer); {x dan y parameter variabel }
Var temp : integer ;
Begin
  temp := x ;
  x := y ;
  y := temp ;
End ;
BEGIN
  write ('Masukkan bilangan a : ');readln (a); {Masukan nilai a dan b }
  write ('Masukkan bilangan b : ');readln (b);
  writeln ('Nilai a dan b sebelumnya : ',a,' ',b);
  Besar(a,b,maks);
  writeln('Maksimum : ',maks) ;           {panggil fungsi Max}
  Tukar(a,b) ;                           {panggil prosedur }
  writeln ( ' Nilai a dan b setelah panggil Tukar : ',a,' ',b);
END.

```

Perhatikan hasilnya, kenapa demikian ????. Apa beda parameter nilai dengan parameter variabel

4. Buatlah procedure untuk menampilkan pola kotak dari karakter '*' seperti Contoh 3., tetapi bagian tengah berisi spasi. Contoh untuk $m = 4$ dan $n = 5$, hasilnya berbentuk :

```

*****
*   *
*   *
*****

```

Simpanlah program dengan **nama Prakt05d**

5. Buatlah program sederhana untuk menghitung gerak pada bidang, jika diketahui kecepatan awal dan sudut elevasi sebagai data masukan. Prosedur yang dibuat adalah :
- tampilan_judul
 - masukkan_data
 - kecepatan_vertikal
 - kecepatan_horizontal

- e. posisixy
- f. tampilkan_hasil

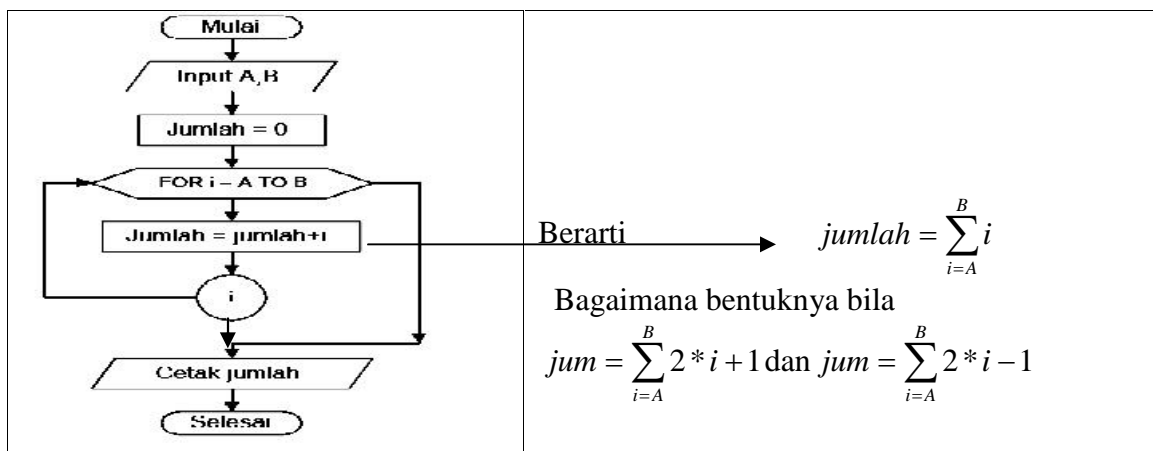
Simpan hasil program sdr. dengan nama Prakt05e

6. Bila jumlah deret bilangan ganjil dapat ditulis dengan $jum = \sum_{i=1}^N 2*i - 1$;

Jumlah deret bilangan genap dapat ditulis dengan $jum = \sum_{i=1}^N 2*i$

Jumlah deret bilangan cacah dapat ditulis dengan $jum = \sum_{i=A}^B i$ yang diagram alir

penjumlahannya adalah sebagai berikut:



Berdasarkan pengertian di atas buatlah program menggunakan prosedur genap dan prosedur ganjil untuk menghitung jumlah bilangan genap dan bilangan ganjil. Simpanlah program dengan nama **Prakt05f**.

7. Buatlah program untuk menghitung potensial pada titik (r) solenoida tak terhingga dengan jumlah lilitan N persatuan panjang. Bila penampang solenoida berbentuk lingkaran berjari-jari R dan arus mengalir arus sebesar I dengan ketentuan program dibagi menjadi prosedur:

- a. tampilan_judul;
- b. masukkan_data (jari-jari, kuat arus, konstan permeabilitas ruang hampa μ_0);
- c. menghitung_potensial; persamaan yang berlaku

$$V = \frac{\mu_0 NI}{2} r, \quad (r < R \text{ atau titik berada di dalam solenoida})$$

$$V = 0, \quad (r = R \text{ atau titik berada di solenoida})$$

$$V = \frac{\mu_0 NI}{2} \frac{R^2}{r}, \quad (r > R \text{ atau titik berada di luar solenoida}) \quad (\text{Penjelasannya baca}$$

Waloejo Loeksmanto, Medan Elektromagnetik, halaman : 139 -140)

- d. tampilkan_hasil
- e. Harga potensial dari dihitung dari r = 0 sampai dengan r tak terhingga, untuk setiap jarak 0.2 cm

- f. Hasil ditampilkan dalam bentuk Tabel
(Simpan hasil program sdr. dengan nama **Prakt05g**)

D. Tugas Pendahuluan

Buatlah diagram alir dan program untuk soal dibawah ini dengan tampilan sebgus mungkin (gunakan perintah Window, Textcolor dll). Jumlah suku sesuai dengan input dari keyboard.

1. Apa yang dimaksud dengan program pembantu
2. Jelaskan perbedaan variabel lokal dengan variabel global
3. Apa guna parameter dalam subprogram ?
4. Jelaskan perbedaan parameter formal dengan aktual.
5. Jelaskan perbedaan parameter nilai parameter variabel
6. Buat desain program dalam algoritma atau diagram alir dan program kegiatan program 5
7. Buat desain program dalam algoritma atau diagram alir dan program kegiatan program 6.
8. Buat desain program dalam algoritma atau diagram alir dan program kegiatan program 7

MODUL PRAKTIKUM VI

A. Kompetensi yang diharapkan

Setelah kegiatan berlangsung diharapkan praktikan mampu:

1. Membuat program sederhana yang user friendly dan mudah ditelusuri
2. Memahami Konsep Pemrograman menggunakan pseudocode pada bahasa Turbo Pascal
3. Memahami pembuatan program secara procedural dengan menggunakan program pembantu dalam bentuk Function
4. Memahami Function memanggil Function lain
5. Memahami pemrograman fungsi dalam bentuk rekursi

B. Kegiatan Sebelum Praktikum

1. Pelajari Bahan ajar yang relevan
2. Pelajarilah teori dan contoh-contoh berikut ini

Prosedur dan fungsi merupakan sub program yang sangat bermanfaat dalam pemrograman, terutama untuk program atau proyek yang besar. Manfaat penggunaan sub program antara lain adalah :

- a. Prosedur dan fungsi merupakan sub program yang sangat bermanfaat dalam pemrograman, terutama untuk program atau proyek yang besar. Manfaat penggunaan sub program antara lain adalah :

- 1) meningkatkan *readability*, yaitu mempermudah pembacaan program
- 2) meningkatkan *modularity*, yaitu memecah sesuatu yang besar menjadi modul-modul atau bagian-bagian yang lebih kecil sesuai dengan fungsinya, sehingga mempermudah pengecekan, testing dan lokalisasi kesalahan.
- 3) meningkatkan *reusability*, yaitu suatu sub program dapat dipakai berulang kali dengan hanya memanggil sub program tersebut tanpa menuliskan perintah-perintah yang semestinya diulang-ulang.

Fungsi (**Function**) hampir sama dengan prosedur, dengan sedikit perbedaan bahwa nama fungsi sekaligus berfungsi sebagai suatu deklarasi (pada blok pemanggil fungsi tersebut). Sehingga setiap fungsi harus diumumkan tipe datanya. Function memiliki nilai kembalian(**Return Value**) sehingga dapat dipanggil dan digunakan dalam suatu proses perhitungan. Sintaks untuk mendeklarasikan fungsi adalah sebagai berikut:

```
FUNCTION nama_fungsi(daftar_parameter ): tipe;  
Bagian deklarasi / pengumuman;  
Bagian pernyataan;
```

Untuk lebih memahami tentang fungsi dengan parameter, pelajarilah program sederhana dibawah yang menerapkan penggunaan fungsi dalam program. Fungsi Tambah hanya memakai parameter nilai. Program pembantu fungsi memiliki nilai kembalian. Sehingga pemanggilan nama fungsi tersebut dapat berperan seperti deklarasi.

```

Contoh program 1;
Program fungsi; { Contoh Program pembuatan Fungsi }
uses crt;

Function Tambah (A,B : Real):Real; {Fungsi penjumlahan }
var
    Hasil: Real;          { variabel lokal }
Begin
    hasil := A + B;
    Tambah := hasil;
End;
Var
    s,t : integer;        { variable global }
Begin{ Program Utama }
ClrScr;
Write ('Masukkan bilangan I: '); Readln(s);
Write ('Masukkan bilangan II: '); Readln(t);
Writeln(Tambah(s,t));    { pemanggilan fungsi }
Writeln('Hasilnya = ', hasil);
Readln;
end.

```

3. Dalam Pascal, ada satu kelebihan dalam cara pemanggilan subprogram. Pascal mengizinkan pemanggilan suatu subprogram dari dalam subprogram itu sendiri. Untuk lebih jelasnya perhatikan potongan program di bawah ini :

```

procedure Z;
begin
    { statement }
    Z;
end;

```

Pada baris terakhir prosedur Z di atas, terdapat pemanggilan kembali terhadap prosedur Z, sehingga prosedur di atas tidak akan pernah selesai dijalankan sebab begitu sampai pada baris terakhir dari prosedur, program akan kembali lagi ke awal prosedur dan terjadi semacam perulangan tanpa perintah perulangan Pascal, dan perulangan dengan cara ini disebut dengan rekursi.

4. Rekursi

- a. *Sub Program Rekursif* adalah sub program yang memanggil dirinya sendiri selama kondisi pemanggilan dipenuhi, adalah dengan melihat sifat sub program rekursif di atas maka sub program rekursif harus memiliki :

- 1) kondisi yang menyebabkan pemanggilan dirinya berhenti (disebut **kondisi khusus** atau **special condition**)
- 2) pemanggilan diri sub program (yaitu bila kondisi khusus tidak dipenuhi).

Secara umum bentuk dari sub program rekursif memiliki statemen kondisional :

if kondisi khusus tak dipenuhi
then panggil diri-sendiri dengan parameter yang sesuai
else lakukan instruksi yang akan dieksekusi bila kondisi khusus dipenuhi

Sub program rekursif umumnya dipakai untuk permasalahan yang memiliki langkah penyelesaian yang terpola atau langkah-langkah yang teratur. Bila kita memiliki suatu permasalahan dan kita mengetahui algoritma penyelesaiannya, kadang-kadang sub program rekursif menjadi pilihan kita bila memang memungkinkan untuk dipergunakan. Secara algoritmis (dari segi algoritma, yaitu bila kita mempertimbangkan penggunaan memori, waktu eksekusi sub program) sub program rekursif sering bersifat tidak efisien. Dengan demikian sub program rekursif umumnya memiliki efisiensi dalam penulisan perintah, tetapi kadang tidak efisien secara algoritmis. Meskipun demikian banyak pula permasalahan-permasalahan yang lebih sesuai diselesaikan dengan cara rekursif (misalnya dalam pencarian/*searching*, yang akan dibahas pada pertemuan-pertemuan yang akan datang).

1) Contoh permasalahan perkalian bilangan

Perkalian dua bilangan bulat positif a dengan b (ditulis ab atau $a \times b$) pada hakekatnya merupakan penjumlahan dari a sebanyak b suku, yaitu $a + a + a + \dots + a$ sebanyak b suku. Misalnya 2×3 dapat diartikan sebagai $2 + 2 + 2 = 6$, yaitu penjumlahan 2 sebanyak 3 suku. Dengan mengingat bahwa suatu bilangan bila dikalikan dengan angka 1 (satu) akan menghasilkan bilangan itu sendiri, maka permasalahan perkalian dengan menyatakannya dalam bentuk penjumlahan di atas dapat diselesaikan dengan komputer secara mudah.

1) Menggunakan fungsi biasa adalah sebagai berikut:

Contoh Pemakaian dua cara pemakaia Fungsi

```
Function KALI_BIASA_F(a,b:integer):longint;
Var hasil : longint; i: integer;
begin
  hasil := 0;
  for i:= 1 to b do hasil := hasil + a;
  KALI_BIASA_F := hasil;
end;
```

2) Menggunakan rekursif Fungsi adalah sebagai berikut:

```
Function KALI_REK_F(a,b:integer):longint;
Begin
  if b>1 then
    KALI_REK_F := KALI_REK_F(a,b-1)+a
  else
    KALI_REK_F := a;
end;
```

5. Selanjutnya perhatikan algoritma untuk menghitung factorial berikut.

```
5! = 4! * 5
4! = 3! * 4
2! = 1! * 2
1! = 1 atau
```

Faktorial N = $N! = N (N-1) (N-2) \dots 3.2.1$
 $N! = N (N-1)!$,untuk $N > 0$
 $N! = N (N-1) (N-2)!$

Jika ditulis dalam program menjadi :

Faktorial(0) = 1 ;
Faktorial(N) = N (N-1)!

Ini merupakan sebuah hubungan rekurens yang berarti nilai suatu fungsi dengan argument tertentu dapat dihitung dengan menggunakan fungsi yang sama hanya saja dengan argument yang lebih kecil. Jadi kita secara programming dapat dilakukan dengan :

Contoh Program 2;

```

Program MenghitungFaktorial;
Uses Crt;
Var
  x : byte;

Function factorial (n:byte):integer;
Begin
  if n<2 then
    factorial:=1
  else
    factorial:=n*factorial(n-1);
End;
BEGIN
  writeln('Masukkan Angka : '); readln(x);
  writeln(x,'! adalah ',factorial(x));
  readln;
END.

```

C. Kegiatan Praktikum

1. Berikut ini adalah program untuk menghitung luas segitiga dengan menggunakan program pembantu Function. Edit program berikut dan pelajari hasilnya serta prosesnya. Untuk meningkatkan pemahaman sdr. ganti-gantilah input data (harga a dan t) saat program telah dirunning. **(Simpan hasil program sdr. dengan nama Prakt06a)**

```

Program Fungsi_pascal;
Uses Crt;

