

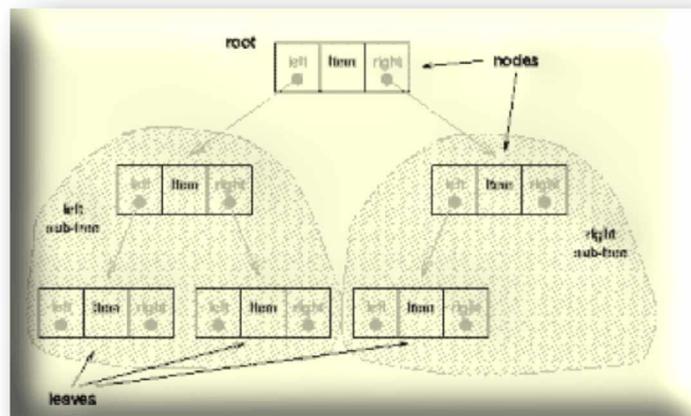
KODE MODUL

DTA.DEV.101.(3).A



**SEKOLAH MENENGAH KEJURUAN
BIDANG KEAHLIAN TEKNOLOGI INFORMASI DAN KOMUNIKASI
PROGRAM KEAHLIAN REKAYASA PERANGKAT LUNAK**

MELAKUKAN PERANCANGAN PENGUMPULAN DATA



**BAGIAN PROYEK PENGEMBANGAN KURIKULUM
DIREKTORAT PENDIDIKAN MENENGAH KEJURUAN
DIREKTORAT JENDERAL PENDIDIKAN DASAR DAN MENENGAH
DEPARTEMEN PENDIDIKAN NASIONAL
2004**

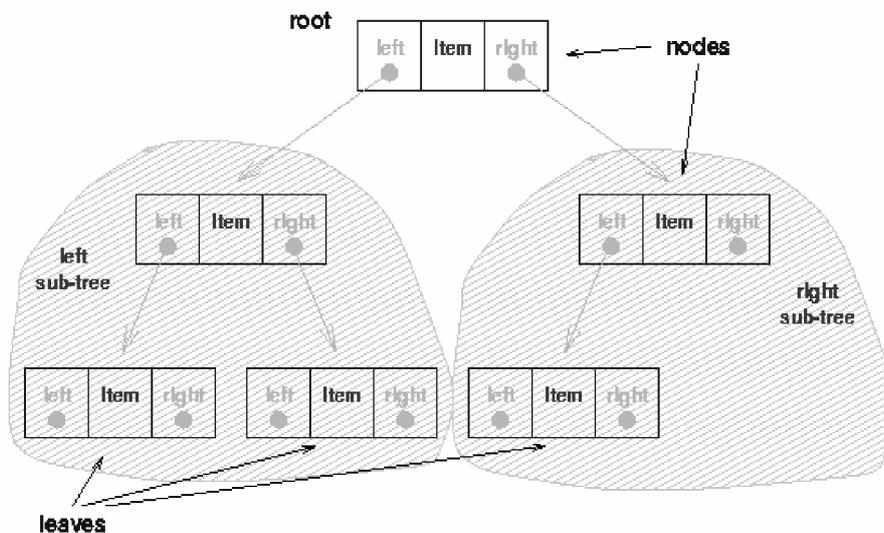
KODE MODUL

DTA.DEV.101.(3).A



SEKOLAH MENENGAH KEJURUAN
BIDANG KEAHLIAN TEKNOLOGI INFORMASI DAN KOMUNIKASI
PROGRAM KEAHLIAN REKAYASA PERANGKAT LUNAK

Melakukan Perancangan Pengumpulan Data



BAGIAN PROYEK PENGEMBANGAN KURIKULUM
DIREKTORAT PENDIDIKAN MENENGAH KEJURUAN
DIREKTORAT JENDERAL PENDIDIKAN DASAR DAN MENENGAH
DEPARTEMEN PENDIDIKAN NASIONAL
2004



SEKOLAH MENENGAH KEJURUAN
BIDANG KEAHLIAN TEKNOLOGI INFORMASI DAN KOMUNIKASI
PROGRAM KEAHLIAN REKAYASA PERANGKAT LUNAK

KODE MODUL

DTA.DEV.101.(3).A*

Melakukan Perancangan Pengumpulan Data

PENYUSUN
TIM FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA

BAGIAN PROYEK PENGEMBANGAN KURIKULUM
DIREKTORAT PENDIDIKAN MENENGAH KEJURUAN
DIREKTORAT JENDERAL PENDIDIKAN DASAR DAN MENENGAH
DEPARTEMEN PENDIDIKAN NASIONAL
2004

KATA PENGANTAR

Modul dengan judul “Melakukan Perancangan Pengumpulan Data” merupakan bahan ajar yang digunakan sebagai panduan praktikum peserta diklat Sekolah Menengah Kejuruan (SMK) untuk membentuk salah satu bagian dari kompetensi pada Bidang Keahlian Teknik Telekomunikasi.

Modul ini membahas tentang perancangan dalam pengumpulan data untuk membuat suatu algoritma yang akan diaplikasikan pada program komputer dengan bahasa pemrograman tertentu. Modul ini terdiri dari 3 (tiga) kegiatan belajar. Kegiatan belajar 1 berisi tentang Pemahaman prosedur pengumpulan data/ pemetaan data. Kegiatan belajar 2 berisi tentang Persiapan pokok-pokok kebutuhan data. Kegiatan belajar 3 berisi tentang Perancangan pengumpulan data.

Diharapkan dengan menguasai modul ini peserta diklat dapat melakukan perancangan pengumpulan data sesuai dengan SOP yang berlaku.

