

MODUL PENDIDIKAN PEMAKAI
PERPUSTAKAAN UNIVERSITAS NEGERI PADANG

Paket Dasar : Pengenalan Perpustakaan dan
Sumber-sumber Informasi

NO. TITIK	20 APRIL 2008
NO. HARI	Hd
NO. K1	K1
NO. 86/Hd/2008 - m. (1)	
NO. 025.56	Mel m.1

Oleh

Dra. Nelwati, M.Pd
Pustakawan UNP

Disampaikan dalam kegiatan "*WORKSHOP On ELECTRONIC LIBRARY
USER EDUCATION PROGRAM*", 25-26 Mei 2007
di Perpustakaan UNP

PERPUSTAKAAN UNIVERSITAS NEGERI
PADANG

2007

MODUL PENDIDIKAN PEMAKAI

Paket Dasar

Pengenalan Perpustakaan dan Sumber-sumber Informasi

1. TUJUAN INSTRUKSIONAL

Paket ini merupakan bagian pertama dari Modul User Education untuk mahasiswa Universitas Negeri Padang. Paket Dasar ini bertujuan untuk memberikan pengenalan kepada mahasiswa tentang perpustakaan UNP dan sumber-sumber informasi dan cara pemanfaatannya.

2. TUJUAN INSTRUKSIONAL KHUSUS

- a. Mahasiswa dapat mengetahui Misi Pendidikan Nasional
- b. Mahasiswa dapat mengetahui Visi perpustakaan
- c. Mahasiswa dapat mengetahui Misi perpustakaan UNP
- d. Mahasiswa dapat mengetahui fungsi perpustakaan
- e. Mahasiswa dapat mengetahui tugas perpustakaan
- f. Mahasiswa dapat mengetahui tata letak ruangan perpustakaan
- g. Mahasiswa dapat mengetahui pengorganisasian koleksi
- h. Mahasiswa dapat mengetahui penelusuran dan penemubalihan informasi
- i. Mahasiswa dapat memahami prosedur peminjaman dan pengembalian

Modul Pendidikan pemakai terdiri dari tiga paket yaitu:

- | | |
|-------------------|---|
| Paket Dasar | : Pengenalan Perpustakaan dan Sumber-sumber informasi |
| Paket Lanjutan I | : Strategi pencarian informasi di Internet dan Online Data Base |
| Paker Lanjutan II | : Plagiarisasi, Hak Cipta, HAKI dan Sitasi Bibliografi |

Paket Dasar ini terdiri dari:

- a. Tujuan Umum
- b. Tujuan Khusus
- c. Materi Pelatihan.
- d. Waktu Pembelajaran
- e. Presentase
- f. Latihan
- g. Daftar Istilah.

3. MATERI PELATIHAN

- Misi Pendidikan Nasional
- Misi dan visi Perpustakaan
- Fungsi dan tugas perpustakaan
- Tata letak ruangan perpustakaan
- Pengorganisasian koleksi
- Penelusuran dan pemubalihan informasi
- Layanan sirkulasi dan layanan informasi
- Tata tertib Perpustakaan

4.. HASIL PEMBELAJARAN

Paket Dasar membahas hal-hal yang berhubungan dengan pengenalan perpustakaan dan sumber-sumber informasi. Setelah mengikuti paket ini, mahasiswa dapat :

- a. Mengetahui Misi Pendidikan Nasional
- b. Mengetahui Visi perpustakaan
- c. Mengetahui Misi perpustakaan UNP
- d. Mengetahui fungsi perpustakaan
- e. Mengetahui tugas perpustakaan
- f. Mengetahui tata letak ruangan perpustakaan
- g. Mengetahui pengorganisasian koleksi
- h. Memahami penelusuran dan penemubalihan informasi
- i. Memahami prosedur peminjaman & pengembalian

4. WAKTU PEMBELAJARAN

Paket Dasar ini dilaksanakan pada awal semester I untuk mahasiswa baru. Bentuk kegiatan dalam Paket Dasar ini adalah : ceramah 60 menit dan kunjungan mahasiswa ke bagian-bagian perpustakaan mulai dari lantai I sampai lantai V.

5. KRITERIA PESERTA

Peserta Paket Dasar ini adalah mahasiswa baru dan mahasiswa lama yang belum pernah mengikuti orientasi perpustakaan UNP. Mahasiswa yang tidak mengikuti orientasi perpustakaan tidak dibenarkan untuk menjadi anggota perpustakaan.

MODUL PENDIDIKAN PEMAKAI

PAKET DASAR
Pengenalan Perpustakaan
dan Sumber-sumber Informasi

Perpustakaan
Universitas Negeri Padang
2007

Paket Dasar
Pengenalan Perpustakaan dan
Sumber-sumber Informasi

- Mengetahui Misi Pendidikan Nasional
- Mengetahui Visi perpustakaan
- Mengetahui Misi perpustakaan UNP
- Mengetahui fungsi perpustakaan
- Mengetahui tugas perpustakaan
- Mengetahui tata letak ruang perpustakaan
- Mengetahui pengorganisasian koleksi
- Memahami penelusuran dan penemuan informasi
- Memahami prosedur peminjaman & pengembalian.

MISI PENDIDIKAN NASIONAL

Mewujudkan pendidikan yang mampu
menghasilkan insan Indonesia yang cerdas
komprehensif dan kompetitif

VISI PERPUSTAKAAN

sebagai pusat jasa akses informasi untuk kepentingan pendidikan dan pengajaran, pengembangan ilmu pengetahuan dan teknologi, budaya dan seni.

MISI PERPUSTAKAAN UNP

Menjadi pusat layanan akses informasi yang handal untuk meningkatkan kualitas pendidikan dan pengajaran, pengembangan ilmu pengetahuan dan teknologi, budaya dan seni dalam rangka membangun bangsa Indonesia yang cerdas komprehensif dan kompetitif-kolaboratif

FUNGSI PERPUSTAKAAN

Perpustakaan sebagai :

- Pusat Informasi
- Pendidikan dan pembelajaran
- Penelitian
- Pengabdian pada masyarakat
- Re-kreasi
- Publikasi
- Pelestarian
- Deposit
- Interpretasi

TUGAS PERPUSTAKAAN

- Perpustakaan bertugas untuk:
- Mengumpulkan informasi
- Mengolah informasi
- Menyebarkan informasi
- Menghasilkan informasi
- Melestarikan Informasi

LETAK DAN ALAMAT PERPUSTAKAAN UNP

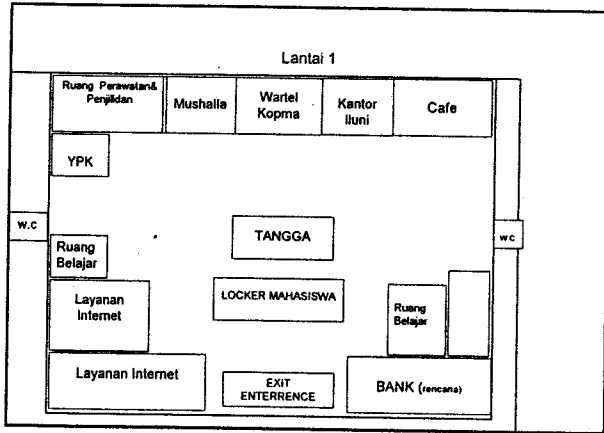
– Perpustakaan UNP terletak di tengah-tengah kampus UNP Air Tawar, dengan batas-batas :

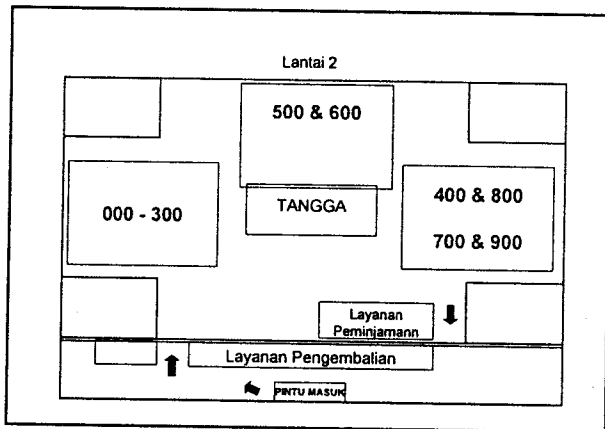
- Sebelah Utara : Labor MIPA
- Sebelah Selatan : Fakultas FBSS, & FIK
- Sebelah Barat : Fakultas FMIPA
- Sebelah Timur : Fakultas FIP, FIS & FE
- Alamat : Gedung Perpustakaan UNP
- Telepon : 0751(7057636) -
- Faksimili : 0751(7057636)
- Email : -

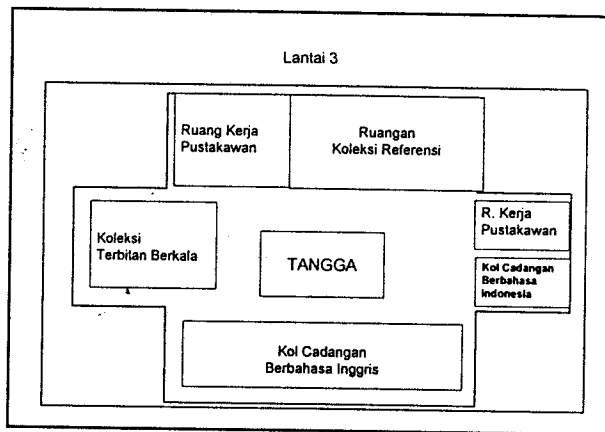
TATA LETAK RUANGAN PERPUSTAKAAN UNP

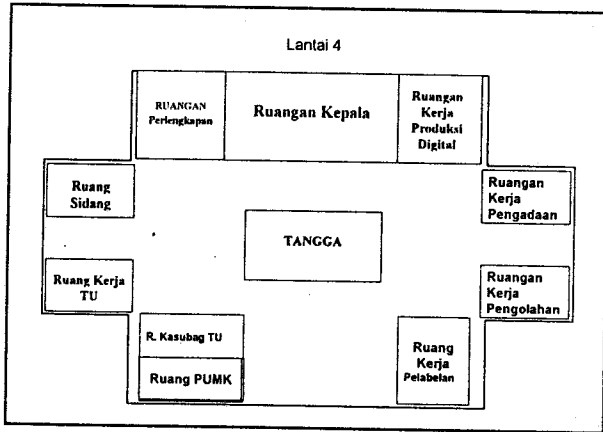
Lokasi Layanan & Fasilitas di Perpustakaan