Function luassegitiga(alas: integer; tinggi:integer) : real;
  var luas:real;
Begin
  luas:= alas*tinggi/2;
  luassegitiga:= luas;
End;
Var

```



```

a,t   : integer;
luas  : real;

BEGIN
  write('masukkan nilai alas : '); readln(a);
  write('masukkan nilai tinggi : '); readln(t);
  write('luas segitiga adalah : ', luassegitiga(a,t):0:2);
END.

```

2. Editlah program MenghitungFaktorial (contoh program 2), coba cheek hasilnya, apakah sudah benar. Bila sudah **Simpan hasil program sdr. dengan nama Prakt06b.**
3. Modifikasilah program faktorial di atas untuk menghitung $CNR = \frac{N! * R!}{(N - R)!}$, menggunakan parameter variabel. **Simpan hasil program sdr. dengan nama Prakt06c**
4. Pascal tidak menyediakan Fungsi untuk perpangkatan tinggi, yang ada hanyalah SQR, yaitu untuk pemangkatan kuadrat. Bila akan melakukan perpangkatan lebih dari pangkat 2, maka harus dibuat program tersendiri. Untuk buatlah program untuk menghitung bilangan dengan pangkatnya > 2 (contoh 5 pangkat 4, 5^4 dan sebagainya). Salah satu alternatif dapat Sdr. lakukan adalah memodifikasi program *Function KALI_REK*. **(Simpan hasil program sdr. dengan nama Prakt06d)**
5. Buatlah algoritma untuk menghitung rumus-rumus persamaan kuadrat ($ax^2 + bx + c = 0$) berikut ini:
 - Rumus diskriminan ($D = \sqrt{b^2 - 4 * a * c}$)
 - Jika $D > 0$
 Rumus mencari akar-akar persamaan

$$(X_1 = \frac{-b + \sqrt{b^2 - 4 * a * c}}{2 * a}; X_2 = \frac{-b - \sqrt{b^2 - 4 * a * c}}{2 * a}).$$
 - Jika $D < 0$, akarnya imajiner.

Setiap rumus dijadikan sebuah fungsi tersendiri dengan parameter-parameter yang dimasukkan dari luar fungsi. **(Simpan hasil program sdr. dengan nama Prakt06e)**
6. Saudara ditugaskan menghitung massa jenis benda berbentuk Bola dan Kubus yang terbuat Besi dan Aluminium dengan ukuran dan kerapatan yang berbeda-beda. Agar perhitungan dapat dilakukan dengan cepat dan akurat, instruktur memerintahkan Saudara agar membuat program dengan ketentuan sebagai berikut:
 - a. untuk menghitung Volume, gunakan program pembantu (subprogram) PROCEDURE
 - b. untuk menghitung Massa jenis, gunakan program pembantu (subprogram) FUNCTION
 - c. Input adalah sisi kubus atau jari-jari bola, massa benda

d. Tampilan menu:

- Menu Jenis Benda:
 - 1. Data Kubus
 - 2. Data BolaSelasai
Pilihan Anda !!!
- Menu Bahan Benda
 - a. Alumunium
 - b. Besi
 - c. TembagaPilihan Bahan Anda !!!

e. Input data selesai setelah pada bagian input **jenis benda** dituliskan kata “Selesai” dan Simpan hasil program sdr. dengan nama **Prakt06f**

D. Tugas Pendahuluan

Buatlah diagram alir dan program untuk soal dibawah ini dengan tampilan sebgus mungkin (gunakan perintah Window, Textcolor dll). Jumlah suku sesuai dengan input dari keyboard.

1. Apa guna parameter dalam subprogram ?
2. Jelaskan perbedaan syntax dari procedures dan functions.
3. Jelaskan perbedaan keduanya
4. Kapan kita menggunakan PROCEDURE dan kapan pula menggunakan FUNCTION?
5. Buat algoritma atau diagram alir serta program kegiatan program 3
6. Buat algoritma atau diagram alir serta program kegiatan program 4
7. Buat algoritma atau diagram alir serta program kegiatan program 5
8. Buat algoritma atau diagram alir serta program kegiatan program 6

MODUL PRAKTIKUM VII

A. Kompetensi yang diharapkan

Setelah menyelesaikan modul praktikum ini, diharapkan praktikan mampu:

1. Menggunakan fasilitas editor Turbo Pascal untuk efisiensi penulisan program (penggunaan IIDE).
2. Membuat program sederhana yang user friendly dan mudah ditelusuri
3. Menggunakan flow chart untuk merencanakan suatu program
4. Membuat program dengan menggunakan Array sebagai indeks dari data numerik.
5. Membuat program mengurutkan data
6. Membuat program operasi dasar matriks

B. Kegiatan Praktikum.

1. Pelajari dengan baik Bahan ajar yang relevan dengan baik
2. Bagi yang belum selesai modul praktikum ke-6, selesaikan terlebih dahulu sebelum mengerjakan modul ini.
3. Salin dengan benar program berikut:

Program Fungsi_dan_Prosedur; **{Menggunakan array}**

Uses crt;

var Rp:Real;

 r : array[1..5] of real;

 pilih :char;

 n : integer;

Function R_Paralel(n:integer):real;

var

 i :integer; hasil :

 real; begin

 hasil :=0;

 For i:= 1 to n do

 hasil := hasil + 1/r[i];

 R_Paralel:=1/hasil;

end;

Procedure R_Seri(n:integer);

var

 hasil :real;

 i :integer;

 begin

 hasil:=0;

 for i:=1 to n do

 hasil :=hasil +r[i];

end;

```

Begin
  clrscr; n:=4;
  R[1] :=100; R[2] :=100;
  R[3] :=100; R[4] :=100;
  Writeln('Pilih S : Rangkaian Seri');
  writeln('Pilih P : Rangkaian Paralel');
  write ('pilihan anda :');readln(pilih);
Case pilih of
'S','s': Begin
      R_Seri(n:integer);
      Rp := Hasil;
      End;