Yogyakarta, November 2004

Penyusun

Tim Fakultas Teknik

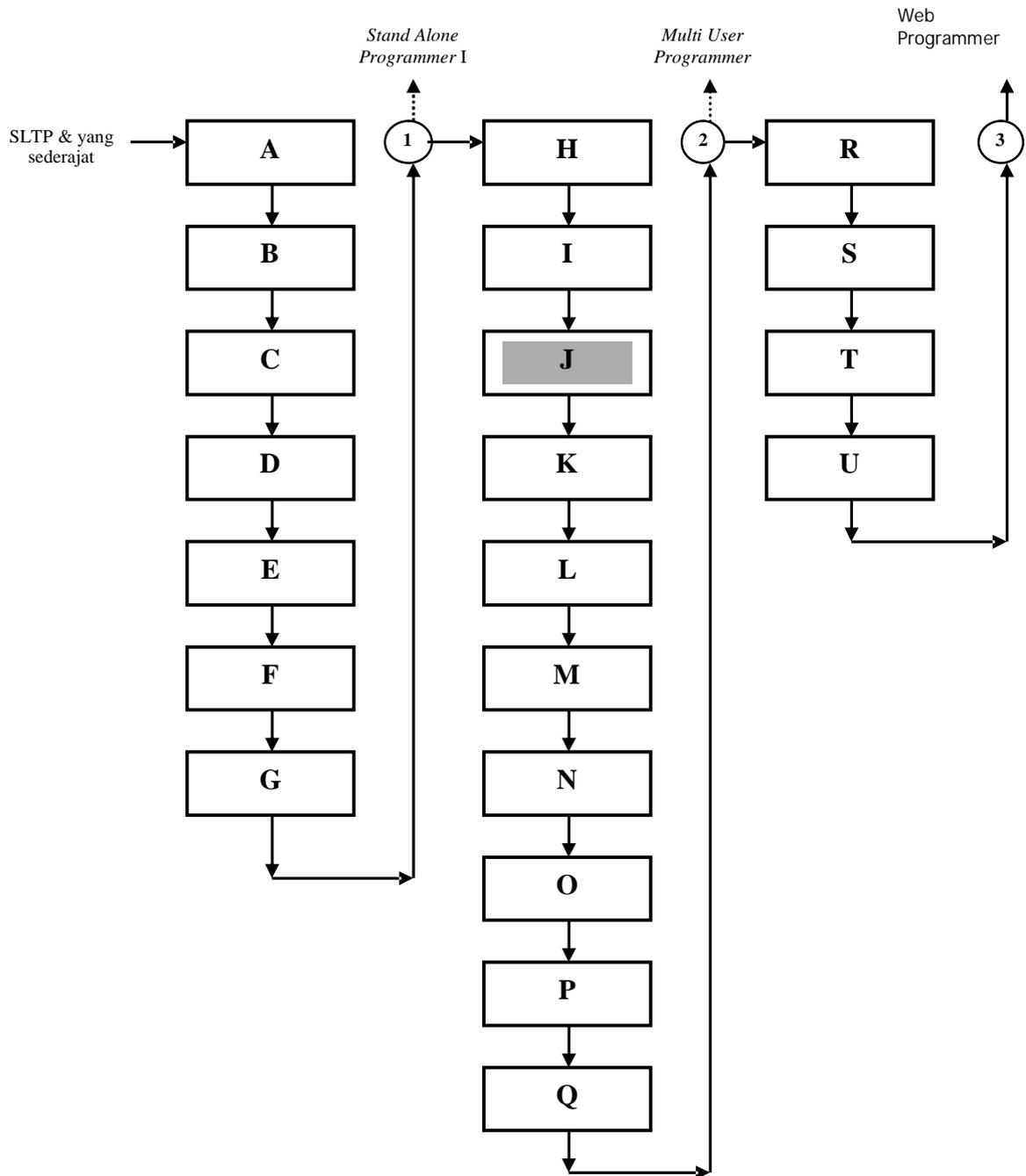
Universitas Negeri Yogyakarta

DAFTAR ISI MODUL

	Halaman
HALAMAN DEPAN	i
HALAMAN DALAM.....	ii
KATA PENGANTAR	iii
DAFTAR ISI.....	iv
PETA KEDUDUKAN MODUL	vi
PERISTILAHAN/ GLOSSARY	viii
I. PENDAHULUAN	1
A. DESKRIPSI JUDUL	1
B. PRASARAT	1
C. PETUNJUK PENGGUNAAN MODUL	1
1. Petunjuk bagi Peserta Diklat	1
2. Peran Guru	2
D. TUJUAN AKHIR	3
E. KOMPETENSI	3
F. CEK KEMAMPUAN	4
II. PEMELAJARAN	6
A. RENCANA PEMELAJARAN PESERTA DIKLAT.....	6
B. KEGIATAN BELAJAR	8
1. Kegiatan Belajar 1: Pemahaman prosedur pengumpulan data/ pemetaan data.....	8
a. Tujuan Kegiatan Pemelajaran	8
b. Uraian Materi 1.....	8
c. Rangkuman 1	21
d. Tugas 1	21
e. Tes Formatif 1.....	22

f. Kunci Jawaban Formatif 1	22
g. Lembar Kerja 1.....	25
2. .Kegiatan Belajar 2: Mempersiapkan pokok-pokok kebutuhan data.....	27
a. Tujuan Kegiatan Pemelajaran	27
b. Uraian Materi 2	27
c. Rangkuman 2	33
d. Tugas 2	33
e. Tes Formatif 2	33
f. Kunci Jawaban Formatif 2	34
g. Lembar Kerja 2.....	39
3. Kegiatan Belajar 3:Melakukan Perancangan Pengumpulan Data.....	40
a. Tujuan Kegiatan Pemelajaran.....	40
b. Uraian Materi 3.....	40
c. Rangkuman 3	45
d. Tugas 3	45
e. Tes Formatif 3	45
f. Kunci Jawaban Formatif 3.....	48
g. Lembar Kerja 3.....	52
III. EVALUASI	53
A. PERTANYAAN	53
B. KUNCI JAWABAN	53
C. KRITERIA KELULUSAN	58
IV. PENUTUP	59
DAFTAR PUSTAKA	60

PETA KEDUDUKAN MODUL



Keterangan :

Kode	Kode Kompetensi	Kompetensi
A	SWR.OPR.200.(1).A	Mengoperasikan Sistem Operasi
B	SWR.OPR.100.(1).A	Menginstalasi software
C	SWR.MNT.100.(1).A	Mengubah konfigurasi software
D	SWR.DEV.100.(1).A	Menyiapkan dan melakukan survey untuk menentukan kebutuhan data
E	SWR.OPR.309.(1).A	Mengoperasikan software bahasa pemrograman level 1
F	DTA.OPR.115.(1).A	Konversi data level 1
G	SWR.DEV.500.(1).A	Menguji program level 1
H	HDW.OPR.103.(1).A	Mengoperasikan sistem operasi jaringan komputer berbasis teks
I	HDW.OPR.104.(1).A	Mengoperasikan sistem operasi jaringan komputer berbasis GUI
J	DTA.DEV.101.(3).A	Melakukan perancangan pengumpulan data
K	SWR.DEV.300.(2).A	Melakukan desain dan perancangan software
L	SWR.DEV.400.(2).A	Melakukan pengkodean program
M	DTA.MNT.101.(2).A	Melakukan back up data
N	DTA.MNT.102.(2).A	Melakukan restore data
O	SWR.OPR.303.(2).A	Mengoperasikan software aplikasi basis data
P	DTA.OPR.119.(2).A	Membuat query data
Q	SWR.DEV.500.(2).A	Menguji program
R	SWR.DEV.401.(2).A	Membangun interface dengan bahasa pemrograman berorientasi objek
S	SWR.DEV.402.(3).A	Melakukan pengkodean program
T	SWR.OPR.304.(3).A	Mengoperasikan bahasa pemrograman berbasis web
U	SWR.DEV.403.(2).A	Membangun program aplikasi remote data interaktif

PERISTILAHAN/ GLOSSARY

- Algoritma : Rancangan penyelesaian masalah dengan prosedur dan terstruktur. Rangkaian perintah dan langkah tatacara yang teratur.
- Program : Susunan prosedural yang akan dieksekusi. Didalam dunia komputer, prosedural ini adalah merupakan urutan perintah yang sesuai dengan bahasa pemrograman yang digunakan.
- Pemrograman : Proses transformasi suatu deskripsi sebuah sistem ke dalam perintah-perintah yang dapat ditafsirkan oleh komputer .
- Programmer : Orang yang membuat program yang akan dijalankan oleh komputer.
- Compiler : Perangkat lunak yang melakukan proses penterjemahan kode (dibuat programmer) kedalam bahasa mesin.
- Case Sensitive : Suatu metode yang membedakan penulisan dengan huruf besar dan huruf kecil, sehingga kata "Komputer" berbeda dengan komputer dan KOMPUTER.
- Flowchart : Bagan alir, aliran dari prosedur kerja adalah bagan (chart) yang menunjukkan alir (flow) didalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.

BAB I

PENDAHULUAN

A. DESKRIPSI JUDUL

Melakukan perancangan pengumpulan data merupakan modul berisi materi dasar tentang pemahaman prosedur pengumpulan data. Modul ini terdiri dari 3 (tiga) kegiatan belajar. Kegiatan belajar 1 berisi tentang memahami prosedur pengumpulan data/pemetaan data. Kegiatan belajar 2 berisi tentang mempersiapkan pokok-pokok kebutuhan data. Kegiatan belajar 3 berisi tentang melakukan perancangan pengumpulan data.

Dengan menguasai modul ini peserta diklat mampu menguasai perancangan pengumpulan data.

B. PRASYARAT

Modul perancangan pengumpulan data merupakan modul lanjutan pembelajaran maka membutuhkan persyaratan modul yang lain atau kemampuan lain yang harus dimiliki peserta diklat sebelum mempelajari modul ini yaitu modul mengoperasikan PC stand alone dengan sistem operasi berbasis GUI dan mengoperasikan periferal.

C. PETUNJUK PENGGUNAAN MODUL

1. Petunjuk bagi Peserta Diklat

Peserta diklat diharapkan dapat berperan aktif dan berinteraksi dengan sumber belajar yang dapat digunakan, karena itu harus memperhatikan hal-hal sebagai berikut :

a. Langkah-langkah belajar yang ditempuh

- 1) Persiapkan alat bantu berupa komputer!
- 2) Bacalah dengan seksama uraian materi pada setiap kegiatan belajar!
- 3) Cermatilah langkah-langkah kerja pada setiap kegiatan belajar sebelum mengerjakan, bila belum jelas tanyakan pada instruktur!
- 4) Jangan menghidupkan komputer sebelum disetujui oleh instruktur!
- 5) Rapikan komputer yang telah digunakan!

b. Perlengkapan yang Harus Dipersiapkan

Guna menunjang keselamatan dan kelancaran tugas/ pekerjaan yang harus dilakukan, maka persiapkanlah seluruh perlengkapan yang diperlukan. Beberapa perlengkapan yang harus dipersiapkan adalah:
Alat tulis

c. Hasil Pelatihan

Peserta diklat mampu membuat program dalam bahasa Pascal maupun bahasa C dengan merancang algoritma terlebih dahulu dan jika diperlukan menggunakan flowchart untuk membantu dalam pembuatan algoritma yang kompleks.

2. Peran Guru

Guru yang akan mengajarkan modul ini hendaknya mempersiapkan diri sebaik-baiknya yaitu mencakup aspek strategi pembelajaran, penguasaan materi, pemilihan metode, alat bantu media pembelajaran dan perangkat evaluasi.

Guru harus menyiapkan rancangan strategi pembelajaran yang mampu mewujudkan peserta diklat terlibat aktif dalam proses pencapaian/ penguasaan kompetensi yang telah diprogramkan. Penyusunan

rancangan strategi pembelajaran mengacu pada kriteria unjuk kerja (KUK) pada setiap subkompetensi yang ada dalam GBPP.

D. TUJUAN AKHIR

Peserta diklat dapat melakukan perancangan pengumpulan data.