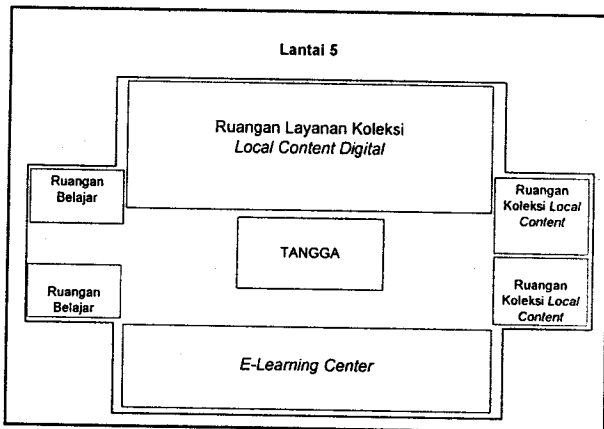
- Lantai 1 : laboratorium komputer, ruangan belajar, kafeteria, mushala, penjilidan, kantor Alumni UNP.
- Lantai 2 : Peminjaman & Pengembalian, koleksi umum, ruang baca & foto copy.
- Lantai 3 : koleksi referensi, koleksi Jurnal, terbitan berkala, cadangan & ruang baca.
- Lantai 4 : Ruang Kepala perpustakaan, Tata usaha, pengadaan, pengolahan & perawatan koleksi.
- Lantai 5 : *local content* (Tesis, Disertasi, laporan penelitian) dan ruangan *E-Learning Center*.











PENGGORGANISASIAN KOLEKSI

Suatu cara penjajaran koleksi perpustakaan dengan menggunakan standar penempatan tertentu, seperti penempatan relatif (*relatif location*) dan penempatan tetap (*fixed location*). Susunan koleksi di rak disusun menurut nomor klasifikasi memakai sistem *Dewey Decimal Classification (DDC)*.

Dewey Decimal Classification (DDC)

Dalam DDC, penyusunan koleksi berdasarkan pada kelompok disiplin ilmu yang telah dibagi dalam sepuluh subjek yaitu:

- >000 Generalities
- >100 Philosophy & Psychology
- >200 Religion
- >300 Social Sciences
- >400 Language & Linguistics
- >500 Natural Science
- >600 Technology (Applied Science)
- >700 Arts

Dewey Decimal Classification (DDC)

• Setiap subjek diuraikan lagi secara rinci seperti contoh dibawah ini:

- 500 → 510 Mathematics 590 → 591 Zoology
- 520 Astronomy 592 Invertebrata
- 530 Physics 593 Protozoa
- 540 Chemistry 594 Mollusa
- 550 Earth Sciences 595 Other Invertebrates
- 560 Paleontology 596 Vertebrates
- 570 Life Sciences 597 Fishes
- 580 Botanical Sciences 598 Reptil and Bird
- 590 Zoological Sciences 599 Mammals

JENIS KOLEKSI PERPUSTAKAAN UNP

Koleksi Umum

Kumpulan koleksi dari berbagai bidang ilmu yaitu : karya umum, filsafat, agama, ilmu-ilmu sosial, ilmu-ilmu murni. Ilmu bahasa, ilmu terapan, olah raga dan seni, kesusasteraan, sejarah dan biografi

KOLEKSI CADANGAN

- Koleksi cadangan juga terdiri dari berbagai bidang ilmu, tetapi koleksi ini khusus untuk tidak dipinjamkan keluar. Koleksi ini adalah satu eksemplar dari semua judul buku yang ada diperpustakaan UNP. Koleksi ini hanya bisa dibaca di tempat atau difoto copy.

KOLEKSI TERBITAN BERKALA

Koleksi yang terbit secara berkala atau periodik (harian, mingguan, bulanan, triwulan, atau tahunan) dengan judul yang sama namun isinya berbeda seperti jurnal ilmiah, buletin, newsletter, majalah, surat kabar dan tabloid. Koleksi ini hanya untuk dibaca di tempat.

KOLEKSI REFERENSI

Koleksi yang dapat membantu pengguna secara cepat untuk mendapatkan informasi yang dibutuhkan. Koleksi ini terdiri dari kamus, ensiklopedia, direktori, buku petunjuk, buku tahunan, biografi, peta, globe dan lain-lain. Koleksi ini hanya untuk dibaca ditempat.

muatan lokal.

KOLEKSI LOCAL CONTENT

- Koleksi hasil karya civitas akademika dari lembaga sendiri seperti karya dosen, mahasiswa dan pustakawan. Koleksi ini terdiri dari tugas akhir, skripsi, tesis dan disertasi mahasiswa. Laporan penelitian, laporan pengabdian pada masyarakat, buku ajar, makalah dari karya dosen dan pustakawan. Koleksi ini berbentuk manual dan digital. Koleksi ini untuk dibaca ditempat dan tidak boleh difoto copi bila tidak seizin yang berwenang.

PENELUSURAN DAN PENEMUBALIKAN INFORMASI

Alat bantu untuk penelusuran informasi adalah :

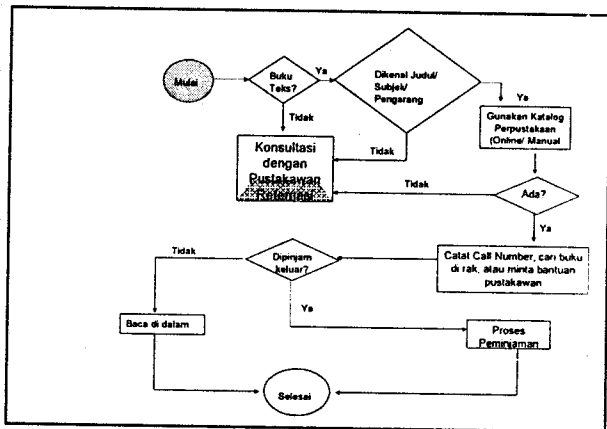
- Katalog kartu
- *Online Public Acces Catalog (OPAC)*
- *Offline Information Systems*
- *WebPac*
- *World-wide Web*
- Bibliografi
- Indeks dan Abstrak.

KATALOG KARTU

Katalog kartu adalah:
wakil dokumen dalam bentuk kartu dengan menggunakan format tertentu yang merujuk pada judul koleksi yang dimiliki oleh perpustakaan setempat. Unsur data terdiri dari pengarang, judul, subjek dan data fisik lainnya.

ONLINE PUBLIK ACCES CATALOG(OPAC)

Online Public Acces Catalog (OPAC) wakil dokumen dalam bentuk *electronic* yang bentuk dan unsur datanya sama dengan katalog kartu tetapi format OPAC tidak selalu sama dengan format katalog.



Katalog Pengarang	
Nomor Panggil	020
Pengarang	Sutarno, NS
Judul	Manajemen perpustakaan: suatu pendekatan praktis/ Sutarno NS.—
Penerbit, Tahun terbit	Jakarta: Sagung Seto, 2006. xviii, 324 him : ill. ; 14 cm.
Tebal Buku	Termasuk bibliografi dan indeks ISBN: 979-984-43-0-4
Subjek	1. Library Science 2. Library Management I. Judul
Tajuk Tambahan	

Katalog Subjek

LIBRARY SCIENCE

020

Sut

m

Sutamo, NS

Manajemen perpustakaan: suatu pendekatan praktis/ Sutamo NS.— Jakarta: Sangung Seto, 2006.

xviii, 324 hlm.: ill.; 14 cm.

Termasuk bibliografi dan indeks

ISBN: 979-984-43-0-4

1. Library Science 2. Library Management
I. Judul

Katalog Judul

Manajemen perpustakaan ...

020

Sut

m

Sutamo, NS

Manajemen perpustakaan: suatu pendekatan praktis/ Sutamo NS.— Jakarta: Sangung Seto, 2006.

xviii, 324 hlm.: ill.; 14 cm.

Termasuk bibliografi dan indeks

ISBN: 979-984-43-0-4

1. Library Science 2. Library Management
I. Judul

INDEKS

- Indeks adalah : daftar subjek atau pengarang yang disusun secara alfabetis dan merujuk ke artikel-artikel yang terdapat pada jurnal ilmiah disebut indek artikel jurnal, atau yang terdapat di majalah disebut indeks artikel majalah atau *index to periodicals* yang mencakup topik atau bidang tertentu.

BIBLIOGRAFI

- Bibliografi adalah : terbitan yang berisi daftar buku, artikel atau karya dari bidang tertentu seperti ; Bibliografi Nasional Indonesia.
- Bibliografi beranotasi memberikan informasi singkat/ringkas tentang karya tersebut

ABSTRAK

- Abstrak adalah terbitan yang berisi daftar buku, artikel atau karya lainnya yang dilengkapi dengan ringkasan/intisari karya yang terdaftar.
- Publikasi tersebut dilengkapi data bibliografi sehingga karya itu mudah diidentifikasi kembali.
- Susunan : judul karangan, nama penulis (tanpa gelar), nama instansi, kata kunci/keyword, nama penyari maupun data bibliografi, dan sari karangan.