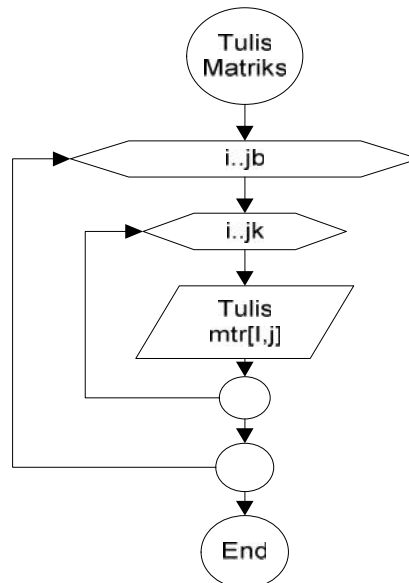
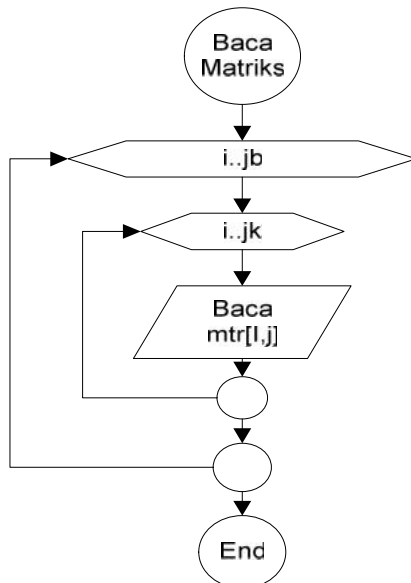
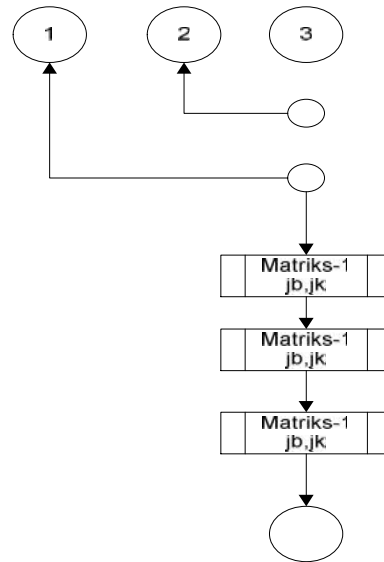
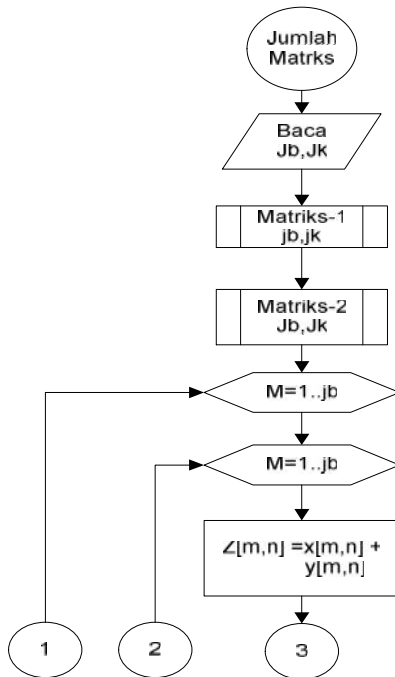
'P','p': Begin
      Rp :=R Paralel(n);
      end;
end;
writeln(Rp:10:3);
readln;
END.

```

4. RUN program di atas dan perhatikan dengan cermat hasil tampilannya. Simpan program ini dengan nama **Prakt07a**
5. Sekarang coba ganti tipe i dari integer ke real dengan cara ganti (i : integer dengan i : real). RUN program tersebut dan perhatikan pesan kesalahan yang muncul. Benarkan kesalahan dengan mengganti tipe i ke tipe semula.
Ganti cara mendeklarasikan data array melalui deklarasi type. Simpan program ini dengan nama **Prakt07b**
6. RUN program di atas dan perhatikan dengan cermat hasil tampilannya. Simpan program ini dengan nama **Prakt07c**
7. Buatlah program yang dapat digunakan untuk mengurutkan dari harga terkecil sampai dengan nilai terbesar. Untuk memudahkan Saudara perhatikan kembali Algoritma yang terdapat pada Bahan ajar sebelumnya, program dibuat dalam bentuk program pembantu Procedure yang terdiri dari:
 - a. Procedure Masukkan_data
 - b. Procedure Mengurutkan_data
 - c. Procedure Penampilkkan_hasil
 Simpan program ini dengan nama **Prakt07d**.
8. Buatlah program yang dapat digunakan untuk menghitung jumlah dua matrik dalam bentuk program pembantu yang terdiri dari:
 - a. Procedure Baca_matriks;
 - b. Procedure Tulis_Matriks;
 - c. Funtion Jumlah_Matriks
 Simpan program ini dengan nama **Prakt07e**

C. Tugas Awal

1. Apakah yang dimaksud dengan array.
2. Dapatkah indeks array diberi tipe data bilangan pecahan ?, jelaskan alasannya
3. Dapatkan array dibuat lebih dari dua dimensi ?, jelaskan dan beri contoh.
4. Buatlah algoritma, diagram alir dan program untuk Prakt07d atau no.7
5. Dengan diagram alir terlampir, sempurnakan jawaban tugas praktikum no.8



MODUL PRAKTIKUM VIII

A. Kompetensi yang diharapkan

Setelah kegiatan berlangsung diharapkan praktikan mampu:

1. Membuat program sederhana yang user friendly dan mudah ditelusuri
2. Memahami pembuatan program secara procedural dengan menggunakan program pembantu untuk operasi karakter dan string
3. Membuat program dengan menggunakan UpCase untuk karakter
4. Memahami pembuatan program dengan perintah operasi karakter dan string
5. Menggunakan string sebagai array dari Karakter (huruf)

B. Kegiatan Sebelum Praktikum

Pada contoh maupun tugas modul-modul sebelumnya kita telah banyak menggunakan string, kalau begitu apakah sebenarnya yang dimaksud dengan string pada Pascal ? String adalah array (susunan) dari character (huruf). Jadi pada hakekatnya kita dapat mengambil masing-masing character dari suatu string dengan memperlakukannya sebagai elemen dari array.

Contoh 1

```
Program Operasi_Contoh;  
Uses Wincrt;  
Var  
Nama : String;  
Begin  
    Nama := 'Suci Rahmawati Roselina';  
    Writeln(Nama[1]);  
    Writeln(Nama[8]);  
    Nama[6] := 'K';  
    Nama[16] := ' ';  
    Writeln(Nama);  
End.
```

Jadi contoh 1 yaitu Program Operasi_Contoh di atas akan menghasilkan Nama[1] adalah huruf 'S', dan Nama[8] adalah huruf 'h', Sedangkan Nama[6] := 'K'; dan nama[8] serta nama[16] := ' ' akan menyebabkan string **nama** ('Suci Rahmawati Roselina') berubah menjadi "Suci Kahmawati oselina".

Kita juga dapat mendeklarasikan string dengan cara seperti berikut :

```
Var  
Nama : String[40];  
Alamat : String;
```

Cara pendeklarasian di atas berarti bahwa variabel **Nama** dapat menampung maksimal 40 character, sedangkan variabel **Alamat** dapat menampung maksimal 255 character. Apabila panjang string pada deklarasi tidak ditentukan, secara otomatis pascal

menyediakan 255 character untuk string. Disamping hal di atas, kita juga dapat memesan variabel array string dengan membuat pada deklarasi variable sebagai berikut:

```
NamaSiwa : Array[1..10] Of String[30];
```

Untuk memudahkan anda memahaminya perhatikan contoh 2 berikut:

```
Program Operasi_Contoh2;
Uses Wincrt;
Var
  NamaSiswa : Array[1..10] Of String[30];
  i      : integer;
Begin
  for i := 1 to 5 do
  Begin
    Write('nama siswa ke ',i); Readln(namasiswa[i]);
  end;
    Writeln(Namasiswa[1]);
    Writeln(Namasiswa[7]);
    Writeln(Namasiswa[5]);
  End.
```

Apabila program ini kita running dengan memasukan nama Badu, Mince, Roselina, Fajar, Suci, maka akan menghasilkan Badu dan Suci. Kenapa demikian ?, Pelajarilah contoh 2 dengan baik. Tentu anda mendapatkan jawabannya.

Pada Pascal turbo juga terdapat beberapa fungsi bawaan pascal yang dapat digunakan untuk oprasi string. Adapun fungsi pembantu tersebut antara lain adalah :

a. Length

Length berfungsi untuk mendapatkan panjang string yang telah didefenisikan, dengan sintaks pemakaiannya adalah sebagai berikut:

Syntax : length(s)

Contoh : : n:=length(s);

Misalnya s:='Apa kabar ?'; n akan berisi 11.

Copy, mendapatkan bagian dari suatu string.

Syntax : copy(s,from,howmuch)

Contoh : : st:=copy(s,5,5);

Menggambil 5 character mulai posisi ke 5

Misalnya s:='Apa kabar ?'; st akan berisi 'kabar'.

b. Pos

Pos berfungsi untuk mengambil posisi substring dari suatu string yang telah didefenisikan, dengan sintaks pemakaiannya adalah sebagai berikut:

Syntax : Pos(substr,s)

Contoh : : n:=pos('kabar','Apa kabar ?'); { n:=5; }

Jika substring tidak ditemukan, maka akan mengembalikan 0.

c. Val

Val berfungsi untuk mengkonversi tipe data string menjadi tipe data numeric, dengan sintaks pemakaiannya adalah sebagai berikut:

Syntax : val(strvar,numvar,errorcode)

strvar adalah suatu variabel string yang akan di konversi

numvar adalah variabel integer atau real

errorcode adalah variabel integer yang akan berisi nilai kesalahan, jika berisi 0 sukses.

Contoh : :