E. KOMPETENSI

Sub Kompetensi	Kriteria Unjuk Kerja	Lingkup Belajar	Materi Pokok Pembelajaran		
			Sikap	Pengetahuan	Ketrampilan
1	2	3	4	5	6
1. Memahami prosedur pengumpulan data/ pemetaan data	<ul style="list-style-type: none"> § Gambaran struktur data dijelaskan sesuai dengan SOP § Prinsip dan fungsi tingkatan dalam struktur data § Prinsip dan fungsi struktur data dalam definisi formal 	<ul style="list-style-type: none"> § Konsep algoritma pemrograman § Diagram alir data 	Menerangkan konsep algoritma pemrograman	<ul style="list-style-type: none"> • Menunjukkan fungsi algoritma pemrograman 	<ul style="list-style-type: none"> • Menggunakan fungsi dan operator algoritma pemrograman
2. Mempersiapkan pokok-pokok kebutuhan data	<ul style="list-style-type: none"> § Tahapan perencanaan kebutuhan data ditentukan sesuai SOP § Prinsip dan fungsi tahapan perancangan struktur data ditentukan sesuai SOP 	<ul style="list-style-type: none"> • Algoritma pemrograman 	<ul style="list-style-type: none"> • Menentukan data yang akan digunakan sesuai dengan SOP 	<ul style="list-style-type: none"> • Menguraikan fungsi bermacam jenis data 	<ul style="list-style-type: none"> • Mengidentifikasi kebutuhan data
3. Melakukan perancangan pengumpulan data	<ul style="list-style-type: none"> • Rancangan pengumpulan data menggunakan prinsip dan fungsi stack • Rancangan pengumpulan data menggunakan prinsip dan fungsi polindrom • Rancangan pengumpulan 	<ul style="list-style-type: none"> § Metode pada algoritma pemrograman • Struktur data 	<ul style="list-style-type: none"> • Teliti dan cermat • Mengikuti prosedur pengumpulan data sesuai SOP 	<ul style="list-style-type: none"> • Menyusun hasil pengumpulan data menjadi suatu acuan pembuatan program 	<ul style="list-style-type: none"> • Menyelesaikan desain rancangan pengumpulan data

	data menggunakan prinsip dan fungsi antrian (queue) <ul style="list-style-type: none"> • Rancangan pengumpulan data menggunakan prinsip dan fungsi tree/pohon • Rancangan pengumpulan data menggunakan prinsip dan fungsi graph 				
--	---	--	--	--	--

F. CEK KEMAMPUAN

Untuk mengetahui kemampuan awal yang telah dimiliki, maka isilah cek list (√) seperti pada tabel di bawah ini dengan sikap jujur dan dapat dipertanggung jawabkan.

Sub Kompetensi	Pernyataan	Saya dapat Melakukan Pekerjaan ini dengan Kompeten		Bila Jawaban "Ya" Kerjakan
		Ya	Tidak	
Menyiapkan pengoperasian PC dengan program pascal dan C	1. Sebutkan bahasa pemrograman prosedural			Tes Formatif 1
	2. Memahami sintaks program dengan menggunakan bahasa pascal dan bahasa C			Tes Formatif 2
	3. Mampu membuat algoritma mencari jumlah 3 buah bilangan bulat dengan flowchart			Tes Formatif 3
	4. Mampu membuat algoritma mencari hasil kali dari 2 buah bilangan dengan flowchart			Tes Formatif 4
Mempersiapkan pokok-pokok kebutuhan data	1. Mengetahui tipe data dasar dan range nilainya			Tes Formatif 1
	2. Mampu membuat algoritma dan flowchartnya untuk menentukan bilangan terbesar antara dua bilangan bulat			Tes Formatif 2
	3. Sebutkan 3 macam notasi struktur pengulangan dan jelaskan			Tes Formatif 3

	Buatlah algoritma dan flowchartnya untuk mencetak bilangan 1 sampai 4 menggunakan perulangan			Tes Formatif 4
Melakukan perancangan pengumpulan data	1. Mengetahui operasi push dan pop pada tumpukan/stack			Tes Formatif 1
	2. Mengetahui antrian/queue			Tes Formatif 2
	3. Mengetahui operasi enqueue dan dequeue pada antrian/queue			Tes Formatif 3
	4. Mengetahui maksud Tree dan istilah : tingkat (level), derajat (degree), daun (leaf), tinggi (height) atau kedalaman (depth) dan ancestor			Tes Formatif 4
	5. Mengetahui Graph			Tes Formatif 5

Apabila anda menjawab TIDAK pada salah satu pernyataan di atas, maka pelajarilah modul ini.

BAB II

PEMELAJARAN

A. RENCANA PEMELAJARAN PESERTA DIKLAT

Kompetensi : Melakukan perancangan pengumpulan data

1. Sub Kompetensi: Memahami prosedur pengumpulan data/ pemetaan data

Jenis Kegiatan	Tanggal	Waktu	Tempat Belajar	Alasan Perubahan	Tanda Tangan Guru
Pengenalan algoritma dan pemrograman					
Menjelaskan mekanisme pelaksanaan algoritma oleh pemroses					
Mengenalkan jenis-jenis bahasa pemrograman dan dasar-dasar algoritma					
Menjelaskan struktur program bahasa pascal dan bahasa C					
Menjelaskan variabel, konstanta, ekspresi matematika dan flowchart					

2. Sub Kompetensi: Mempersiapkan pokok-pokok kebutuhan data

Jenis Kegiatan	Tanggal	Waktu	Tempat Belajar	Alasan Perubahan	Tanda Tangan Guru
Mengenalkan serta menjelaskan tipe data dasar					
Menjelaskan pemilihan/kontrol program dan perulangan					

3. Sub Kompetensi: Melakukan perancangan pengumpulan data

Jenis Kegiatan	Tanggal	Waktu	Tempat Belajar	Alasan Perubahan	Tanda Tangan Guru
Menjelaskan prinsip umum dari fungsi stack, queue, tree dan graph					

B. KEGIATAN BELAJAR

1. Kegiatan Belajar 1: Pemahaman prosedur pengumpulan data/ pemetaan data

b. Tujuan Kegiatan Pemelajaran

Setelah mempelajari kegiatan belajar ini peserta diklat mampu memahami dan menjelaskan prosedur pengumpulan data/ pemetaan data dan mampu menyusunnya dalam sebuah algoritma pemrograman serta menentukan peralatan bantu dalam melakukan pengumpulan data/pemetaan data.

c. Uraian Materi

1) Pengertian Algoritma

Pandangan mengenai komputer sebagai sebuah mesin yang “pintar” adalah pendapat yang salah, karena komputer hanyalah suatu alat yang diberi serangkaian perintah oleh manusia sehingga dapat menyelesaikan permasalahan secara cepat, akurat, bahkan berulang-ulang tanpa kenal lelah dan bosan. Sekumpulan instruksi yang merupakan penyelesaian masalah itu dinamakan program. Agar program dapat dilaksanakan oleh komputer, program tersebut harus ditulis dalam suatu bahasa yang dimengerti oleh komputer. Bahasa komputer yang digunakan dalam menulis program dinamakan bahasa pemrograman. Urutan langkah-langkah yang sistematis untuk menyelesaikan sebuah masalah dinamakan algoritma. Jadi algoritma adalah urutan logis pengambilan keputusan untuk pemecahan masalah. Kata logis merupakan kata kunci. Langkah-langkah tersebut harus logis, ini berarti nilai kebenarannya harus dapat ditentukan, benar atau salah. Langkah-langkah yang tidak benar dapat memberikan hasil yang salah. Sebagai contoh tinjau

persoalan mempertukarkan isi dua buah bejana, A dan B. Bejana A berisi larutan yang berwarna merah, sedangkan bejana B berisi air berwarna biru. Kita ingin mempertukarkan isi kedua bejana itu sedemikian sehingga bejana A berisi larutan berwarna biru dan bejana B berisi larutan berwarna merah. Ada 2 algoritma untuk menyelesaikan permasalahan tersebut, yaitu :

a) Algoritma yang pertama, ada dua langkah :

- (1) Tuangkan larutan dari bejana A ke dalam bejana B
- (2) Tuangkan larutan dari bejana B ke dalam bejana A

Algoritma tukar isi bejana di atas tidak menghasilkan pertukaran yang benar, karena langkah-langkahnya tidak logis sehingga yang terjadi adalah percampuran keduanya.

b) Algoritma yang kedua, ada 3 langkah :

- (1) Tuangkan larutan dari bejana A ke dalam bejana C
- (2) Tuangkan larutan dari bejana B ke dalam bejana A
- (3) Tuangkan larutan dari bejana C ke dalam bejana B

Sekarang, dengan algoritma tukar isi bejana yang sudah diperbaiki ini, isi bejana A dan B dapat dipertukarkan dengan benar.

Dari kedua contoh algoritma di atas dapat diambil 2 pesan penting. Pertama, algoritma harus benar. Kedua, algoritma harus berhenti, dan setelah berhenti, algoritma memberi hasil yang benar.