LAYANAN SIRKULASI

PROSEDUR PEMINJAMAN

- Menelusur melalui katalag kartu atau OPAC
- Mencari kolesi di rak
- Mengisi kartu buku (nama, nim dan tanda tangan).
- Memberikan kartu anggota perpustakaan pada pustakawan.
- Peminjaman diproses, kartu peminjaman tinggal sebagai bukti peminjaman.

PROSEDUR PENGEMBALIAN

- Menulis nama, nim dan tanggal kembali buku.
- Memberikan buku yang dipinjam
- Pengembalian diproses, selesai, kartu anggota perpustakaan dikembalikan pada mahasiswa.

LAYANAN INFORMASI

- Mencakup semua bentuk layanan yang bertujuan membantu pengguna mengakses informasi baik yang ada di perpustakaan UNP, maupun tidak. Layanan ini mencakup layanan referensi, penelusuran informasi, konsultasi dan layanan paket informasi.

PERPUSTAKAAN MERUPAKAN PUSAT PEMBELAJARAN

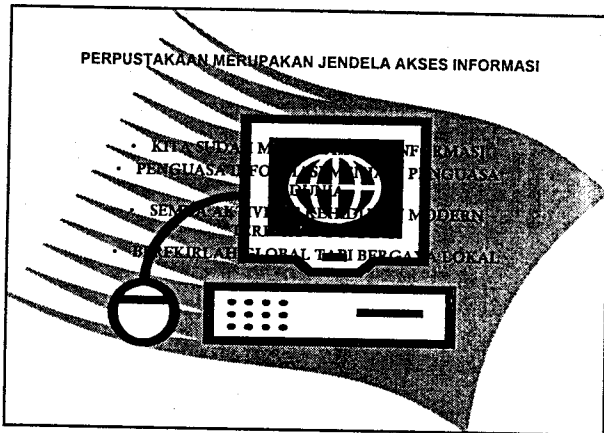
- Dosen adalah fasilitator dan pengarah
- Kemandirian belajar adalah kunci keberhasilan belajar di perguruan tinggi
- Perpustakaan merupakan sahabat belajar Anda, sabar menanti, setia melayani.
- Ia adalah lautan ilmu yang tak pernah kering sampai akhir zaman

JAM LAYANAN

- Senin- Jumat, jam 8.00 wib - 18.00 wib
- Sabtu dan Minggu, jam 8.00 wib – 13.00 wib

- PELAYANAN
- PERPUSTAKAAN UNP
"NON STOP"

PERPUSTAKAAN MERUPAKAN JENDELA AKSES INFORMASI



Delmas

L - utam

- dimensi yang = 20 k

- ijazat - kegunaan

Arnold —>

- Ref → jml

- ref & internet

Idi:

Tu - Perwujudan fungsi

ME - melayani klf yg (Kolon. →
- skripsi

Idi - faktor

ME ke & perdisi.

UNIVERSITAS NEGERI PADANG

SLIDE PERKULIAHAN DASAR-DASAR PEMOGRAMAN KOMPUTER

FIS006 3 SKS



PAKHRUR RAZI, M.SI

NO. SURAT PERMOJONGAN	158/HA/2009 - d1 (1)
TANGGAL	2-5-2009
KELOMPOK	HAHAH
KELOMPOK	K1
KELOMPOK	158/HA/2009 - d1 (1)
KELOMPOK	005.3 RAZ d.1

**JURUSAN FISIKA FMIPA
UNIVERSITAS NEGERI PADANG**

© 2008

An Introduction to Programming

Basic understanding

by: Fahrur Razi, M. Si

Penggunaan komputer dewasa ini

- Ekonomi / bisnis
- Pendidikan
- Kesehatan
- Telekomunikasi
- Pertahanan Keamanan
- Game / Hiburan
- dll

APAKAH KOMPUTER BENAR-BENAR MESIN YANG BERTU PINTAR ?

ATURAN UTAMA :

COMPUTER IS JUST A MACHINE
IT WILL DO NOTHING IF WE NOT
GIVE INSTRUCTION TO IT

Pemrograman ???

Langkah-langkah terstruktur
untuk memberikan instruksi
kepada komputer

How to instruct computer ?

- Using Indonesian language ??
- Using English ??
- Using Arabics ??
- Using Madura language ??
- Using Java language ??

YOUR COMPUTER WILL STAY IN QUITE

WHY ?

Because the computer do not understand what you said / instructed to it.

HOW CAN WE INSTRUCT ?

Use the language that computer understood

Bahasa komputer ??

• BAHASA MESIN :

--> 1100 0011 1001 0000 0101

Sulit ?

Tentu saja

• BAHASA ASSEMBLY

--> CMP AL, OD = 0011 1010 0000 1011

Bahasa generasi ke 3 / 3GL / HLL

--> Bahasa pemrograman yang menggunakan bahasa Inggris dan kode-kode yang lebih bisa difahami manusia

--> Sebenarnya terdiri dari puluhan bahkan ratusan instruksi dalam bahasa mesin

Contoh HLL :

- Basic
- Pascal
- C
- C ++
- COBOL
- Dll

Software yang menterjemahkan bahasa manusia ke dalam bahasa mesin / assembly ada 2 macam :
(1) Compiler dan
(2) Interpreter

- **INTERPRETER :**

Menterjemahkan baris per baris baru dan mengeksekusi perintah per baris
contoh : Basic

- **COMPILER :**

menterjemahkan secara keseluruhan baru mengeksekusi perintah
contoh : Pascal, C, C++

Langkah-langkah pemrograman

- Mendefinisikan masalah
- Menentukan solusi
- Memilih algoritma
- Menulis program
- Menguji program
- Menulis dokumentasi
- Merawat program

1. Mendefinifikan Masalah

- Apa permasalahan yang harus dipecahkan oleh komputer
 - Apa saja input yang tersedia
 - Apa input dari program yang akan dibuat
 - Bagaimana output dari program
- Contoh : Program mencari luas lingkaran
- Permasalahan : Mencari luas lingkaran
 - Input : Jari-jari lingkaran
 - Output : gambar dan nilai luas lingkaran

2. Menentukan Solusi

- Setelah input dan output sudah jelas
- Mencari jalan bagaimana permasalahan tersebut diselesaikan
- Jika program terlalu kompleks dapat dibagi menjadi beberapa modul.

Contoh : Program mencari inverse matriks

- Permasalahan : Mencari inverse matriks
- Modulasi : (1) input data, (2) perhitungan dan (3) menampilkan output

3. Memilih Algoritma

- Pada hakikatnya banyak cara untuk menyelesaikan suatu permasalahan
- Merupakan tahap yang vital dalam pemrograman
- Menentukan kinerja program yang akan dibuat

4. Menulis program

- Menuliskan instruksi-instruksi kepada komputer melalui bahasa pemrograman tertentu.
- Pertimbangan penguasaan bahasa dan aplikasi dari program yang akan dibuat.
- Pertimbangan trade off antara kecepatan melakukan pemrograman dengan biaya.
Untuk kuliah ini kita akan menggunakan bahasa TPW (Turbo Pascal for Windows)

5. Menguji program

- Meneliti :
 - (1) Apakah instruksi telah dituliskan dengan benar
--> Menggunakan fasilitas compile
 - (2) Menguji kapasitas dan keandalan program yang akan dibuat
--> mengujinya dalam berbagai kasus

6. Menulis dokumentasi

- Menuliskan kegunaan sebuah instruksi /bagian/modul dll dalam sebuah program

Manfaat :

- Memudahkan tracking program
- Memudahkan koreksi / modifikasi program
- Memudahkan perawatan program
- Share knowledge

7. Merawat Program

- Dilakukan setelah program selesai dibuat
- Mengantisipasi munculnya bug yang tidak terdeteksi sebelumnya
- melakukan modifikasi / menambahkan fasilitas baru.

Elemen Program Pascal

A. Simbol-simbol dasar

- Program pascal dapat dibentuk dari simbol-simbol yang terdiri dari huruf-huruf, angka-angka dan simbol-simbol khusus
- Setiap simbol yang digunakan akan mempunyai konsekuensi tertentu yang berlainan, terutama dalam kaitannya dengan jenis data, reserved words dan operator.
- Simbol dasar dalam pascal terdiri dari 3 macam yaitu :
 - Huruf-huruf
 - huruf yang dapat digunakan adalah huruf A - Z, a - z dan underscore (_)
 - Dalam pascal huruf besar dan huruf kecil dianggap sama
 - Angka-angka
 - Angka-angka yang dapat dipergunakan dalam pascal adalah angka 0 1 2 3 4 5 6 7 8 9 (0 - 9)
 - Simbol khusus
 - Simbol khusus yang dapat digunakan adalah sebagai berikut
* , / = ^ () [] () . . . ; : # \$
< = > = := (') (,)

2

Pemrograman Komputer

Tujuan Perkuliahan

- Setelah mengikuti kelas ini anda diharapkan mampu :
- Memahami dan membedakan berbagai simbol dasar dalam pascal
 - Memahami yang dimaksud dengan reserved words
 - Memahami apa yang dimaksud dengan identifier serta mampu membuat identifier yang baik
 - Memahami berbagai tipe data yang dikenal di dalam pascal (serta beberapa bahasa pemrograman yang lain)
 - Membuat deklarasi tipe data dalam pascal
 - Memahami tanda-tanda operasi dalam pascal
 - Memahami apa itu komentar program dan apa itu statement
 - Menjelaskan berbagai jenis statement yang ada dalam pascal

3

Pemrograman Komputer

B. Reserved Words

- Reserved words atau kata cadangan adalah kata-kata yang sudah didefinisikan oleh program pascal dan mempunyai maksud tertentu.
- Kata-kata cadangan tidak boleh didefinisikan sendiri oleh pemakai (pemrogram) sebagai sebuah identifier.
- Di bawah ini adalah kata-kata cadangan yang ada dalam pascal