```
Program String_Contoh;
Uses WinCRT;
Var
  s : string;
  e : integer;
  r : real;
Begin
  Write('Masukan sebuah bilangan : '); readln(s);
  val(s,r,e);
  if e<>0 then
    writeln('Kesalahan pada posisi : ',e)
  else
    writeln('Hasil konversinya adalah : ',r:4:3);
End.
```

d. Str

Fungsi Str berguna untuk mengkonversi numerik menjadi string, dengan sintaks pemakaiannya adalah sebagai berikut:

Syntax : (numvar,strvar)

Contoh : :

```
Vvar
s : string;
i : integer;
BEGIN
  write('Masukan Bilangan bulat : '); readln(i);
  str(i,s);
  writeln('Bilangannya adalah : ',s);
END.
```

Jika anda bekerja dengan type real, anda perlu melakukan pemformatan sebelum konversi ke string.

Contoh : :

```
Var
  s : string;
  r : real;
Begin
  write('Masukan Bilangan real : '); readln(r);
  str(r:4:3,s);
```



```
writeln('Bilangannya adalah : ',s);  
End.
```

e. Concat,

Concat berguna untuk mengabung dua string atau lebih

Syntax : concat(s1,s2,...,sn)

Contoh : st:=concat(s1,s2);

Jika s1 adalah 'Penta', dan s2 adalah 'COM', maka st menjadi 'PentaCOM'

Sebenarnya kita juga dapat menggunakan operator + untuk mengabung string.

Contoh : st:=s1 + s2; atau sama dengan st:=concat(s1,s2);

f. Insert, menyisip suatu string kedalam string lain dari posisi tertentu

Syntax : insert(source,target,index)

Contoh :

```
Var  
  s1, s2 : string;  
Begin  
  s1:='not '  
  s2:='I do love you';  
  insert(s1,s2,6);  
  writeln(s2); { I do not love you }  
End.
```

g. Delete

Delete berguna untuk menghapus sejumlah character dari string mulai posisi ke i

Syntax : delete(s,i,n);

```
Var  
  s : string;  
Begin  
  s:='I am not responsible for that !';  
  delete(s,6,3);  
  writeln(s); { I am responsible for that }  
End.
```

Sebagai program tambahan, perhatikan program berikut ini

```
Program String_tambahan;  
Uses WinCrt;  
var nama : string;  
    urutan, posisi: integer;  
Begin  
  nama:='Dijas Keren';  
  writeln('Yang ini delete');  
  for urutan:=1 to 11 do  
    Begin  
      posisi:= 13 - urutan;  
      delete(nama, posisi, 1);
```

```

        writeln(nama);
    End;
    writeln;
    nama:='DijasKeren';
    writeln('Yang ini insert');
    writeln('Sebelum di Insert : ', nama);
    insert('-', nama, 6);
    writeln('Setelah di Insert : ', nama);
    readln;
End.

```

Jadi isi penggunaan delete itu maksudnya menghapus (string, posisinya, jumlah yang di hapus). Sedangkan insert berguna untuk menyisip (apa yang mau disisipkan, string, posisinya).

h. Upcase.

Upcase berfungsi untuk merubah huruf kecil menjadi huruf capital untuk mudahnya perhatikan contoh berikut ini:

```

PROGRAM convert_upper;
    USES WinCRT;
    VAR
        text : string;
        i : integer;

    BEGIN
        clrscr;
        writeln ('Input some text in all lowercase letters');
        readln (text);
        for i := 1 to length(text) do
            begin
                text[i] := upcase(text[i]);
            end;
        writeln ('The uppercase text is: ');
        writeln (' ', text);
        readln;
    END.

```

C. Tugas Pendahuluan

1. Apakah dimaksud dengan string ?
2. Jelaskan hubungan antara string dan character pada pascal !
3. Jelaskan dengan contoh sintaks dan penggunaan kata simpan **upcase** pada pascal
4. Bagaimana cara mendeklarasikan string ?
5. Jelaskan cara mengetahui panjang string !
6. Buat diagram alir dan program untuk praktikum 3,4 dan 5

D. Kegiatan PraktikumLatihan di laboratorium

1. Salin dengan benar program berikut:

```
program PermainanHuruf;
uses crt;
type az kalimat = array[az] of byte;
var
nama : string;
i : byte;
begin
  clrscr;
  write('tuliskan nama anda dengan baik ! '); readln(nama);
  if(length(nama) >= 2) AND (length(nama) < 4) then
    writeln('Anda Orangnya Royal');
  if((length(nama) >= 4) AND (length(nama) < 8) then
    writeln('Anda Orangnya Praktis');
  if((length(nama) >= 8)AND (length(nama) < 20)) then writeln('Anda
Orangnya Pelit');
  repeat until
    keypressed;
END.
```

RUN program di atas dan perhatikan dengan cermati hasil tampilannya. Simpan program ini dengan nama Prakt08a.pas.

2. Buat program untuk latihan berikut ini menghitung jumlah huruf dalam sebuah kalimat, salin dengan banar pelajari keluarannya:

```
Program Permaianan_huruf;
uses crt;
type
  az = 'A'..'Z';
  kalimat = array[az] of byte;
var
  huruf : char;
  kata : string;
  besar : az;
  pernyataan : kalimat;
  i : byte;
do
  pernyataan[besar] := 0; write('tuliskan kalimat '); Readln(kata);
  for i := 1 to length(kata) do
  begin
    huruf := kata[i];
    if(huruf >='A') and (huruf <='Z') then
      inc(pernyataa
```