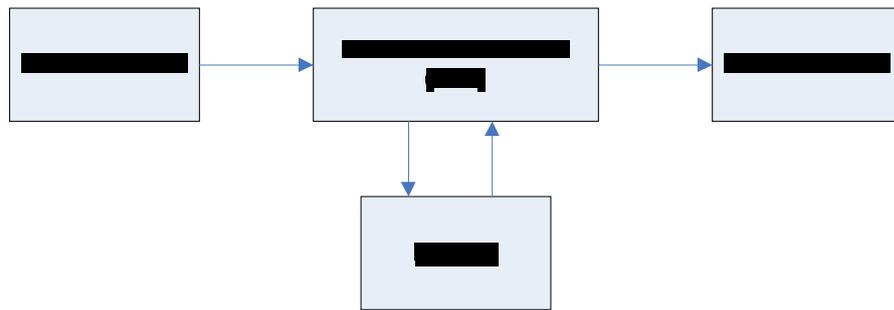
2) Pengertian Pemrograman

Komputer hanyalah salah satu pemroses. Agar dapat dilaksanakan oleh komputer, algoritma harus ditulis dalam notasi bahasa pemrograman sehingga dinamakan program. Jadi program adalah

perwujudan atau implementasi algoritma yang ditulis dalam bahasa pemrograman tertentu sehingga dapat dilaksanakan oleh komputer. Program ditulis dalam salah satu bahasa pemrograman, dan kegiatan membuat program disebut pemrograman (programming). Orang yang menulis program disebut pemrogram (programmer). Tiap-tiap langkah di dalam program disebut pernyataan atau instruksi. Jadi, program tersusun atas sederetan instruksi. Bila suatu instruksi dilaksanakan, maka operasi-operasi yang bersesuaian dengan instruksi tersebut dikerjakan oleh komputer.

3) Mekanisme Pelaksanaan Algoritma oleh Pemroses

Secara garis besar komputer tersusun atas empat komponen utama: piranti masukan, piranti keluaran, unit pemroses utama dan memori. Unit pemroses utama (Central Processing Unit – CPU) adalah “otak” komputer, yang berfungsi mengerjakan operasi-operasi dasar seperti operasi perbandingan, operasi perhitungan, operasi membaca dan operasi menulis. Memori adalah komponen yang berfungsi menyimpan atau mengingat-ingat. Yang disimpan di dalam memori adalah program (berisi operasi-operasi yang akan dikerjakan oleh CPU) dan data atau informasi (sesuatu yang diolah oleh operasi-operasi). Piranti masukan atau keluaran (I/O devices) adalah alat yang memasukkan data atau program ke dalam memori, dan alat yang digunakan komputer untuk mengkomunikasikan hasil-hasil aktivitasnya. Contoh piranti masukan adalah : keyboard, mouse, scanner dan disk. Contoh alat keluaran adalah : monitor, printer, plotter dan disk.



Gambar 1.

Komponen-komponen Utama Komputer

4) Bahasa Pemrograman

Saat ini kita dapat berkomunikasi dengan komputer dengan menggunakan bahasa yang kita mengerti. Hal ini dapat kita lakukan karena para ahli telah berhasil membuat kamus yang disebut dengan bahasa pemrograman yang akan menterjemahkan bahasa yang kita buat menjadi bahasa mesin, kamus ini disebut dengan Compiler. Proses penterjemahan bahasa manusia ke bahasa mesin disebut dengan kompilasi. Adapaun bahasa-bahasa pemrograman tersebut antara lain :

Bahasa Pemrograman	Tipe	Dibuat
FORTTRAN	Prosedural	1950
BASIC	Prosedural	1960
LISP	Fungsional	1950
Prolog	Deklaratif	1970
Ada	Prosedural	1970
SmalTalk	Berorientasi Objek	1970
Pascal	Prosedural	1970
C	Prosedural	1970
C++	Berorientasi Objek	1980

5) Dasar-dasar Algoritma

a) Proses, Instruksi dan Aksi

Pada dasarnya, sebuah algoritma merupakan deskripsi pelaksanaan suatu proses. Algoritma disusun oleh sederetan langkah instruksi

yang logis. Tiap langkah instruksi tersebut melakukan suatu tindakan atau aksi. Bila suatu aksi dilaksanakan, maka sejumlah operasi yang bersesuaian dengan aksi itu dikerjakan oleh pemroses. Efek dari pengerjaan suatu aksi dapat diamati dengan membandingkan keadaan pada saat aksi belum dimulai, t_0 , dan keadaan pada saat aksi selesai dikerjakan, t_1 .

To : Keadaan sebelum aksi dikerjakan

Aksi

T1 : Keadaan setelah aksi dikerjakan

Sebagai contoh, tinjau kembali algoritma yang menggambarkan proses mempertukarkan larutan dari dua buah bejana A dan B. Pada setiap pelaksanaan aksi kita amati keadaan awal dan keadaan akhirnya.

To : bejana A berisi larutan berwarna merah, bejana B berisi larutan berwarna biru (bejana C masih kosong).

(1) Tuangkan larutan dari bejana A ke dalam bejana C

T1 : bejana A kosong, bejana C berisi larutan berwarna Merah

To : bejana A kosong, bejana B berisi larutan berwarna biru.

(2) Tuangkan larutan dari bejana B ke dalam bejana A

T1 : bejana A berisi larutan berwarna biru, bejana B kosong

To : bejana B kosong, bejana C berisi larutan berwarna merah.

(3) Tuangkan larutan dari bejana C ke dalam bejana B

T1 : Bejana B berisi larutan berwarna merah, bejana A sudah berisi larutan berwarna biru (bejana C kosong).

Keadaan awal dan keadaan akhir algoritma dapat dijadikan acuan bagi pemrogram dalam merancang suatu algoritma. Keadaan akhir mencerminkan hasil yang diinginkan dari sebuah keadaan awal. Algoritma berisi langkah-langkah pencapaian keadaan akhir dari keadaan awal yang didefinisikan.

Tahap-tahap penyusunan algoritma seringkali dimulai dari langkah yang global lebih dahulu. Langkah global ini diperhalus sampai ke langkah yang lebih rinci. Pendekatan desain algoritma seperti ini dinamakan perancangan puncak turun (top-down design). Cara pendekatan seperti ini sangat bermanfaat dalam membuat algoritma untuk masalah yang cukup rumit atau kompleks.

b) Struktur Dasar Algoritma

Algoritma berisi langkah-langkah penyelesaian suatu masalah. Langkah-langkah tersebut dapat berupa runtunan aksi, pemilihan aksi dan pengulangan aksi. Ketiga jenis langkah tersebut membentuk konstruksi suatu algoritma. Jadi, sebuah algoritma dapat dibangun dari 3 buah struktur dasar, yaitu :

- (1) Runtunan (sequence)
- (2) Pemilihan (selection)
- (3) Pengulangan (repetition)

c) Runtunan

Sebuah runtunan terdiri dari satu atau lebih instruksi. Tiap instruksi dikerjakan secara berurutan sesuai dengan urutan penulisannya. Urutan instruksi menentukan keadaan akhir dari suatu algoritma. Bila urutannya diubah, maka hasil akhirnya mungkin juga akan berubah.

d) Pemilihan

Ada kalanya sebuah instruksi dikerjakan jika kondisi tertentu dipenuhi. Penulisan pemilihan secara umum :

If kondisi then

Aksi

Dalam bahasa indonesia, if berarti "jika" dan then artinya "maka". Kondisi adalah persyaratan yang dapat bernilai salah atau benar. Aksi hanya dilakukan jika kondisi bernilai benar. Perhatikan kata yang digarisbawahi, if dan then merupakan kata kunci(keywords) untuk struktur pemilihan ini. Dalam kehidupan sehari-hari, kita sering menuliskan pernyataan tindakan bila suatu persyaratan dipenuhi. Misalnya :

If Zaki memperoleh juara kelas then

Ayah akan membelikannya sepeda

If jalan panenan macet then

Ambil alternatif jalan dipati ukur

Struktur pemilihan if-then hanya memberikan satu pilihan aksi jika kondisi dipenuhi (bernilai benar), dan tidak memberi pilihan aksi lain jika bernilai salah. Bentuk pemilihan yang lebih umum ialah memilih satu dari dua buah aksi bergantung pada nilai kondisinya :

If kondisi then

Aksi 1

Else

Aksi 2

Else artinya "kalau tidak". Bila kondisi bernilai benar, aksi 1 akan dikerjakan, tetapi kalau tidak, aksi 2 yang akan dikerjakan. Misalnya pada pernyataan berikut :

If hari hujan then

Pergilah dengan naik beca

Else

Pergilah dengan naik motor

Jika kondisi "hari hujan" bernilai benar, maka aksi "pergilah dengan naik beca" dilakukan, sebaliknya aksi "pergilah dengan naik motor" akan dilakukan jika "hari hujan" tidak benar.

e) Pengulangan

Salah satu kelebihan komputer adalah kemampuannya untuk melakukan melakukan pekerjaan yang sama berulang kali tanpa mengenal lelah. Struktur pengulangan disebut kalang (loop), dan bagian algoritma yang diulang (aksi) dinamakan badan kalang (loop body).

6) Struktur Program Bahasa Pascal dan Bahasa C

a) Algoritmik

Algoritma nama_algoritma

{kepala algoritma berisi penjelasan seperlunya}

Deklarasi {berisi variabel yang terlibat}

Deskripsi {berisi detail algoritma}

b) Bahasa Pascal

Program nama_program;

[deklarasi label]

[deklarasi konstan]

[deklarasi tipe]

[deklarasi variable]

[deklarasi subprogram]

```

Begin
    Pernyataan;
    .....
    Pernyataan
End.