4

Pemrograman Komputer

C. Identifier

- Identifier adalah suatu nama yang dipergunakan dalam program pascal.
- Pemrogram dapat mendefinisikan sendiri suatu nama sebagai identifier untuk
 - Tipe
 - Konstanta
 - Variabel
 - Label
 - Prosedur
 - Fungsi
 - Unit
 - Field dalam record

Pemrograman Komputer

5

Syarat identifier

- Terdiri atas gabungan huruf dan atau angka dengan karakter pertama harus berupa huruf
- Huruf besar dan huruf kecil dianggap sama
- Tidak boleh mengandung blank (spasi)
- Tidak boleh mengandung simbol khusus
- Panjang bebas tapi hanya 63 karakter pertama yang dianggap signifikan
- Contoh identifier :

Pemrograman Komputer

6

D. Tipe data

- Jika anda mempergunakan identifier variabel dalam program, maka variabel tersebut harus anda deklarasikan lebih dahulu.
- Saat melakukan deklarasi anda harus menentukan tipe data variabel tersebut.
- Tipe data ini menunjukkan suatu nilai dari variabel bersangkutan.
- Tipe data dalam turbo pascal ada beberapa macam yaitu :
 - Tipe data standar
 - Tipe data didefinisikan oleh pemrogram
 - Tipe data terstruktur
 - Tipe data penunjuk (pointer)
- Dalam slide ini akan dijelaskan tipe data standar dalam pascal

Pemrograman Komputer

7

Tipe Bilangan Bulat (Integer)

- Merupakan nilai bilangan bulat baik dalam bentuk desimal maupun hexadesimal.
- Pascal menyediakan 5 jenis data integer yang masing masing mempunyai jangkauan nilai yang berbeda yaitu :

Tipe	Jangkauan	Ekstensi

Urut data tipe Integer ini, turbo pascal juga menyediakan konstanta standar bernama *MaxInt* yang bernilai 32767 dan *MaxLongInt* yang bernilai 2147483647.

Kedua konstanta ini dapat digunakan dalam program tanpa harus didefinisikan terlebih dahulu.

Pemrograman Komputer

8

Tipe Real

- Jenis tipe data yang dapat memuat variabel yang berupa nilai numerik real.
- Nilai data tipe ini berkisar antara 1E-38 sampai dengan 1E+38 dengan lebar hingga 11 digit
- E menunjukkan nilai 10 pangkat. Nilai numerik real menempati memori sebesar 6 byte
- Jenis tipe data real adalah sebagai berikut :

Tipe	Lebar data	Lebar	Indikator

Penrograman Komputer

9

Tipe Boolean

- Jenis tipe data yang digunakan untuk menyimpan variabel yang hanya mempunyai nilai benar atau salah
- Hanya ada 2 nilai yang bisa diberikan pada variabel dengan tipe data boolean yaitu *true* dan *false*

Cara deklarasi
 Var
 x : boolean;
 contoh pemberian nilai
 x := true;
 x := false;

Penrograman Komputer

10

Tipe Karakter (Char)

- Jenis tipe data yang digunakan untuk menyimpan sebuah data alfanumeris (karakter) diantara 2 tanda petik seperti 'A', 'a', 'Z', '@', '\$', '!', '9', '-', dll.
- Ada 2 cara untuk memberi nilai pada tipe data karakter ini yaitu :
 - Menuliskan karakter dalam tanda petik tunggal
 - Menuliskan tanda # diikuti kode ASCII dari karakter yang ingin ditulis

Penrograman Komputer

11

Tipe String

- Jenis tipe data yang digunakan untuk menyimpan rangkaian data alfanumeris (karakter) diantara 2 tanda petik seperti 'Adalah', '12-12-2004' dan lain-lain.
- Bila tanda petik merupakan bagian dari konstanta string maka dapat ditulis dengan menggunakan dua buah petik tunggal berurutan ...
- Nilai data string akan menempati memori sebesar banyaknya karakter string ditambah dengan satu byte.
- Bila panjang string tidak didefinisikan dalam deklarasi maka dianggap panjangnya adalah 255 karakter.

Penrograman Komputer

12

Tipe Pointer

- Pointer adalah suatu variabel yang berisi alamat (address) di memori (RAM) dimana suatu data disimpan, dan bukan berisi data itu sendiri. Dengan kata lain, pointer akan menunjukkan letak dan data di memori.
- Contoh

```
Type
  TipeString = string(40);
  PointerString = ^tipestring;
  Var
  LetakNama : pointerstring;
  Begin
  Letaknama = Turbo Pascal;
  WriteLn(LetakNama);
  End.
```

- Jika program ini dijalankan maka akan ditampilkan isi dan memori di alamat yang ditunjukkan oleh pointer letaknama dan diperoleh output sebagai berikut.

Turbo Pascal

13
Pemrograman Komputer

Tanda Operasi

- Tanda operasi dalam bahasa pascal dikelompokkan menjadi :
 - Assignment operator
 - Binary operator
 - Unary operator
 - Bitwise operator
 - Relational operator
 - Address operator
 - Set operator
 - String operator

14
Pemrograman Komputer

- Assignment operator
 - Merupakan simbol titik dua diikuti oleh simbol sama dengan (=)
- Binary operator
 - Disebut binari karena operator ini digunakan untuk mengoperasikan dua buah operand.
 - Operand dapat berupa konstanta ataupun variabel.
 - Operator ini digunakan untuk operasi aritmatika yang berhubungan dengan tipe data integer dan real.

Operator	Contoh	Tipe operand	Tipe hasil

15
Pemrograman Komputer

- Unary operator
 - Disebut unary operator karena operator ini hanya menggunakan sebuah operand saja.
 - Unary operator dapat berupa unary minus atau unary plus
 - Contoh : -5, -2.5, +15 dll
- Bitwise operator
 - Digunakan untuk operasi bit per bit pada nilai integer
 - Jenis-jenis operator ini adalah sebagai berikut

Operator	Contoh	Tipe operand	Tipe hasil

16
Pemrograman Komputer

◊ Relational Operator

- Digunakan untuk membandingkan antara dua buah operand
- Hasil operasi dengan operator ini akan mendapatkan hasil bertipe boolean yaitu *True* atau *false*
- Jenis-jenis operator hubungan adalah sebagai berikut :

Operator	Contoh

◊ Address operator

- Turbo pascal memberikan dua buah operator khusus yang berhubungan dengan alamat (address) di memori yaitu :
 - ◊ Address of operator (&) → menghasilkan alamat dari suatu variabel
 - ◊ Indirect operator (^) → memberikan nilai di alamat yang ditunjukkan

◊ Set operator

- Operator ini digunakan untuk operasi himpunan
- Operator ini terdiri dari :



◊ String operator

- Digunakan untuk operasi string
- Terdiri dari satu buah operator saja yaitu operator +
- Contoh program

```

Var
  nama1, nama2, nama3 : string(15);
Begin
  nama1 = 'budiman';
  nama2 = 'sujatmiko';
  nama3 = nama1 + nama2;
  writeln (nama3);
End
    
```

Jika dieksekusi hasilnya adalah :
Budiman sujatmiko

Komentar

- ◊ Komentar adalah tulisan dalam listing program tapi tidak termasuk dalam program (tidak ikut dieksekusi).
- ◊ Fungsinya hanya untuk dokumentasi yang memudahkan pemrogram ketika memeriksa atau memperbaiki programnya.
- ◊ Penulisan komentar dapat dilakukan didalam tanda kurung kurawal { } atau (* *)

Statement

- ◊ Statement adalah perintah pengerjaan program
- ◊ Kumpulan dari statemen membentuk program
- ◊ Statemen terletak di bagian statement dengan diawali dengan kata begin dan ditutup dengan kata end.
- ◊ Masing-masing statement harus diakhiri dengan titik koma (;)
- ◊ Dalam pascal, statemen dapat berupa statement sederhana atau statement terstruktur.

Statement

- Statement sederhana terdiri dari :
 - Statement pengerjaan
 - Statement prosedur
 - Statemen goto
- Statement tersruktur terdiri dari :
 - Statemen jamak
 - Statement penyeleksian kondisi
 - Statemen perulangan

21

Pemrograman Komputer

Two: Algoritma dan Flowchart

How to Think

DASAR PEMROGRAMAN
by: Fakhur Razi, M.Si

2

Objectives

Setelah menyelesaikan bab ini, anda diharapkan dapat:

- Mengerti tentang algoritma.
- Membuat algoritma dari suatu permasalahan.
- Mengerti tentang flowchart.
- Membuat flowchart dari suatu permasalahan.

Tahapan Pembuatan Program

- Mengerti permasalahan yang akan diselesaikan.
 - Input layout
 - Proses-proses
 - Output layout
- Menganalisa penyelesaian masalah.
- Membuat algoritma dan flowchart.
- Menulis program.
- Melakukan testing → program selesai.
- Membuat dokumentasi

3

Algoritma

- Algoritma adalah gambaran langkah-langkah dari penyelesaian suatu masalah.
- Problem = langkah-langkah penyelesaian.
- Algoritma adalah blueprint dari program.
- Sebaiknya disusun sebelum membuat program.

Contoh Algoritma 1

Mengirim surat kepada teman:

1. Tulis surat pada secarik kertas surat.
2. Ambil sampul surat.
3. Masukkan surat ke dalam sampul.
4. Tutup sampul surat menggunakan perekat.
5. Jika kita ingat alamat teman tersebut, maka tulis alamat pada sampul surat.
6. Jika tidak ingat, lihat buku alamat, kemudian tulis alamat pada sampul surat.
7. Tempel perangko pada surat.
8. Bawa surat ke kantor pos untuk diposkan.