```

do
if pernyataan [besar] <> 0 then
    writeln('jumlah huruf,':S,besar,':':2,pernyataan[besar]);
repeat until keypressed;
END.

```

Perhatikan keluaran program permainan huruf di atas, kemudian modifikasi agar dapat digunakan untuk menghitung huruf Kapital dan huruf Kecil, dalam sebuah kalimat. Simpan program ini dengan nama Prakt08b.pas.

Buat program animasi teks, dengan menampilkan kalimat pada tengah-tengah layar dan yang sebelumnya bergerak dari kiri ke kanan dan kanan ke kiri. Simpan program ini dengan nama Prakt08c.pas.

3. Buat program animasi teks, dengan menampilkan kalimat pada tengah-tengah layar dengan huruf muncul satu per satu. Simpan program ini dengan nama Prakt08d.pas.
4. Palindrom adalah sebuah kalimat yang jika dibaca dari belakang sama dengan dibaca dari depan. Contoh “murah ada harum”, atau “kasur ini rusak”, Rencanglah program untuk mengetahui apakah sebuah kalimat merupakan palindrome atau tidak. Simpan program ini dengan nama Prakt08e.pas.
5. Buatlah program untuk menentukan apakah kata yang kita inginkan terdapat dalam sebuah kalimat yang ditulis. Simpan program ini dengan nama Prakt08f.pas.

MODUL PRAKTIKUM IX

DASAR-DASAR PEMOGRAMAN

A. Tujuan:

Setelah menyelesaikan modul praktikum ini, diharapkan praktikan mampu:

1. Menggunakan fasilitas editor Turbo Pascal untuk efisiensi penulisan program (penggunaan IDE).
2. Membuat program sederhana yang user friendly dan mudah ditelusuri
3. Menggunakan program grafik yang sederhana.
4. Menggunakan perintah penginisialisasian penggunaan grafik dan mengakhiri penggunaan grafik.
5. Mengenal koordinat layar.
6. Membuat gambar-gambar sederhana pada mode grafik seperti lingkaran, kotak, titik, garis, ellips, busur, teks dan sebagainya.
7. Membandingkan beberapa perintah mode teks dengan mode grafik.

B. Pelajari bahan perkuliahan Grafik I

C. Kegiatan Praktikum.

1. Buat rancangan sendiri program yang akan sdr. kerjakan di laboratorium.
2. Buat dengan ketentuan sebagai berikut:
Nama file program : Prakt9a.pas
Judul program : Inisialisasi Grafik
Untuk membuka grafik dibuat dalam procedure Buku_Grafik

Sebaiknya procedure ini bersifat fleksibel sehingga bila terjadi kesalahan, maka program akan berhenti sambil memberikan pesan kesalahan yang terjadi. Tekan_Enter merupakan Procedure yang digunakan untuk menahan tampilan layar hingga ditekan tombol Enter.

Tutup_grafik dibuat dalam procedure khusus pula, procedure ini berfungsi untuk menutup modus grafik dan kembali ke modus teks.

Program hanya berisi perintah untuk membuka grafik, tahan tampilan sampai ditekan tombol enter dan kemudian dilanjutkan dengan menutup grafik.

Usahakan pemakaian memori sedikit mungkin.

4. Apa yang terjadi bila di dalam program disisipkan perintah CLRSCR, bagaimana seharusnya.
5. Buat program untuk melukiskan beberapa segiempat dengan berbagai ukuran dan letaknya pada layar monitor.
6. Buat program untuk melukiskan beberapa buah garis berbagai ukuran (panjang), arah, dan letak pada layar monitor.
7. Buat program untuk melukiskan beberapa lingkaran dengan berbagai ukuran dan letak pada layar monitor.

8. Buat program untuk melukiskan beberapa ellips dengan berbagai ukuran dan letak pada layar monitor.
9. Buat program untuk melukiskan beberapa busur lingkaran dengan posisi yang berbeda dan letak pada layar monitor.
10. Simpan semua program ini dalam sebuah file, dengan Prakt09b.pas
11. Buat program dengan judul Teks Grafik dengan nama file Prakt9c, yang berisikan garis horizontal dan garis vertikal yang berpotongan di tengah-tengah layar. Pada tengah-layar tersebut muncul tulisan "Belajar Grafik Nii Yee". Buat tulisan ini dalam 3 jenis ukuran.
12. Dengan menggunakan perintah menggambar grafik yang telah ada, coba sdr. buat suatu bentuk (misalnya gambar mobil, pesawat, komputer, wajah, dan sebagainya). Penilaian tugas praktikum ini ditentukan oleh kreativitas sdr. dalam membuat bentuk ini. Simpan program dalam file Prakt09d.pas.

D. Tugas Awal

1. Apakah fungsi perintah-perintah di bawah ini, dan bagaimana format pemakaiannya:

| | | | |
|----------------|---------------|-----------------|-------------------|
| a. Initgraph | b. Closegraph | c. Rectangle | d. Putpixel |
| e. Circle | f. Ellips | g. Line | h. Arc |
| i. Outtextxyj. | j.Outtext | k. settextstyle | l. settextjustify |
2. Apakah fungsi perintah-perintah di bawah ini, dan bagaimana format pemakaiannya:

| | | | |
|----------------|-----------------|------------------|-------------|
| a. Setfillsyle | b. bar | c. setaspecratio | d. pieslice |
| e. setcolor | f. setlinestyle | g. moveto | |
3. Apakah perbedaan antara perintah line, lineto, dan linerel ?
4. Bagaimana cara masuk ke mode grafik dari mode teks ?
5. Bagaimana cara membersihkan layar dengan mode graph, dan bagaimana perbedaan cara menghapus layar pada mode teks ?
6. Pada mode grafik, tipe data numerik tidak dapat ditampilkan secara langsung, bagaimana cara sdr. untuk menampilkan tipe data tersebut pada mode grafik. Jelaskan secara algoritma ?
7. Apakah perbedaan antara koordinat monitor komputer dengan koordinat kartesian ?

MODUL PRAKTIKUM X

DASAR-DASAR PEMOGRAMAN

A. Tujuan:

Setelah menyelesaikan modul praktikum ini, diharapkan praktikan mampu:

1. Menggunakan fasilitas editor Turbo Pascal untuk efisiensi penulisan program (penggunaan IIDE).
2. Membuat program sederhana yang user friendly dan mudah ditelusuri
3. Membuat program grafik lanjut.
4. Membuat program pembantu beda file dalam bentuk Unit
5. Memakai koordinat layar dan transformasi untuk pemograman
6. Membuat animasi sederhana

B. Pelajari bahan perkuliahan X

C. Kegiatan Praktikum.

1. Dalam kegiatan ini, sdr. merancang sendiri program yang akan dikerjakan di laboratorium dengan ketentuan sebagai berikut:
 - a. Sebagai procedure dapat diambil dari program Prakt09a.pas.
2. Buat program dengan memanfaatkan program pembantu di atas untuk menggambar grafik gerak peluru (gerak pada bidang, ingat kembali prakt06d atau prakt02h)

$$y = v_o \sin (\text{teta}) t - 0.5 g t^2$$

$$x = v_o \cos (\text{teta}) t.$$

Simpan program ini dengan nama Prakt10a.pas.

3. Buat program untuk membuat garis yang menyapu layar dari kiri ke kanan. Garis ini memakai warna latar belakang layar. Buat juga cara menyapu layar dari arah yang lain. Simpan program ini dengan nama file : Prakt10b.pas
4. Buat animasi suatu gerak lingkaran dari kiri ke kanan.
Caranya: Buat lingkaran pada posisi tertentu, kemudian hapus lingkaran tersebut dengan lingkaran baru yang warnanya sama dengan warna latar belakang. Ulangi kembali proses pertama pada posisi yang berbeda dan seterusnya.
Simpanlah program ini dengan nama Prakt10c.pas.
5. Buatlah program untuk membuat lingkaran yang bergerak pada lintasan lingkaran, dengan menggunakan persamaan:
$$x = R \cos (\text{teta}) t.$$
$$y = R \sin (\text{teta}) t.$$

Simpan program ini dengan nama Prakt10d.pas.

6. Buatlah program untuk membuat lingkaran yang bergerak pada lintasan yang dibentuk oleh hasil pemaduan gerak yang saling tegak lurus (lissajous), baca Fisika Untuk Universitas I halaman 280 – 283)

$$x = A \sin (\text{teta}) t - \text{phasa-1} .$$

$$y = A \sin (\text{teta}) t + \text{phasa-2}$$

Simpan program ini dengan nama Prakt10e.pas.

D. Tugas Awal

1. Buat rumusan transformasi koordinat alam semesta ke koordinat layar monitor dengan menggunakan transformasi linier. (umpamakan transformasinya adalah : $X_S = A + Bx$, dan $Y_S = C + Dy$, dimana (X_S, Y_S) adalah koordinat layar monitor dan (x_l, y_l) adalah koordinat alam semesta.)

2. Buat program pembantu untuk rumusan yang telah dibuat di atas.

3. Buat juga flow chart untuk program transformasi koordinat.

4. Untuk membantu Sdr dalam merencanakan program di atas dapat diikuti program berikut:

```
program Transformasi;
uses
```

```
var
```

```
Procedure bidang gambar(var x min,y min,x max,y max :real);
```

```
begin
```

```
    writeln(' Masukan Data Bidang Gambar');
```

```
    write(' X Minimum      : ');readln(x min);
```

```
    write(' Y Minimum      : ');readln(y min);
```

```
    write(' X Maksimum     : ');readln(x max);
```

```
    write(' Y Maksimum     : ');readln(y max);
```

```
end;
```

```
Procedure transformasi (x min,y min,x max,y max,x_g,y_g:real;var x l,y l : real);
```

```
begin
```

```
    x_l := Getmaxx /(x max-x min)*(-x min+x.g);
```

```
    y_l := Getmaxy /(y max-y min)*(y max- y.g);
```

```
end;
```

5. Perintah apakah yang digunakan untuk mendapatkan warna latar belakang layar dan menunda waktu jalannya program ?

MODUL PRAKTIKUM XI

DASAR-DASAR PEMOGRAMAN

A. Tujuan:

Setelah menyelesaikan modul praktikum ini, diharapkan praktikan mampu:

1. Menggunakan fasilitas editor Turbo Pascal untuk efisiensi penulisan program (penggunaan IDE).
2. Membuat program sederhana yang user friendly dan mudah ditelusuri 3. Membuat program grafik lanjut.
3. Membuat program animasi sederhana
4. Menggunakan variabel pointer, perintah putimage, getimage.

B. Baca referensi yang terkait

C. Kegiatan Praktikum.

1. Kembangkan program no.6 pada modul (praktl0c.pas) untuk memperoleh animasi suatu lingkaran dimana lingkaran tsb memantul dalam arah sumbu-x dan berhenti bila ditekan tombol tertentu.

Caranya : pada saat posisi pusat lingkaran = posisi layar maksimum - jari jari lingkaran atau pada saat posisi pusat lingkaran = posisi layar minimum + jari-jari, ambil perubahan perpindahan titik pusat lingkaran berharga negatif ($dx = -dx$).

Simpan program ini dengan nama Praktl1a.pas

2. Buat program sederhana untuk menampilkan gerak bola yang dapat terus menerus memantul sampai berhenti bila ditekan tombol tertentu, arahnya tidak hanya tergantung pada x saja, tetapi juga tergantung kepada sumbu-y (gerakan bebas).

Dengan persyaratan sebagai berikut:

- ✓ Bola akan memantul pada batas atas monitor apabila $y \leq y_{\text{minimum}}$ layar monitor.
- ✓ Bola akan memantul pada batas bawah monitor apabila $y \geq y_{\text{maksimum}}$ layar monitor
- ✓ Bola akan memantul pada batas kiri monitor apabila $x \leq x_{\text{minimum}}$ layar monitor.
- ✓ Bola akan memantul pada batas kanan monitor apabila $x \geq x_{\text{maksimum}}$ layar monitor
- ✓ Anggaplah bahwa pemantulan bola lenting sempurna (tidak mengalami peredaman) sehingga:
- ✓ Pemantulan dengan batas monitor atas dan bawah adalah $dy = -dy$;
Pemantulan dengan batas monitor kiri dan kanan adalah $dx = -dx$;

Simpanlah program ini dengan nama program praktl2c.pas.

D. Tugas Awal

1. Buat flow chart untuk program-program di atas.
2. Apa yang dimaksud dengan animasi
3. Apa fungsi dan bagaimana format perintah-perintah: putimage, getimage, imagesize, dan freemem.

MODUL PRAKTIKUM XII

A. Kompetensi yang diharapkan

Setelah kegiatan berlangsung diharapkan praktikan mampu:

1. *Membuat program sederhana yang user friendly dan mudah ditelusuri*
2. *Memahami pembuatan program secara procedural dengan menggunakan record*
3. *Membuat program dengan record dengan with*
4. *Memahami pembuatan program perintah-perintah operasi record*
5. *Memahami pembuatan program untuk menyimpan sebuah file data*

B. Kegiatan Sebelum Praktikum

1. Pelajarilah dengan baik diktat perkuliahan.
2. Record adalah tipe data kompleks yang elemen-elemennya boleh mempunyai tipe data yang berbeda.

Record lebih kompleks daripada array karena record merupakan kumpulan beberapa variabel dengan tipe data yang berbeda. Berbeda dengan array yang tiap elemennya ditandai dengan nomer indeks maka record ditandai dengan nama variabel anggotanya. Cara mengakses elemen dari record dilakukan dengan menyebutkan nama variabel anggota setelah menyebutkan nama record yang akan diakses. Di antara nama record dan nama variabel anggota dipisahkan tanda titik (.).

Cara pendeklarasian record adalah sebagai berikut :

```
var
  B : record
    X : integer;
    Y : real;
  end;
```

Pada pendeklarasian di atas, maksudnya adalah sebagai berikut : variabel B berupa record dengan dua elemen yaitu X bertipe integer dan Y bertipe real.

Untuk mengakses elemen dari variabel B seperti berikut :

```
B.nama_variabel
contoh : B.X := 10;
```

3. Pada contoh program merupakan contoh suatu program dengan record untuk mencatat waktu ulang tertentu. Pelajarilah program berikut ini:

```
Program RECORD_INTRO;
Uses Crt;
type
  badu = record
    bulan      :integer;
    Hari       : string[20];
    tahun      : integer;
  end;
var
```