```

c) Bahasa C

```

#include <stdio.h>
[deklarasi subprogram]
Main()
{
    [deklarasi variabel]
    Pernyataan;
    .....
    Return 0;
}

```

7) Variabel

Variabel (perubah) merupakan suatu nama yang menyiratkan lokasi memori komputer yang dapat digunakan untuk menyimpan nilai, dimana isinya dapat diubah-ubah. Variabel dapat dipandang sebagai abstraksi dari lokasi. Hasil evaluasi dari variabel adalah nilai dari variabel itu. Nilai dari suatu variabel dapat diubah dengan assignment statement. Sebuah assignment statement terdiri dari sebuah variabel di sebelah kirinya dan suatu ekspresi disebelah kanannya.

Algoritmik	Bahasa Pascal	Bahasa C
Deskripsi Jumlah B B1 + B2	Jumlah := B1 + B2	Jumlah = B1 + B2

Variabel jumlah diubah nilainya menjadi nilai dari ekspresi $B1 + B2$ setelah dievaluasi. Dalam suatu program Pascal maupun C, setiap variabel yang akan digunakan terlebih dahulu dideklarasikan, dimana setiap variabel harus mempunyai tipe. Deklarasi variabel berguna untuk memberi informasi kepada compiler serta membantu programmer untuk berpikir secara jelas dan berencana.

Algoritmik	Bahasa Pascal	Bahasa C
Deklarasi B1,B2,jumlah:integer	Var B1,B2,jumlah:integer;	Int B1,B2,jumlah;

Aturan penamaan suatu variabel

- a) Pada bahasa pascal penulisan dengan huruf besar dan kecil tidak dibedakan, sedangkan pada bahasa C dibedakan (Case Sensitif).
- b) Boleh terdiri dari angka dengan syarat penulisannya setelah huruf, misal :

Umur31

Sepeda3

- c) Tidak boleh mengandung spasi, terdiri dari underscore (karakter `_`), misal :

Suhu_ruang (boleh)

Suhu ruang (tidak boleh)

SuhuRuang (boleh)

- d) Tidak boleh ada 2 atau lebih nama yang sama
- e) Tidak boleh menggunakan reserved word (kata kunci), misal :

∅ Dalam pascal : program, end, begin, var, dll

∅ Dalam C : #include, void, dll

8) Konstanta

Variabel yang mempunyai nilai yang sifatnya tidak bisa diubah, nilai ditentukan pada saat pendefinisian. Misal :

`<nama konstanta1> = <nilai1>;`

`Phi = 3.14;`

9) Jenis/Tipe Data

Pada waktu suatu variabel dideklarasikan, maka tipenya sekaligus ditentukan. Tipe dari suatu variabel menyatakan :

- a) Jenis nilai yang dapat disimpan dalam lokasi memori untuk variabel tersebut, (membatasi himpunan nilai-nilai yang dapat dipunyai variabel tersebut).
- b) Jenis operasi yang dapat dilakukan terhadap variabel bersangkutan.

10) Ekspresi Matematika

Ekspresi matematika adalah kalimat matematika yang akan memberikan nilai jika dievaluasi.

- a) Ekspresi aritmatika : Suatu ekspresi matematika yang memberikan hasil bertipe angka jika dievaluasi. Operator yang digunakan : +, -, *, /, (,)

Contoh :

Ø `6 * 5 + 7 à 37`

Ø `4 * 6 / 3 à 8`

- b) Ekspresi Logika : Suatu ekspresi matematika yang memberikan hasil berupa nilai kebenaran (benar/salah, true/false). Operator yang digunakan :

(1) Perbandingan : =, <, >, <=, >=, <>

(2) Logika : and, or

Contoh :

Ø True and true à true

Ø True and false à false

Ø (4<=5) and (5>=1)

Ø True and true à true

11) Prioritas Operator

Sebuah ekspresi matematika yang terdiri dari beberapa operator, beberapa operator dievaluasi terlebih dahulu dibandingkan operator yang lain. Sebuah operator didahulukan berarti mempunyai prioritas lebih tinggi. Urutan prioritas :

a) Operator aritmatika

b) Operator logika dan perbandingan

Contoh : $2 + 5 <= 5 + 7$

à $7 <= 12$

à true

Urutan prioritas operator aritmatika :

a) *, /

b) +, -

Jika 2 operator mempunyai prioritas sama, maka yang paling kiri didahulukan.

Contoh :

Ø $7 + 5 * 6 - 10$ à $7 + 30 - 10$ à $37 - 10$ à 27

Ø $(4 + 5) / (2 + 1)$ à $9 / 3$ à 3

(Ekspresi yang terletak dalam tanda kurung harus didahulukan).

Aritmatika	Arti	Pascal	C
+	Penjumlahan	+	+
-	Pengurangan	-	-
div	Pembagian integer	div	/
modulo	Sisa pembagian	mod	%

Dalam bahasa pascal dan bahasa C, pernyataan yang berkaitan dengan operasi dasar adalah sebagai berikut :

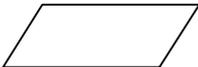
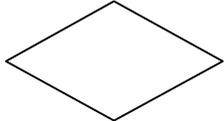
Pernyataan	Algoritmik	Pascal	C
Input	read	read atau readln	scanf
output	write	write atau writeln	printf
Penugasan	β	:=	=
Akhir pernyataan		;	;

Fungsi yang sering digunakan :

Pascal	C	Arti
Sqr	Pow	Kuadrat
Sqrt	Sqrt	Akar kuadrat
Ln	Log	Logaritma alami
Exp	Exp	eksponensial

12) Flowchart

Kadang-kadang perlu digambarkan bagaimana arus data dari algoritma yang sudah dibuat, terutama kalau algoritma sudah cukup kompleks. Untuk itu algoritma dapat disajikan dalam bentuk flowchart. Simbol yang diperlukan diantaranya :

No	Simbol	Makna
1.		start / mulai end / selesai
2.		Input/output
3.		Kondisi
4.		Nilai awal/inisialisasi
5.		Perulangan for
6.		Proses/penugasan

c. Rangkuman 1

- 1) Dalam membuat suatu program komputer, menyusun algoritma adalah langkah pertama yang harus dilakukan
- 2) Dalam membuat algoritma dapat digunakan flowchart

d. Tugas 1

- 1) Pelajarilah uraian materi tentang prinsip algoritma!
- 2) Pelajarilah uraian materi tentang prinsip pemrograman!
- 3) Pelajarilah uraian materi tentang prinsip dari mekanisme oleh pemroses
- 4) Pelajarilah beberapa macam bahasa pemrograman
- 5) Pelajarilah uraian materi tentang dasar-dasar algoritma
- 6) Pelajarilah simbol-simbol dari flowchart

e. Tugas Formatif 1

- 1) Sebutkan bahasa pemrograman prosedural?
- 2) Memahami sintaks program dengan menggunakan bahasa pascal dan bahasa C?
- 3) Mampu membuat algoritma mencari jumlah 3 buah bilangan bulat dengan flowchart?
- 4) Mampu membuat algoritma mencari hasil kali dari 2 buah bilangan dengan flowchart?

f. Kunci Jawaban Formatif 1

- 1) Bahasa pemrograman prosedural : Pascal, C, Cobol, Basic, Fortran.
- 2) Sintaks program dengan menggunakan bahasa pascal dan bahasa C adalah sebagai berikut :

Bahasa Pascal

```
Program nama_program;  
[deklarasi label]  
[deklarasi konstan]  
[deklarasi tipe]  
[deklarasi variable]  
[deklarasi subprogram]  
Begin  
    Pernyataan;  
    .....  
    Pernyataan  
End.
```

Bahasa C

```
#include <stdio.h>
[deklarasi subprogram]
Main()
{
    [deklarasi variabel]
    Pernyataan;
    .....
    Return 0;
}
```

- 3) Algoritma mencari jumlah 3 buah bilangan bulat dengan flowchart =

Algoritma mencari jumlah 3 buah bilangan bulat

Deklarasi

a,b,c : integer

jumlah: integer

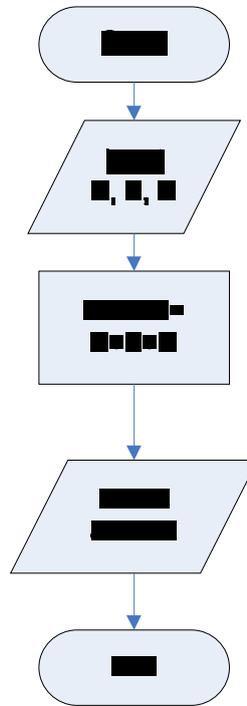
Deskripsi

Read(a,b,c)

Jumlah β $a + b + c$

Write(jumlah)

Flowchartnya :



Gambar 2.