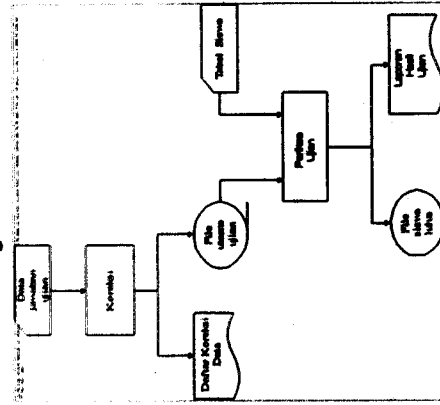
5

Flowchart

- Bagian-bagian yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah.
- Merupakan cara penyajian dari suatu algoritma.
- Ada 2 macam Flowchart:
 - System Flowchart → urutan proses dalam system dengan menunjukkan alat media input, output serta jenis media penyimpanan dalam proses pengolahan data.
 - Program Flowchart → urutan instruksi yang digambarkan dengan symbol tertentu untuk memecahkan masalah dalam suatu program.

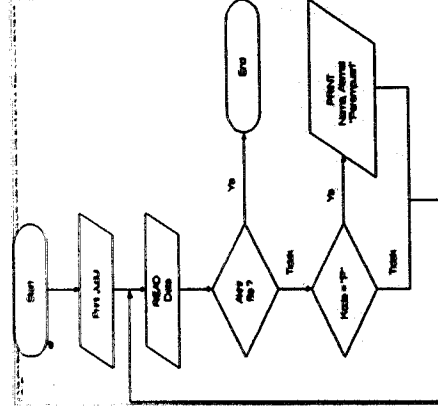
6

Contoh System Flowchart



7

Contoh Program Flowchart




8


Symbol-simbol Flowchart


- *Flow Direction Symbols* (Simbol penghubung alur)
- *Processing Symbols* (Simbol proses).
- *Input-output Symbols* (Simbol input-output)

9

Flow Direction Symbols

- 


Simbol arus /flow
Untuk menyatakan jalannya atau suatu proses
- 


Simbol Connector
Untuk menyatakan sambungan dan satu proses ke proses lainnya dalam halaman tersebut yang sama
- 


Simbol Offline Connector
Untuk menyatakan sambungan dan satu proses ke proses lainnya dalam halaman tersebut yang berbeda


10

Processing Symbols

- 

Simbol Process
Untuk menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.
- 


Simbol Manual
Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual).
- 


Simbol Decision / logika
Untuk menunjukkan suatu kondisi tertentu yang akan mengakibatkan dua kemungkinan jawaban, ya / tidak.
- 


Simbol Predefined Proses
Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.


11

Processing Symbols - continued

- 

Simbol Terminal
Untuk menyatakan pemitahuan atau akhir suatu program
- 

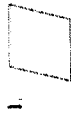
Simbol Keying Operation
Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard
- 

Simbol off-line storage
Untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
- 

Simbol Manual Input
Untuk menyatakan data secara manual dengan menggunakan online keyboard

12

Input-output Symbols



Simbol Input-output
Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatnya.



Simbol Punched Card
Untuk menyatakan input berasal dari kartu atau output ditulis ke kartu.



Simbol Magnetic-tape unit
Untuk menyatakan input berasal dari pita magnetic atau output dituliskan ke pita magnetic.

13

Input-output Symbols - continued



Simbol Disk storage
Untuk menyatakan input berasal dari disk atau output dituliskan ke disk.



Simbol Document
Untuk mencetak laporan ke printer.



Simbol Display
Untuk menyatakan peralat output yang digunakan berupa layar (video, komputer).

14

Pembuatan Flowchart

- Tidak ada kaidah yang baku.
- Flowchart = gambaran hasil analisa suatu masalah → Flowchart dapat bervariasi antara satu pemrogram dengan pemrogram lainnya.
- Secara garis besar ada 3 bagian utama:
 - Input
 - Proses
 - Output

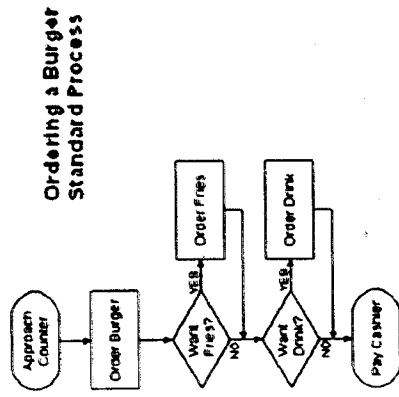
15

Pembuatan Flowchart - continued

- Hindari pengulangan proses yang tidak perlu dan logika yang berbelit sehingga jalannya proses menjadi singkat.
- Jalannya proses digambarkan dari atas ke bawah dan diberikan tanda panah untuk memperjelas.
- Sebuah flowchart diawali dari satu titik START dan diakhiri dengan END.

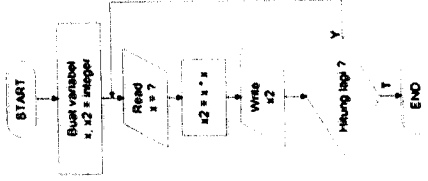
16

Another Flowchart Example



17

Again Another Example



18

Exercise

1. Buatlah algoritma untuk menghitung luas dan keliling lingkaran. Dengan masukan berupa jari-jari.
2. Buatlah flowchart dari algoritma pada soal no 1 diatas.
3. Buatlah program untuk flowchart no 2.

19

Exercise - continued

4. Buatlah algoritma seperti soal no 1, dengan menambahkan pilihan untuk mencoba lagi atau tidak.
5. Buat flowchart dari algoritma pada soal no 4 diatas.
6. Buatlah program untuk flowchart no 5.

20

Input dan Output

Materi Minggu ke-4

by Fakhur Razi, M Si

Sasaran Belajar

- Setelah mengikuti kuliah ini anda diharapkan mampu :
 - Menerapkan teknik-teknik memasukkan data dalam berbagai tipe data
 - Mengetahui tampilan default berbagai tipe data
 - Mengeset tampilan terformat melalui penentuan parameter tipe data
 - Menggunakan beberapa prosedur standar untuk mengatur tampilan (output) di layar.

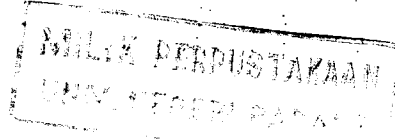
Pendahuluan

- Setiap program aplikasi pasti memerlukan operasi input dan output
- Memasukkan data input secara interaktif merupakan ciri program aplikasi yang banyak ditemukan.
- Suatu program yang tidak dapat menghasilkan output maka proses program tersebut akan sia-sia.
- Tampilan sebuah program yang baik adalah yang mudah difahami oleh penggunaanya (user friendly) serta tidak menimbulkan salah pengertian.
- Keindahan dan estetika tampilan program merupakan nilai lebih

Memasukkan Data

- Cara yang paling banyak digunakan untuk memasukkan data adalah dengan mengetikkannya melalui keyboard.
- TPW menyediakan prosedur standar *Read* dan *Readln*
- Prosedur *read* dan *readln* mempunyai aturan tertentu untuk tiap tipe identifier variabel.
- Tiap-tiap data yang dimasukkan harus sesuai dengan tipe variabelnya.

Note : Masih ingat penjelasan mengenai perbedaan *read* dan *readln* ?



Tipe Var & Datanya

Tipe variabel dan data yang diijinkan

Type Variabel	Keterangan
Char	Memasukkan sebuah karakter, bisa lebih atau lebih sedikit yang dianggap hanya yang pertama
String	Memasukkan string maksimum sepanjang yang diinginkan
Integer	Memasukkan data numerik bulat diantara nilai -32768 s/d 32767
Word	Memasukkan data numerik bulat antara 0 - 65535
Byte	Memasukkan data numerik bulat antara 0 - 255
Shortint	Memasukkan data numerik bulat antara -128 - 127
Longint	Memasukkan data numerik bulat antara -2147483648 - 2147483647
Real	Memasukkan data numerik real maksimum 30 digit dapat tanpa desimal
Boolean	Data tipe ini tidak diijinkan

Try this

Cobalah program dibawah ini :

```

Program masuk,
Uses winct,
Var
  A, B : Integer,
  C : Real,
  D : String[10],
  E : Char,
Begin
  (* memasukkan data untuk masing-masing variabel *)
  Readln(A), Readln(B), Readln(C), Readln(D), Readln(E),
  (* menampilkan data yang telah dimasukkan *)
  Writeln,
  Writeln ('A = ', A, ', B = ', B, ', C = ', C),
  Writeln ('D = ', D),
  Writeln ('E = ', E),
  Readln,
End
Jalankan dan amati hasilnya untuk masing-masing tipe data
  
```

Teknik Memasukkan Data Type Boolean

- Telah disebutkan di slide ke 5 di atas bahwa input data boolean tidak diperbolehkan.
- Akan tetapi hal ini bisa diatasi dengan cara memasukkan data dengan tipe char terlebih dahulu baru kemudian dilakukan pengecekan boolean.

- Try and learn this

```

Program Memasukkanboolean;
Uses winct;
Var
  jwb_c : char;
  jwb_b : boolean;
Begin
  Writeln('Masukkan atau salah (B/S) ? ');
  Readln(jwb_c);
  jwb_b := (jwb_c = 'B');
  Writeln('yang anda maksud adalah : ', jwb_b);
End
  
```

Menampilkan Hasil

- Untuk menampilkan hasil dalam TPW dapat dilakukan dengan menggunakan prosedur standar write atau writeln.
- Masih ingat perbedaan antara write dan writeln ?
- Betul, prosedur write berfungsi untuk menampilkan hasil tanpa ganti baris dan tampilan berikutnya akan akan disambung dalam baris yang sama.
- Sedangkan prosedur writeln berfungsi untuk menampilkan hasil dan ganti baris, sehingga tampilan berikutnya akan ditampilkan di baris dibawahnya.