```

    waktu : badu;
begin
    write('tanggal sekarang');readln(waktu.hari);
    write('bulan sekarang');readln(waktu.bulan);
    write('tahun sekarang');readln(waktu.tahun);
    writeln('hari ini adalah ',waktu.hari,':',waktu.bulan,':', waktu.tahun);
end.

```

4. Data suatu proses dapat disimpan dalam sebuah file data
5. Proses untuk menulis data dalam suatu adalah sebagai berikut:

C. Kegiatan Sebelum Praktikum

1. Berdasarkan contoh program RECORD_INTRO, buatlah record yang berisikan :
 1. Nama
 2. NIM
 3. Alamat
2. Lengkapi program di atas dengan menambahkan alamat dan kelas dari siswa dalam recordnya dan datanya dalam sebuah file data.
3. Berdasarkan program program di atas buatlah record perintah WITH untuk banyak siswa. Caranya pada bagian variable buatlah tipe data/bagian variabel
 1. DTM : Array[1..50] of RecordMhs;
 2. Ganti perintah **With DatMhs do** dengan **With DTM[i] do**, perhatikan catatan bahan kuliah yang telah diberikan
 3. Ingat bagian atasnya harus ada perintah pengulangan, beloh dengan for .. to .. do, atau repeat .. until atau while .. do.
4. Buatlah algoritma dan program record data base sejumlah mahasiswa yang terdiri dari
 - a. NAMA
 - b. NIM
 - c. ALAMAT
 - d. IPK
 (ingat semua variable ini harus dalam bentuk array)
 Dengan ketentuan
 - a. Program terdiri dari prosedur masukkan data mahasiswa
 - b. Data setiap mahasiswa secara keseluruhan dapat dilihat dengan dimasukkan nama atau nim atau alamat atau IPK,
 - c. Simpan data record di atas dalam sebuah file.

Sebagai pertimbangan perhatikan pola pada hand out perkuliahan IX

D. Tugas Praktikum

1. Berlatih lebih dahulu berdasarkan tugas awal no.1, kemudian simpan dengan nama **Prakti12a**.
2. Editlah program record menggunakan perintah WITH berikut ini:

```
Program RecordWith;
uses crt;
Type
RecordMhs = Record
  Nama : String[20];
  NIM : LongInt;
  na : Real;
  nilai : Array[1..3] of Real;
END;

Var
j,i,n : integer;
DatMHs : RecordMhs;

Begin
With DatMhs do
Begin
  write(' nama '); readln(nama);
  write(' NIM '); readln(NIM);
  for j := 1 to 3 do
  begin
    write('nilai ke ',j);readln(nilai[j]);
  end;
  na := (nilai[1]+nilai[2]+nilai[3])/3;
end;
  With DatMhs do
  Begin
    writeln(nama);
    writeln(NIM);
    for j := 1 to 3 do
    begin
      writeln(na);
    end;
  end;
  readln;
END.
```

Simpan dengan nama prakti12b

3. Lengkapi program di atas dengan menambahkan alamat dan kelas dari siswa dalam recordnya dan datanya dalam sebuah file data. Simpan dengan nama **prakti12c**.

4. Buatlah program berdasarkan tugas awal No. 3, simpan hasilnya dengan nama **prakti12d**.
5. Buatlah program berdasarkan tugas awal No. 4, simpan hasilnya dengan nama **prakti12e**

Untuk membantu Sdr. Praktikum ini pelajaryl program-program berikut ini:

```

1. Program TulisDataRecordWith;
   uses Crt;
   Type  RecordMhs = Record
     Nama : String[20];
     NIM  : LongInt;
     na   : Real;
     nilai : Array[1..3] of Real;
   END;
   TFile = File of RecordMhs;
   Var
     i,j,n,k : integer;
     nfile : string[12];
     vfile : Tfile;
     Dtm : array[1..5] of RecordMhs;
   Begin
     write('nama file');readln(nfile);
     write('jumlah siswa ');readln(n);
     write('jumlah komponen nilai');readln(k);
     Assign(vfile,nfile);
     Rewrite(Vfile);
     for i := 1 to n do
       begin
         With Dtm[i] do
           Begin
             write(' nama '); readln(nama);
             write(' NIM '); readln(NIM);
             na := 0;
             for j := 1 to k do
               begin
                 write('nilai ke ',j);readln(nilai[j]);
                 na := na+nilai[j];
               end;
             na := na/k;
           end;
         write(Vfile,dtm[i]);
       end;
     Close(vfile);
     Clrscr;

```

```

for i := 1 to n do
begin
With dtm[i] do
Begin
gotoxy(10,2+i);writeLN(nama);
gotoxy(20,2+i);writeLN(NIM);
gotoxy(30,2+i);writeLN(na:8:2);
gotoxy(10,1);writeLN('nama');
gotoxy(20,1);writeLN('NIM');
gotoxy(30,1);writeLN('nilai');
end;
end;
END.

```

2. Program BacaDataRecordWith

```

Program BacaDataRecordWith;
uses Crt;
Type
RecordMhs = Record
    Nama : String[20];
    NIM  : LongInt;
    na   : Real;
    nilai : Array[1..6] of Real;
END;
TFile = File of RecordMhs;
Var
i,n : integer;
nfile : string[12];
vfile : Tfile;
Dtm  : array[1..100] of RecordMhs;
Begin
write('nama file');readln(nfile);
Assign(vfile,nfile);
Reset(Vfile);
repeat
I := i +1;
read(Vfile,dtm[i]);
until eof(vfile);
n := i;
Close(vfile);

for i:= 1 to n do
begin
With Dtm[i] do
Begin
writeln(' nama ', nama);

```

```

        writeln(' NIM ', NIM);
        writeln(' NIM ', na);
    end;
end;
readln;
END.

```

3. Program untuk menambah data berikut ini :

```

Program RecordWith;
uses Crt;
Type
RecordMhs = Record
    Nama : String[20];
    NIM : LongInt;
    na : Real;
    nilai : Array[1..6] of Real;
END;
TFile = File of RecordMhs;

var
i,j : integer;
nfile : string[12];
vfile : Tfile;
Procedure tulis(var Vfile :TFile;NmrRec :word);
Var
    Posx : Byte;
    DatMHs : Array[1..100] of RecordMhs;
Begin
    With DatMhs[NMrRec] do
        Begin
            ClrScr;
            Gotoxy(10,2);write(' Masukkan data Ke-',NmrRec);
            Gotoxy(10,4);write(' nama ');
            Gotoxy(10,5);write(' NIM ');
            Gotoxy(Posx,4);readln(nama);
            Gotoxy(Posx,5);readln(NIM);
            for j := 1 to 3 do
                begin
                    Gotoxy(10,5+j);write('nilai ke ',j);
                    Gotoxy(Posx+7,5+j);readln(nilai[j]);
                end;
                na := (nilai[1]+nilai[2]+nilai[3])/3;
            end;
            write(Vfile,DatMhs[NmrRec]);
        end;
    end;

```



```

Procedure TambahData;
Var
Vfile : TFile;
Jwb :Char;
NmrRec,Nmr_kom : Word;
Begin
CLrscr;
write('nama file');readln(nfile);
Assign(vfile,nfile);
Reset(Vfile);
Nmr_kom := FILESIZE(VFile);
SEEK(VFile,Nmr_kom);
NmrRec := Nmr_kom +1;
Repeat
tulis(VFile,NmrRec);
GotoXy(10,10);write('nambah data lagi (y/t)');
Repeat
Jwb := Readkey
until Jwb in ['Y','y','T','t'];
until upcase(jwb)="T";
Write(Upcase(jwb));
End;
BEGIN
TambahData;
readln;
END.

```

MODUL PRAKTIKUM XIII-XVI ALGORITMA DAN PEMOGRAMAN

Praktikum kelompok untuk 2 orang

Tugas:

Buatlah sebuah paket program interaktif yang berisikan:

- a. Perintah semua dasar
- b. Menggunakan perintah struktur control pengulangan dan kondisional
- c. Program dibuat dalam program pembantu (procedure dan function)
- d. Mengandung perintah teks dan grafik dan simulasi atau anemasi
- e. Serahkan desian program sebelum dikerjakan
- f. Desain program diserahkan 1 minggu sebelum praktikum kelompok dilaksanakan
- g. Simpan dengan TA + gabungan nama anda