Flowchart Algoritma Mencari Jumlah 3 Buah Bilangan Bulat

4) Algoritma mencari hasil kali dari dua buah bilangan dengan flowchart =

Algoritma mencari hasil kali dari dua buah bilangan

Deklarasi

a,b : integer

hasil : integer

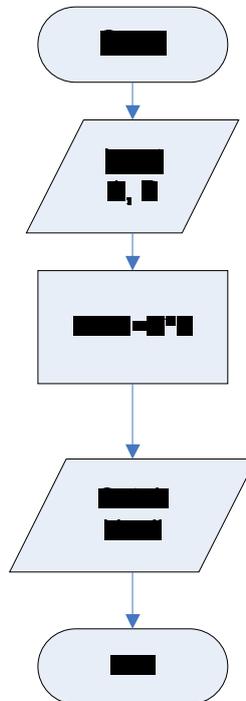
Deskripsi

Read(a,b)

hasil \leftarrow a * b

Write(hasil)

Flowchartnya :



Gambar 3

Flowchart Algoritma Mencari Hasil Kali dari Dua Buah Bilangan

g. Lembar Kerja 1

Alat dan Bahan

Seperangkat komputer yang telah diinstal dengan Microsoft Visio dan program turbo pascal dan turbo C.

Kesehatan dan Keselamatan Kerja

- 1) Berdo'alah sebelum memulai kegiatan belajar !
- 2) Bacalah dan pahami petunjuk praktikum pada setiap lembar kegiatan belajar!
- 3) Bila telah selesai matikan komputer sesuai dengan prosedur!

Langkah Kerja

- 1) Kumpulkan data yang diperlukan, kemudian coba analisis data yang ada dan kelompokkan sesuai dengan jenis datanya.
- 2) Tentukan variabel dan jenis datanya sesuai dengan tipe datanya.
- 3) Analisis struktur algoritma dari data yang ada, tentukan data yang ada menggunakan runtutan, pemilihan atau pengulangan.
- 4) Buat flowchartnya berdasarkan hasil dari analisis struktur algoritma menggunakan program bantu untuk membuat flowchart yaitu Microsoft Visio.
- 5) Aplikasikan flowchart dalam bahasa pemrograman. Bahasa pemrograman yang dipakai yaitu bahasa pascal atau bahasa C.
- 6) Pelajari perbedaan penulisan dan beberapa operator pada bahasa Pascal maupun bahasa C.

2. Kegiatan Belajar 2 : Mempersiapkan pokok-pokok kebutuhan data

a. Tujuan Kegiatan Pemelajaran

Setelah mempelajari kegiatan belajar ini peserta diklat mampu menganalisa data sehingga dapat mengidentifikasinya menjadi beberapa tipe data sesuai dengan sifat datanya.

b. Uraian materi 2

1) Tipe Data Dasar

a) Tipe data bilangan bulat

Nama tipe : integer

Range nilai : -32.768 sd 32.767

Operator binary :

Ø +, -, * : integer x integer à integer

Ø / : integer x integer à real

Ø Mod, div : integer x integer à integer

Keterangan :

§ Mod adalah sisa hasil bagi dari dua bilangan yang bertipe integer

§ Div adalah hasil bagi dari dua bilangan yang bertipe integer

Contoh :

$$15 \text{ div } 3 = 5$$

$$15 \text{ mod } 3 = 0$$

$$45 \text{ div } 4 = 11$$

$$45 \text{ mod } 4 = 1$$

Operator unary :

- : integer à integer

b) Tipe data real

Nama tipe : real

Range nilai : 2.9×10^{-39} s/d 1.7×10^{38}
 -1.7×10^{38} s/d -2.9×10^{-39}

Operator : +, -, *, / : real x real à real

: real x integer à real

: integer x real à real

c) Tipe data karakter/symbol

Nama tipe : char

Range nilai : 1 simbol/karakter dalam tanda petik

d) Tipe data string

Nama tipe : string

Range nilai : rangkaian karakter dalam tanda petik

Panjang string : 0 s/d 255

Contoh : 'Suhu ruangan 55⁰ C'

Operator : + : string x string à string

Contoh : 'saya' + ' pergi' à 'saya pergi'

e) Tipe data boolean

Nama tipe : Boolean

Range nilai : True dan False

Operator:

Ø Binary :

or, and, xor : boolean x boolean à boolean

Ø Unary : not : boolean à boolean

Not true à false

Tabel Kebenaran :

P	Q	P and Q	P or Q	P xor Q
True	True	True	True	False
True	False	False	True	True
False	True	False	True	True
False	False	False	False	False

2) Pemilihan / Kontrol Program

Salah satu kemampuan komputer adalah dapat melakukan proses pemilihan dari beberapa alternatif sesuai dengan kondisi yang diberikan. Dalam persoalan sehari-haripun hampir selalu ada kondisi dimana kita harus memilih diantara alternatif-alternatif yang ada. Sebagai contoh, seorang siswa memperoleh nilai 75. Apakah dengan nilai tersebut siswa itu lulus? Jika batas kelulusan minimal 60 maka siswa tersebut lulus ujian. Jika kurang dari 60 maka siswa tersebut tidak lulus ujian. Bahasa pemrograman seperti pascal maupun C juga mendukung pernyataan bersyarat. Tata cara penulisannya tersaji berikut ini :

Algoritmik	Pascal	C
If (kondisi) then aksi end if	If (kondisi) then aksi;	If (kondisi) aksi;
If (kondisi) then aksi1 else aksi2 end if	If (kondisi) then aksi1 else aksi2;	If (kondisi) aksi1; else aksi2;
Pernyataan majemuk	beginend	{.....}

Baik aksi, aksi1, maupun aksi2 bisa merupakan pernyataan tunggal maupun pernyataan majemuk. Selain itu, bila kondisi bertipe ordinal, seperti : integer, byte atau yang lain (kecuali real atau string), bisa digunakan pernyataan dibawah ini :

Algoritmik	Pascal	C
<pre> case (nama) <label1>:aksi1 <label1>:aksi2 <label1>:aksiN else : aksiX endcase </pre>	<pre> case (nama) of <label1>:aksi1; <label1>:aksi2; <label1>:aksiN; else : aksiX; end; </pre>	<pre> switch (nama) { Case label1:aksi1; break; Case label2:aksi2; break; Case labelN:aksiN; break; default : aksix;} </pre>

3) Perulangan/ Loop

Salah satu kelebihan komputer dibandingkan dengan manusia adalah kemampuannya untuk melaksanakan suatu instruksi berulang kali tanpa mengenal lelah dan bosan. Didalam algoritma, pengulangan atau kalang (repetition atau loop) dapat dilakukan sejumlah kali, atau sampai kondisi berhenti pengulangan tercapai.

a) Struktur pengulangan

Struktur pengulangan terdiri atas dua bagian :

- Ø Kondisi pengulangan, yaitu ekspresi boolean yang harus dipenuhi untuk melakukan pengulangan. Kondisi ini ada yang dinyatakan secara eksplisit oleh pemrogram atau dikelola sendiri oleh komputer (implisit).
- Ø Badan (body) pengulangan, yaitu satu atau lebih aksi yang akan diulang.

b) Struktur WHILE-DO

Bentuk umum struktur while-do adalah :

```
While <kondisi> do  
    Aksi  
endwhile
```

Penjelasan :

Aksi akan dilaksanakan berulang kali sepanjang <kondisi> boolean masih tetap bernilai true. Jika <kondisi> bernilai false, badan pengulangan tidak akan dilaksanakan. Pengulangan selesai.

c) Struktur REPEAT-UNTIL

Bentuk umum struktur repeat-until adalah :

```
Repeat  
    aksi  
Until <kondisi>
```

Penjelasan :

Notasi ini mendasarkan pengulangan pada kondisi berhenti. Aksi didalam badan kalang diulang sampai kondisi berhenti boolean bernilai true. Dengan kata lain, jika kondisi berhenti masih salah, pengulangan masih terus dilakukan. Karena pengulangan harus berhenti, maka didalam badan kalang harus ada aksi yang mengubah harga kondisi.

Struktur repeat-until memiliki makna yang sama dengan while-do, dan dalam beberapa masalah kedua struktur tersebut komplemen satu sama lain. Namun ada perbedaan yang mendasar diantara keduanya. Pada struktur repeat-until, aksi dilakukan minimal satu kali, karena kondisi pengulangan diperiksa pada akhir struktur, sedangkan pada struktur while-do kondisi pengulangan

diperiksa pada awal struktur sehingga memungkinkan pengulangan tidak akan pernah dilaksanakan bila kondisi pengulangan bernilai false.

d) Struktur FOR

Struktur for digunakan untuk menghasilkan pengulangan sejumlah kali tanpa penggunaan kondisi apapun. Struktur ini menyebabkan aksi diulangi sejumlah kali (tertentu). Bentuk umum struktur FOR ada 2 macam : menaik (ascending) atau menurun (descending).

FOR menaik :

```
for peubah $\beta$  nilai_awal to nilai_akhir do
    Aksi
endfor
```

Keterangan :

- Ø Aksi akan dilakukan berulang-ulang selama peubah diberi nilai dari nilai_awal s/d nilai_akhir dengan step1.
- Ø Peubah : bertipe integer atau karakter
- Ø Nilai_awal, nilai_akhir : hasilnya bertipe sama dengan tipe peubah
- Ø Nilai_akhir \geq nilai_awal, jika tidak maka aksi tidak akan dilakukan
- Ø Aksi : satu aksi atau beberapa aksi dalam blok begin – end.