Default Tampilan

Type Data	Format tampilan default	Keterangan
Integer	dddd	Tdk mengandung blank di muka maupun dibelakang
Real	+	b = blank d = digit
	-	s = tanda positif / negatif - = tanda negatif Lebar: 18 digit bentuk eksponensial
Char	c	Tdk mengandung blank di muka maupun dibelakang
String	sss (sesuai panjang isinya)	Tdk mengandung blank di muka maupun dibelakang
Boolean	True atau False	Tdk mengandung blank di muka maupun dibelakang

Tampilan Terformat

- Kadang kala di program aplikasi, memerlukan bentuk tampilan dengan format tertentu yang harus diatur kembali.
- Pada saat mengatur format tampilan dengan prosedur write atau writeln dipergunakan parameter-parameter tampilan.
- Cara pengaturan / penulisan parameter tampilan berbeda-beda tergantung tipe data variabel yang digunakan

PENGULANGAN(LOOPING)

Materi Minggu ke-5

by Fakhur Razi, M.Si

PERULANGAN FOR-DO

Looping for-do digunakan untuk mengulang statement berulang kali sejumlah yang ditentukan.

Bila terdapat lebih dari satu statement yang ada di dalam looping maka menggunakan begin...end;, jika hanya satu blok saja tidak perlu menggunakan begin...end;.

Ada 3 bentuk looping for-do yaitu for-do positif, for-do negatif dan for-do tersarang.

Looping Positif/Negatif

- Looping positif adalah looping dari perhitungan kecil ke perhitungan besar.
- Sedangkan looping negatif kebalikan dari looping positif. Bentuknya sebagai berikut :

For to do/for downto do

```
For var_int := nilai_awal to nilai_akhir do  
statement; (+)
```

```
for := ..to..do
```

```
For var_int := nilai_awal downto nilai_akhir do  
statement; (-)
```

```
for:=..downto..do
```

Var_int adalah variabel kontrol yang menentukan looping tersebut. Variabel kontrol harus bertipe sejenis integer

Contoh

```

program looping_with_begin;
uses winctrl;
var x:integer;
begin
  For x:= 5 downto 1 do
  Begin
    writeln(' Sixiz gitu lho..!!');
  end;
End.

Output program looping_with_begin :
5 Sixiz gitu lho..!!
4 Sixiz gitu lho..!!
3 Sixiz gitu lho..!!
2 Sixiz gitu lho..!!
1 Sixiz gitu lho..!!

```

Contoh

```

program looping_without_begin;
uses winctrl;
var x:integer;
begin
  For x:= 1 to 5 do writeln('Sixiz gitu lho..!!');
end.

Output program looping_without_begin :
Sixiz gitu lho..!!
Sixiz gitu lho..!!
Sixiz gitu lho..!!
Sixiz gitu lho..!!
Sixiz gitu lho..!!

```

Analisis Output

```

program mikir_donk;
Uses winctrl;
var x,y:byte;
begin
for x:= 10 downto 1 do
  y:=y+(x-1);
  writeln(x);
end.

```

Coba perhatikan listing program berikut

```

program yang_mana_bukan_looping;
uses winctrl;
var x:integer;
begin
  For x:= 5 downto 1 do
    write(x);
    writeln(' Sixiz gitu lho..!!');
  end.
Outputnya adalah
54321 Sixiz gitu lho..!! Statement writeln(' Six...')
tidak termasuk ke dalam looping karena looping x berakhir
pada satement write(x);

```


Default Tampilan

Type Data	Format tampilan default	Keterangan
Integer	dddd	Tdk mengandung blank di muka maupun dibelakang b = blank d = digit
Real	+ b.d ddddddEsd	s = tanda positif / negatif .. = tanda negatif Lebar 18 digit.
	-d. ddddddEsd	Bentuk eksponensial
Char	c	Tdk mengandung blank di muka maupun dibelakang
String	sss (sesuai panjang isinya)	Tdk mengandung blank di muka maupun dibelakang
Boolean	True atau False	Tdk mengandung blank di muka maupun dibelakang

Tampilan Terformat

- Kadang kala di program aplikasi, memerlukan bentuk tampilan dengan format tertentu yang harus diatur kembali.
- Pada saat mengatur format tampilan dengan prosedur write atau writeln dipergunakan parameter-parameter tampilan.
- Cara pengaturan / penulisan parameter tampilan berbeda-beda tergantung tipe data variabel yang digunakan

Four: Statemen CASE

How to Think

DASAR PEMROGRAMAN
by: Fakhur Razi, M.Si

2

STATEMEN CASE

- Statemen case dapat berbentuk struktur case of atau case of..else.
- Pernyataan CASE dipergunakan apabila kita menulis program yang memerlukan banyak pernyataan IF...THEN...ELSE, sehingga program akan tampak lebih jelas untuk dibaca
- Struktur Case – Of mempunyai suatu ungkapan logika yang disebut dengan selektor dan sejumlah statemen yang diawali dengan suatu label permasalahan (case label) yang mempunyai tipe sama dengan selektor.

MILIK PERPUSTAKAAN
UNM. NEGERI PADANG

STATEMEN CASE

- Statement yang mempunyai case label yang bernilai sama dengan nilai selektor akan diproses sedang statemen yang lainnya tidak.
- Case-of dapat dikatakan sebagai peluasan dari struktur IF. Karena kalau dalam struktur IF hanya disediakan dua pilihan (berdasarkan kondisi logikanya), maka dalam struktur Case-of dimungkinkan untuk memilih satu pilihan di antara banyak pilihan yang ada.

3

CASE OF

Bentuk struktur case-of :
CASE ungkapan OF
Daftar case label1 : statemen1 ;
Daftar case label2 : statemen2 ;
...
...
Daftar case label n : statemen n ;
END ;

4

CASE OF

- Ekspresi yang digunakan dalam statemen Case adalah yang mempunyai tipe ordinal yaitu dengan batas antara (-32768 s/d 32767). Sehingga tipe lain seperti integer yang bertipe longint, tipe string atau word tidak boleh digunakan
- Daftar Case label dapat berupa konstanta, range dari konstanta yang bukan bertipe real

Contoh:

```
1: (nilai integer 1)
1,2,3,4: (nilai integer 1)
1..5: (nilai integer 1→5)
'A': (nilai karakter 'A')
'A','B': (nilai karakter A dan B)
'A'..'D': (nilai karakter A → D)
...: (nilai karakter '')
```

5

Contoh

```
Contoh program :
Program nilai;
Var
  nilai : Char;
Begin
  Write ('Nilai huruf yang didapat : '); Readln (nilai);
  Case nilai Of
    'A': Writein ('Sangat Baik');
    'B': Writein ('Baik');
    'C': Writein ('Cukup');
    'D': Writein ('Kurang');
    'E': Writein ('Sangat Kurang');
  End;
End.
```

6

Contoh

```
Uses Cr1;
Var
  Pilihan : byte;
  R, L, T, Luas : real;
Begin
  Clrscr;
  GotoXY (10,2); writein ('Pilihan');
  GotoXY (10,4); writein ('1. Menghitung Luas Lingkaran');
  GotoXY (10,6); writein ('2. Menghitung Luas Segitiga');
  GotoXY (10,8); writein ('3. Menghitung Luas Bujur Sangkar');
  GotoXY (10,20); writein ('Pilih nomor(1-3) = '); readln (Pilihan);
  Clrscr;
  Case Pilihan of
    1 : Begin
      write ('Jari lingkaran = '); readln (R);
      Luas := Pi * R * R;
      Writein ('Luas lingkaran =', Luas : 9 : 2);
      End;
    2 : Begin
      write ('Panjang sisi alas = '); readln (L);
      write ('Tinggi segitiga = '); readln (T);
      Luas := 0.5 * L * T;
      Writein ('Luas segitiga =', Luas : 9 : 2);
      End;
  End;
```

7

Contoh

```
3. Begin
  write ('Panjang bujur sangkar = '); readln (T);
  write ('Lebar bujur sangkar = '); readln (L);
  Luas := T * L;
  Writein ('Luas bujur sangkar =', Luas : 9 : 2);
  End;
Else
  Begin
  Writein ('Pilihannya hanya 1, 2, dan 3 !!!');
  Writein ('Silahkan ulangi lagi');
  End;
End.
```

8

Contoh

```
Program_Konversi_nilai;
Uses CrI;
Var Nilai : integer;
Begin
  Clrscr;
  write('Baca nilai =');readln(Nilai);
  Case Nilai of
    0..25 : writeIn('Nilainya = E');
    26..39 : writeIn('Nilainya = D');
    40..64 : writeIn('Nilainya = C');
    65..79 : writeIn('Nilainya = B');
    80..100: writeIn('Nilainya = A');
  else
    writeIn(' nilai hanya sampai 100 omm');
  end;
  readln;
end.
```

9

```
Program konversi;
Uses winct;
Var Celcius, Fah Reel;
    Lagi Char;
Begin
  Lagi = 'Y';
  While lagi = 'Y' Do
  Begin
    Clrscr;
    Write ('Nilai derajat Celcius ? ');
    Readln(Celcius);
    Fah := 1,8 * Celcius +32;
    WriteIn(' Maka Fahrenheit = ', Fah, 8,2;
    Derjat);
    WriteIn(' Mengulang Lagi (Y/N)? ');
    Readln(Lagi);
  End;
End;
```

```
program OPERATOR;
uses WinCrt;
Label 10;
var R,T:isi : real;
Jawab char;
begin
  10. writeIn('Masukan dua buah bilangan');
  write(' bilangan 1 ');readln(bilangan1);
  write(' operator ');readln(operator);
  write(' bilangan 2 ');readln(bilangan2);
  case operator of
    '+': result := bilangan1 + bilangan2;
    '-': result := bilangan1 - bilangan2;
    '*': result := bilangan1 * bilangan2;
    '/': result := bilangan1 / bilangan2;
  else
    writeIn('Invalid operator');
  end;
  writeIn('mau ngulang?');readln(jawab);
  if (jawab = 'Y') or (jawab = 'y') then goto 10;
end.
```

```
program OPERATOR;
uses WinCrt;
var invalid_operator : boolean;
operator char;
bilangan1, bilangan2, result : real;
begin
  invalid_operator := FALSE;
  writeIn('Masukan dua buah bilangan');
  write(' bilangan 1 ');readln(bilangan1);
  write(' operator ');readln(operator);
  write(' bilangan 2 ');readln(bilangan2);
  case operator of
    '+': result := bilangan1 + bilangan2;
    '-': result := bilangan1 - bilangan2;
    '*': result := bilangan1 * bilangan2;
    '/': result := bilangan1 / bilangan2;
  else
    writeIn('Invalid operator');
  end;
  writeIn('bilangan1 + 2. ': result 5.2);
  bilangan2 + 2. ': is. ': result 5.2);
end
```