FOR menurun :

```
for peubah $\beta$  nilai_akhir downto nilai_awal do
    Aksi
Endfor
```

Keterangan :

- Ø Nilai_akhir harus lebih besar atau sama dengan nilai_awal. Jika nilai_akhir lebih kecil dari nilai_awal, maka badan pengulangan tidak dimasuki.
- Ø Pada awalnya, peubah diinisialisasi dengan nilai_akhir. Nilai peubah secara otomatis berkurang satu setiap kali aksi diulangi, sampai akhirnya nilai peubah sama dengan nilai_awal
- Ø Peubah : bertipe integer atau karakter
- Ø Jumlah pengulangan yang terjadi adalah nilai_awal-nilai_akhir +1.

c. Rangkuman 2

Dengan mengetahui tipe data dasar pada suatu bahasa pemrograman tertentu maka diharapkan peserta dapat mengidentifikasi suatu data sesuai dengan tipe datanya. Dalam melakukan pemilihan dan perulangan peserta diharapkan mampu memilih jenis-jenis pemilihan dan perulangan yang sesuai.

d. Tugas 2

- 1) Pelajarilah uraian materi tentang tipe data dasar !
- 2) Pelajarilah uraian materi tentang pemilihan!
- 3) Pelajarilah uraian materi tentang perulangan !

e. Tes Formatif 2

- 1) Tentukan jenis data dari data yang terdapat pada variabel dibawah ini :
 - a) Nama : shofwa, zaki, imam, mira
 - b) Alamat asal : jogja, solo, semarang
 - c) Status Menikah : ya, tidak

d) Berat badan : 23.3, 40.8, 50.7, 1.0

e) Umur : 23, 45, 65, 1, 3, 7

- 2) Buatlah algoritma dan flowchartnya untuk menentukan bilangan terbesar antara dua bilangan bulat?
- 3) Sebutkan 3 macam notasi struktur pengulangan ?
- 4) Buatlah algoritma dan flowchartnya untuk mencetak bilangan 1 sampai 4 menggunakan perulangan ?

f. Kunci jawaban tes formatif 2

1) Tipe Data dari masing-masing variabel adalah sebagai berikut :

- a) Tipe data variabel nama adalah string
- b) Tipe data variabel alamat asal adalah string
- c) Tipe data variabel status menikah adalah boolean
- d) Tipe data variabel berat badan adalah real
- e) Tipe data variabel umur adalah integer

2) Algoritma untuk menentukan bilangan terbesar antara dua bilangan bulat :

Algoritma maksimum

{menentukan nilai terbesar antara 2 buah bilangan bulat}

Deklarasi

A, B : integer

Deskripsi

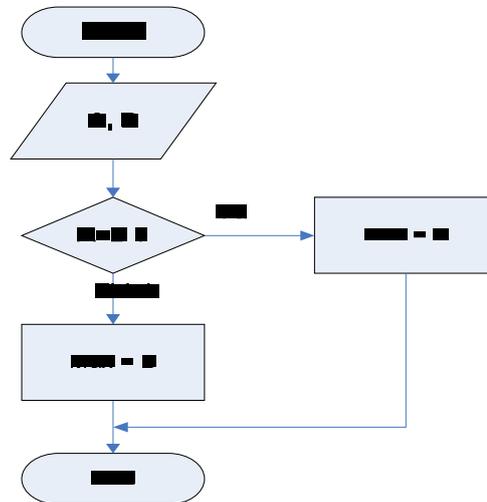
read(A,B)

if (A>B) then write('Bilangan terbesar adalah ='A)

else write('Bilangan terbesar adalah =' B)

endif

Flowchart :



Gambar 4

Flowchat untuk Menentukan Bilangan Terbesar antara Dua Bilangan Bulat

3) Tiga Macam notasi struktur pengulangan yaitu :

a) Struktur WHILE-DO

Bentuk umum struktur while-do adalah :

```
While <kondisi> do
    Aksi
Endwhile
```

b) Struktur REPEAT-UNTIL

Bentuk umum struktur repeat-until adalah :

```
Repeat
    aksi
Until <kondisi>
```

c) Struktur FOR

FOR menaik :

```
for peubah nilai_awal to nilai_akhir do
    Aksi
endfor
```

FOR menurun :

```
for peubah nilai_akhir downto nilai_awal do
  Aksi
endfor
```

4) Algoritma untuk mencetak bilangan 1 sampai 4 menggunakan perulangan

a) For loop

Algoritma cetak_angka

Deklarasi

I : integer

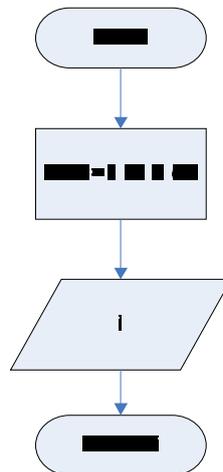
Deskripsi

For i 1 to 4 do

Write(i)

Endfor

Flowchart :



Gambar 5
Flowchat Mencetak Bilangan 1 sampai 4 dengan For Loop

b) While loop

Algoritma cetak_angka

Deklarasi

I : integer

Deskripsi

I = 1

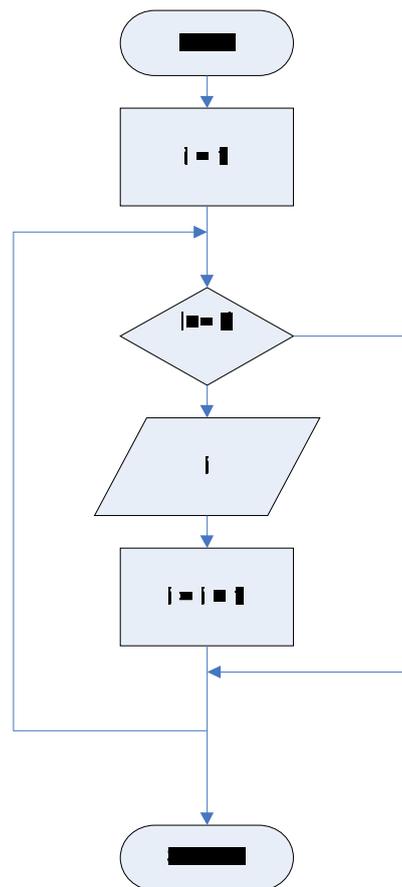
While(i <= 4) do

Write(i)

I = i + 1

Endwhile

Flowchart :



Gambar 6

Flowchat Mencetak Bilangan 1 sampai 4 dengan While Loop Perulangan

c) Repeat until loop

Algoritma cetak_angka

Deklarasi

I : integer

Deskripsi

I_B 1

repeat

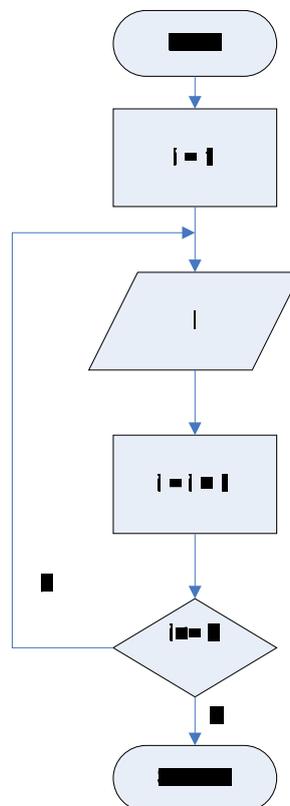
Write(i)

I_B i+1

Until(i>4)

Endrepeat

Flowchart :



Gambar 7

Flowchat Mencetak Bilangan 1 sampai 4 dengan Repeat Until

g. Lembar Kerja 2

Alat dan Bahan

Seperangkat komputer yang telah diinstal dengan Microsoft Visio dan program turbo pascal dan turbo C.

Kesehatan dan Keselamatan Kerja

- 1) Berdo'alah sebelum memulai kegiatan belajar !
- 2) Bacalah dan pahami petunjuk praktikum pada setiap lembar kegiatan belajar!
- 3) Bila telah selesai matikan komputer sesuai dengan prosedur!