10

PROSEDUR DAN FUNGSI

Sebuah program yang baik adalah bagian-bagian kecil dimana setiap permasalahan utama menjadi bagian subprogram, yang disebut dengan bagian kecil dilangani oleh sebuah subprogram, yang disebut dengan modular programming (pemrograman terbagi/terpecah). Cara ini termasuk pemrograman terstruktur dan sangat didukung oleh bahasa Pascal. Untuk itu, Pascal telah menyediakan dua jenis subprogram, yaitu procedure dan function (prosedur dan fungsi).

Dengan modular programming, program lebih mudah dibaca dan dimengerti. Selain itu, pembenahan program dan penelusuran jalannya program (debugging) menjadi lebih mudah sebab dapat langsung diketahui subprogram mana yang berjalan tidak sesuai dengan yang diharapkan.

Procedure (prosedur) adalah suatu program yang terpisah dalam blok sendiri yang berfungsi sebagai program bagian. Bentuk umum dari procedure adalah:

12

PROSEDUR

Procedure (prosedur) adalah suatu program yang terpisah dalam blok sendiri yang berfungsi sebagai program bagian. Bentuk umum dari procedure adalah :

Eight: Statemen CASE

How to Think

DASAR PEMROGRAMAN
by: Fakhur Razi, M.Si

2

ARRAY

- Larik atau *array* adalah tipe terstruktur yang terdiri dari sejumlah komponen yang mempunyai tipe yang sama. Banyaknya suatu komponen tersebut ditunjukkan oleh *indeks*.
- Array juga bisa diartikan sebagai sejumlah variabel yang terdapat dalam satu tipe atau variabel, maka dari itu ada indeks

ARRAY 1D

- Secara umum ada 3 bentuk array, yaitu sebagai berikut :
 - identifier : array[tipe_indeks] of tipe; { var }
 - identifier : array[tipe_indeks] of tipe; { const }
 - identifier = array[tipe_indeks] of tipe; { type }
- Contoh sederhana pada bagian var :
var x:array[1..5] of integer;

3

ARRAY 1D

- Seperti halnya prosedur dan fungsi, tipe data yang boleh langsung dideklarasikan hanyalah tipe ordinal dan real. Untuk string harus dideklarasikan pada bagian type dahulu

```
Ex 1: type huruf = string[30];  
var nama : array[1..10] of huruf;  
Ex 2: type batas = 17..90;  
var usia : array[1..1000] of batas;  
Ex 3: var usia : array[1..1000] of 17..90;  
Ex 4: Type hari=(Sun,Mon,Tue,Wed,Thu,Fri,Sat);  
var masuk: array[1..30] of hari;  
Ex 5: Type hari=(Sun,Mon,Tue,Wed,Thu,Fri,Sat);  
var Masuk : array[mon..sat] of 1..6;
```

4

ARRAY 1D

```
Ex 6 : type batas = 1..5;
        X = array[batas] of char;
        var nilai : x;
Ex 7 : type x = array[1..5] of char;
        var nilai : x;
Ex 8 : const min = 1, Max = 5;
        type x = array[min..max] of char;
        var nilai : x;
Ex 9 : var x : array[#..#] of byte;
```

5

Contoh

```
List Program :
program lipat;
uses winctrl;
var x:array[1..5] of byte;
    y:byte;
begin
  for y:=1 to 5 do
  begin
    x[y]:=y*x[y-1];
    writeln(x[y]);
  end;End.
```

Output dari program
disamping :

```
1 => x[1]
3 => x[2]
6 => x[3]
10 => x[4]
15 => x[5]
```

Indeks array x diisi oleh nilai variabel y dari statement for.

6

Contoh

```
List Program :
program larik1;
uses winctrl;
const x:array[1..5] of byte = (7,32,9,2,56);
var y:byte;
begin
  for y:=1 to 5 do
  begin
    writeln('Konstanta larik ke ',y,' = ',x[y]);
  end;
end.
```

7

Contoh

```
Program larik2:
uses winctrl;
type x=array[0..4] of string[10];
const bahasa:x=('Pascal','C++','Perl','Java','C');
var i: byte;
begin
  for i:=1 to 5 do
  begin
    writeln('Bahasa ke ',i,' = ',bahasa[i-1]);
  end;
end.
```

8

Array 2D

- Array 2D mewakili suatu bentuk tabel atau matrik, yaitu indeks pertama menunjukkan baris dan indeks kedua dapat menunjukkan kolom dari tabel atau matrik

- Bentuk deklarasi

Nama array = array(tipe-indeks1, tipe-indeks2) of tipe-larik

Contoh

Type	Var
x = array[1..3, 1..2] of byte;	Tabel: array[1..3] of array[1..2] of byte;
Tabel : x	

9

Contoh

```

program multi; uses winctrl;
var x:array[1..3,1..2] of byte;
    i,j:byte;
begin
    x[1,1]:=2;
    x[1,2]:=5;
    x[2,1]:=7;
    x[2,2]:=12;
    x[3,1]:=21;
    x[3,2]:=19;
    for i:=1 to 3 do
        begin
            for j:=1 to 2 do
                write(x[i,j]);writeln;
            end;
        end;
end.
    
```

Bila dirunning :

```

2 5
7 12
21 19
    
```

Array multidimensi ini banyak digunakan untuk operasi matriks. Karena, operasi matriks mempunyai elemen banyak

10

Contoh

```

Program penjumlahan-M;
uses winctrl;
type
    angka=array [1..25,1..25] of integer;
var
    i,j,k,B : integer;
    matrik1,matrik2,jumlah : angka;
procedure baca_matriks (var matrik :angka);
begin
    for i:=1 to B do
        begin
            for j:=1 to K do
                begin
                    write (' matrik baris ',j,' kolom ',j,' : ');
                    readln (matrik[i,j]);
                end;
            end;
        end;
end; end;
    
```

11

Contoh

```

procedure tulis_matriks (var matrik : angka);
begin
    writeln;
    for i:=1 to B do
        begin
            writeln;
            for j:=1 to K do
                begin
                    write (matrik[i,j]:6);
                end;
            end;
        end;
end; end;
function jumlah_matriks (matrik1, matrik2 : angka; var jumlah :
angka):integer;
begin
    for i:=1 to B do
        begin
            for j:=1 to K do
                begin
                    jumlah[i,j]:=matrik1[i,j]+matrik2[i,j];
                end;
            end;
        end;
end;end;
    
```

12

Contoh

```
BEGIN
  clrscr;
  write (' banyak kolom matrik : '); readln (K);
  write (' banyak baris matrik : '); readln (B);
  writeln (' masukkan nilai dari matriks III ');
  writeln (' nilai matriks pertama ');
  baca_matriks(matrik1);
  writeln (' nilai matriks kedua ');
  baca_matriks(matrik2);
  write (' matriks pertama adalah ');
  tulis_matriks (matrik1);
  write (' matriks kedua adalah ');
  tulis_matriks (matrik2);
  writeln (' jumlah matriks pertama dan kedua ');
  jumlah_matriks(matrik1,matrik2,jumlah);
  writeln (' jumlah_matriks (jumlah);
  readln;
END.
```

13

Contoh

```

          banyak kolom matrik : 2
          banyak baris matrik : 2
          matriks pertama adalah :
          2 2
          2 2
          masukkan nilai dari matriks III
          matriks kedua adalah :
          2 2
          2 2
          matrik baris 1 kolom 1 : 2
          matrik baris 1 kolom 2 : 2
          matrik baris 2 kolom 1 : 2
          matrik baris 2 kolom 2 : 2
          jumlah matriks pertama dan kedua :
          nilai matriks kedua
          matrik baris 1 kolom 1 : 2
          matrik baris 1 kolom 2 : 2
          matrik baris 2 kolom 1 : 2
          matrik baris 2 kolom 2 : 2
```

14

Contoh

```
Program perkalian-M;
uses winctrl;
type
  larik=array [1..25,1..25] of real;
var
  i,j,K,M,N,L : byte;
  matrik1,Matrik2,matrik : larik;
begin
  clrscr;
  write (' baris matrik pertama = '); readln (M);
  write (' baris m1/ kol m2 = '); readln (N);
  write (' baris matrik pertama = '); readln (L);
  write (' baris matrik pertama ');
  for i:=1 to M do
    begin
      for j:=1 to N do
        begin
          write (' baris ',j,' kolom ',j,' ');readln (matrik1[i,j]);
        end;
      end;
    end;
```

15

MILIK PERPUSTAKAAN
UNIV. NEGERI PADANG

Contoh

```
write (' baris matrik kedua ');
for i:=1 to N do
  begin
    for j:=1 to L do
      begin
        write (' baris ',j,' kolom ',j,' ');readln (matrik2[i,j]);
      end;
    end;
  }perkaliannya
  for i:=1 to M do
    begin
      for j:=1 to N do
        begin
          matrik[i,j]:=0;
          for k:=1 to L do
            matrik[i,j]:=matrik1[i,k]*matrik2[k,j];
          end;
        end;
      end;
```

16

Contoh

```
Clrscr;
(perkaliannya)
WriteIn ('hasil perkalian matrik');
For i:=1 to M do
Begin
  For j:=1 to L do
    write(matrik[i,j]:9:2);
  WriteIn;
End;end.
baris matrik pertama = 2 hasil perkalian matrik=
baris m1/ kol m2 = 2 27.00 27.00 27.00
baris matrik pertama = 2 27.00 27.00 27.00
baris matrik pertama
baris 1 kolom 1 : 2
baris 1 kolom 2 : 2
baris 1 kolom 1 : 2
baris 2 kolom 2 : 2
baris matrik kedua
baris 1 kolom 1 : 2
baris 1 kolom 2 : 2
baris 2 kolom 1 : 2
baris 2 kolom 2 : 2_
```

17

Record

- RECORD adalah suatu tipe data terstruktur dalam pascal.
- Dimana dengan tipe data record dapat dikumpulkan beberapa item data yang masing-masing dapat mempunyai tipe data yang berbeda-beda
- Masing-masing item data disebut dengan FIELD

Jadi record terdiri dari kumpulan field yang dapat berbeda tipe

```
Contoh
var
  Langanan: record
    kode:integer;
    nama:string[20];
    alamat:string[20];
    utangnya:real;
end;
```

18

Record

- Pengaksesan Elemen Record
Cara mengacu pada tiap field pada record pada contoh di atas adalah sebagai berikut:

```
langanan.nama = 'si keren';
```

Pengenal field

Pengenal record

19

Record

```
program recorddata.khs;
uses winctrl;
var
  mahasiswa :Record
    nama : string[25];
    alamat : string[100];
    nim : longint;
end;
BEGIN
  clrscr;
  write (' nama mahasiswa : '); readln (mahasiswa.nama);
  write (' nim mahasiswa : '); readln (mahasiswa.nim);
  write (' alamat mahasiswa : '); readln (mahasiswa.alamat);
  writeIn;
  writeIn (' data mahasiswa ');
  writeIn (' nama : ', mahasiswa.nama);
  writeIn (' nim : ', mahasiswa.nim);
  writeIn (' alamat : ', mahasiswa.alamat);
  readln;
END.
```

20

Record

```
nama mahasiswa : Fakhru Razi
nim mahasiswa : 17964
alamat mahasiswa : pangeran beach hotel

data mahasiswa
nama : Fakhru Razi
nim : 17964
alamat : pangeran beach hotel
```

21

Pengunaan With ... do

□ Penggunaan pernyataan with...do untuk lebih mudah dan lebih pendek pengaksesan field-field pada record

```
program recordstakhs.
uses wincrt.
var
  mahasiswa : Record
  name : string[25];
  alamat : string[100];
  nim : longint;
end.
BEGIN
  With mahasiswa do
  Begin
    clrscr;
    write ( nama mahasiswa ); readln (nama);
    write ( nim mahasiswa ); readln (nim);
    write ( alamat mahasiswa ); readln (alamat);
    writeln ( data mahasiswa );
    writeln ( nama );
    writeln ( nim );
    writeln ( alamat );
    readln;
  End;End.
```

22

Array dari Record

```
program records; uses wincrt;
var
  mahasiswa : array[1..10] of record
  name : string[25];
  nim : string[100];
  alamat : string;
end;
i : integer;
begin
  clrscr;
  writeln ( 'array of mahasiswa' );
  for i := 1 to 10 do
  begin
    with mahasiswa[i] do
    begin
      write ( nama mahasiswa ); readln (nama);
      write ( nim mahasiswa ); readln (nim);
      write ( alamat mahasiswa ); readln (alamat);
      writeln;
    end;
  end;
  writeln ( 'array of mahasiswa' );
  for i := 1 to 10 do
  begin
    with mahasiswa[i] do
    begin
      write ( nama );
      write ( nim );
      write ( alamat );
      writeln;
    end;
  end;
  writeln ( 'array of mahasiswa' );
  for i := 1 to 10 do
  begin
    with mahasiswa[i] do
    begin
      write ( nama );
      write ( nim );
      write ( alamat );
      writeln;
    end;
  end;
  readln;
end.
```

23

Array dari Record

```
jumlah mahasiswa : ?
nama mahasiswa : Fakhru Razi
nim mahasiswa : 17964
tahun masuk up : 1997
alamat mahasiswa : Jl. Gajah 3 (persada)

nama mahasiswa : Mr. Brit
nim mahasiswa : 12345
tahun masuk up : 1998
alamat mahasiswa : jalan belukar

no nama nim kelas alamat
1 Fakhru Razi 17964 1997 Jl. Gajah 3 (persada)
2 Mr. Brit 12345 1998 jalan belukar
```

24

FILE

- Komposisi dari File Teks
File Teks disusun sebagai runtunan beberapa baris
 - Tiap baris terdiri dari runtunan karakter.
 - Tiap baris diakhiri oleh karakter khusus, yaitu END-OF-LINE(EOLN)
 - Karakter yang terakhir dari file teks adalah END-OF-FILE (EOF)
- Sintak untuk mendeklarasikan file yang berisikan bilangan bulat adalah
Contoh.
Type
TFile = File of Integer;
var
vfile : Tfile;

25

FILE

- Contoh untuk menulis data yang disimpan dalam bentuk matrik dapat dibuat sebagai berikut:
Procedure Tulis;
begin
Assign(Vfile,nfile);
rewrite(Vfile);
for k:=1 to bar_A do
begin
for i:=1 to kol_B do
write(Vfile,c[k,i]);
end;
close(Vfile);
end;

26

FILE

- Contoh untuk membaca data yang tersimpan dalam bentuk matrik dapat dibuat sebagai :
Procedure Baca;
begin
Assign(Vfile,nfile);
reset(Vfile);
for k:=1 to bar_A do
begin
for i:=1 to kol_B do
read(Vfile,c[k,i]);
end;
close(Vfile);
end;

27

FILE

- Berikut ini adalah contoh untuk menyimpan data record mahasiswa yang bentuk array dengan menggunakan perintah WITH dapat dilakukan sebagai berikut:

```
Procedure TulisRecordWith;  
Begin  
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(nia(i,j));  
na := na+nial(i);  
end;  
na := na/k;  
end;  
write(Vfile,dtm(i));  
end;  
Close(vfile);  
End;
```

28

FILE

□ Kemudian contoh untuk membaca data record mahasiswa yang bentuk array dengan menggunakan perintah WITH dapat dilakukan sebagai berikut

Procedure BacaRecordWith;

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);

end;

29

FILE

□ EOF :menerima argumen nama file dan menghasilkan nilai true jika sudah tidak ada data yang bisa dibaca lagi.

□ EOLN :menerima argumen nama file dan menghasilkan nilai true jika sudah tidak ada lagi data yang bisa lagi dalam satu baris.

30

FILE

□ Operasi pada file teks

Untuk melakukan operasi pada file teks, kita perlu mendeklarasikan suatu variabel dengan tipe teks seperti berikut:

var

F : text;

F adalah sembarang variabel file teks dan readme adalah nama file teks yang akan dibaca, pertama kita harus memanggilnya dengan fungsi :

assign(F,'README');

31

FILE

□ Membaca File dan Menampilkannya di Layar
Sebelum kita bisa membaca kita harus membuka file tersebut.

 reset(F);

kita bisa membaca file baris demi baris, misalkan dengan menyatakan suatu variabel s sebagai string:

 readln(F, s);

setelah selesai membaca keseluruhan teks kita harus menutupnya dengan:

 close(F);

32

CONTOH

Menulis File Baru
Kita membuat suatu file teks dengan mengubah kata kunci :

reset(F); menjadi rewrite(F);

Kemudian: Gunakan readln(F,s) untuk membaca file yang akan dikopi dan writeIn(F,s) untuk menulis ke file tujuan

Contoh 1..

(misal kita punya suatu file dengan path 'D:\file1.txt') dan kita akan mengkopinya ke file dengan path 'D:\file1.txt')

34

CONTOH

Contoh, (misal kita punya suatu file dengan path 'D:\file1.txt')

Program baca_file_teks:

Uses crt;

var

F : text;

nama_file:string;

s : string;

begin

clrscr;

write('NAMA FILE YANG DIBACA: ');

readln(nama_file); (menuliskan nama file 'D:\file1.txt')

assign(F,nama_file); (memanggil file)

reset(F); (membuka file)

while not EOF(F) do (menampilkan seluruh isi file)

begin

readln(F,s);

writeIn(s);

end;

close(F); (menutup file)

end.

33

CONTOH

```
Program Menulis ke File
Uses crt;
var
  nama_file : record
  string(80)
  nama : string(80)
  nama_file : string(80)
  begin
  end;
  f : file of record_nama;
  readln(nama);
  readln(nama_file);
  assign(f,nama_file);
  reset(f);
  while not EOF(f) do
  begin
  readln(f,s);
  writeIn(f,s);
  end;
  close(f);
end.
```

36

CONTOH

Program baca_tulisfile_teks:

Uses whicrt;

var

F,F1 : text; nama_file:string; s : string;

begin

clrscr;

write('NAMA FILE YANG DIBACA: ');

readln(nama_file); (menuliskan nama file 'D:\file1.txt')

assign(F,nama_file); (memanggil file)

reset(F); (membuka file)

assign(F1,d:\file2.txt');

rewrite(F1);

while not EOF(F) do

begin

readln(F,s); (membaca seluruh isi file)

writeIn(F1,s); (mengopi isi file1.txt ke file2.txt)

end;

close(F); (menutup file file1.txt)

close(F1); (menutup file file2.txt)

end.

35