Langkah Kerja

- 1) Pelajari dan data yang ada dan tentukan variabel dengan tipe data yang sesuai.
- 2) Pilihlah jenis pemilihan kondisi atau perulangan yang sesuai dengan masalah atau data
- 3) Buatlah flowchart dengan microsoft visio
- 4) Aplikasikan dalam program menggunakan turbo pascal dan turbo C

3. Kegiatan belajar 3 : Melakukan perancangan pengumpulan data

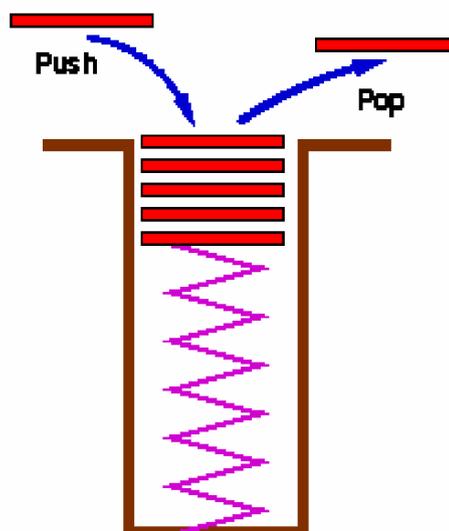
a. Tujuan Kegiatan Pemelajaran

Peserta diklat mampu Melakukan perancangan pengumpulan data

b. Uraian materi 3

1) Stack/Tumpukan

Salah satu konsep yang sangat berguna di dalam ilmu komputer adalah satu bentuk struktur data yang disebut tumpukan (stack). Secara sederhana, tumpukan bisa diartikan sebagai suatu kumpulan data yang seolah-olah ada data yang diletakkan di atas data yang lain. Satu hal yang perlu diingat bahwa dalam stack bisa menambah (menyisipkan) data, dan mengambil (menghapus) data lewat ujung yang sama, yang disebut sebagai ujung atas tumpukan (top of stack). Stack merupakan suatu senarai (list) yang mempunyai sifat "masuk terakhir keluar pertama (last in first out – LIFO)



Gambar 8. Ilustrasi Stack

Operasi pada tumpukan

Ada dua operasi dasar yang bisa dilaksanakan pada sebuah tumpukan, yaitu operasi menyisipkan data atau mempush data dan operasi menghapus data atau mempop data. Contoh pemakaian tumpukan dalam membalik kalimat dengan melakukan pembalikan perkata, sebagai contoh, jika kalimat yang dibaca adalah :

BELAJAR PASCAL ADALAH MUDAH DAN MENYENANGKAN

Setelah dibalik, maka kalimat diatas menjadi :

NAKGNANEYNEM NAD HADUM HALADA LACSAP RAJALEB

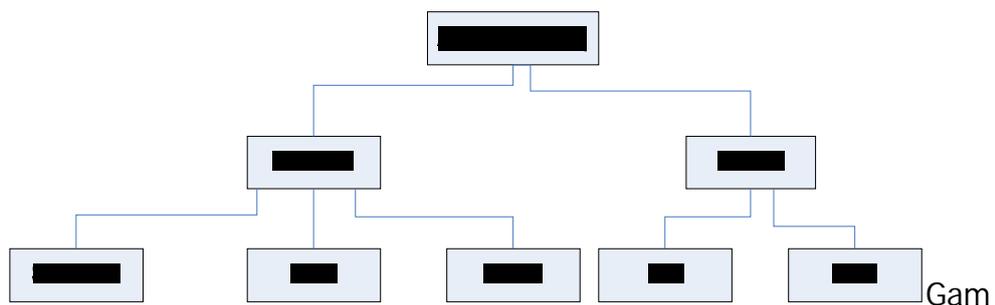
2) Queue / Antrian

Antrian merupakan satu jenis struktur data yang sering digunakan untuk mensimulasikan keadaan dunia nyata. Antrian adalah suatu kumpulan data yang mana penambahan elemen hanya bisa dilakukan pada suatu ujung (disebut dengan sisi belakang atau rear), dan penghapusan (pengambilan elemen) dilakukan lewat ujung lain (disebut dengan sisi depan atau front). Istilah yang digunakan apabila seseorang masuk dalam sebuah antrian adalah Enqueue. Sedangkan istilah yang sering dipakai bila seseorang keluar dari antrian adalah Dequeue. Tumpukan menggunakan prinsip "masuk terakhir keluar pertama" atau LIFO (Last In First Out), maka pada antrian prinsip yang digunakan adalah "masuk pertama keluar pertama" atau FIFO (First In First Out). Dengan kata lain, urutan keluar elemen akan sama dengan urutan masuknya. Antrian banyak kita jumpai dalam kehidupan sehari-hari. Mobil-mobil yang antri membeli karcis di pintu jalan tol akan membentuk antrian; orang-orang yang membeli karcis untuk menyaksikan film akan membentuk

antrian, dan contoh-contoh lain yang banyak dijumpai dalam kehidupan sehari-hari.

3) Tree / Pohon

Tree adalah bentuk struktur data tak linear yang mempunyai sifat-sifat dan ciri-ciri khusus. Struktur ini biasanya digunakan untuk menggambarkan hubungan yang bersifat hirarkis antara elemen-elemen yang ada. Contoh paling sederhana yang bisa kita lihat dalam kehidupan sehari-hari, khususnya dalam keluarga adalah silsilah keluarga. Gambar di bawah menunjukkan pohon silsilah keluarga Abdul Kholiq sampai dengan cucunya. Pohon seperti gambar di bawah disebut dengan linear chart dimana setiap elemen bisa mempunyai lebih dari dua buah cabang.

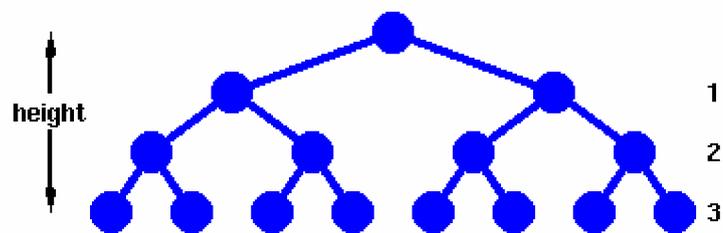


Gambar 9. Contoh Nyata Fungsi Tree

Istilah-Istilah Dasar

Secara sederhana pohon bisa didefinisikan sebagai kumpulan elemen yang salah satu elemennya disebut dengan akar (root), dan sisa elemen yang lain (yang disebut simpul) terpecah menjadi sejumlah menjadi himpunan yang saling tidak berhubungan satu sama lain, yang disebut dengan subpohon (subtree), atau juga disebut dengan cabang. Jika kita lihat pada setiap subpohon, maka subpohon inipun juga mempunyai akar dan sub-subpohonnya masing-masing. Dalam

gambar di atas akarnya adalah Abdul Kholiq, yang mempunyai dua subpohon. Subpohon yang pertama berakar pada Arwani, dan subpohon yang kedua berakar pada Rif'ah. Selanjutnya jika kita lihat pada subpohon Arwani, maka subpohon ini juga mempunyai tiga buah subpohon, yaitu yang berakar pada Shofwa, Mila dan Didin. Begitu juga dengan subpohon yang berakar pada Rif'ah, maka subpohon ini juga mempunyai dua buah subpohon, yaitu yang berakar pada Ida dan Alin.



Gambar 10. Ilustrasi Fungsi Tree

Jika kita perhatikan gambar tree di atas, maka akar dinyatakan sebagai tingkat 0 dan simpul-simpul lainnya dinyatakan bertingkat 1 lebih tinggi dari ayahnya. Selain definisi di atas ada juga beberapa buku yang menyatakan bahwa tingkat (level) suatu simpul ditentukan dengan pertama kali menentukan akar sebagai bertingkat 1. jika suatu simpul dinyatakan sebagai tingkat N, maka simpul-simpul yang merupakan anaknya dikatakan berada dalam tingkat N+1.

Selain tingkat, juga dikenal istilah derajat (degree) dari suatu simpul. Derajat suatu simpul dinyatakan sebagai banyaknya anak atau turunan dari simpul tersebut.

Daun juga sering disebut dengan simpul luar (external node). Sehingga simpul lain , kecuali akar, juga sering disebut dengan simpul dalam (internal node).

Tinggi (height) atau kedalaman (depth) dari suatu pohon adalah tingkat maksimum dari simpul dalam pohon tersebut dikurangi dengan 1.

Ancestor suatu simpul adalah semua simpul yang terletak dalam satu jalur dengan simpul tersebut dari akar sampai simpul yang ditinjau.

Hutan (forest) adalah kumpulan sejumlah pohon yang tidak saling berhubungan.

4) Graph

Graph bisa dibayangkan sebagai kumpulan obyek atau aktivitas, sebagai contoh, rute bis kota dari satu terminal ke terminal lain, rute perjalanan pak pos pada saat ia mengantar surat dari satu rumah ke rumah lain, dan contoh-contoh lain yang bisa disajikan sebagai suatu graph. Contoh kedua diatas merupakan contoh klasik dengan teori graph yang lebih dikenal dengan travelling salesman problem atau shortest path problem, yang pada prinsipnya mencari jalur terpendek dari semua tempat yang harus dipenuhi, sehingga bisa menghemat waktu, tenaga, maupun biaya. Graph secara umum bisa didefinisikan sebagai kumpulan titik (nodes atau vertices) dan garis (arcs atau edges). Karena garis selalu diawali dengan suatu titik dan diakhiri pada titik yang lain, maka garis bisa dituliskan sebagai pasangan antara dua titik.

c. Rangkuman 3

Struktur data yang digunakan pada algoritma pemrograman ada beberapa metode, yaitu : stack, queue, tree, graph. Pemilihan metode yang digunakan sangat bergantung dengan data, baik dalam proses pengumpulan maupun penyimpanannya.

d. Tugas 3

- 1) Pelajarilah uraian materi tentang prinsip dan fungsi stack!
- 2) Pelajarilah uraian materi tentang prinsip dan fungsi antrian (queue)!
- 3) Pelajarilah uraian materi tentang prinsip dan fungsi pohon (tree)!
- 4) Pelajarilah uraian materi tentang prinsip dan fungsi graph!

e. Tes Formatif 3

- 1) Diberikan data sebagai berikut yang diimplementasikan pada suatu stack atau antrian:

a) Data = 081822828998

Lakukan operasi :

- § pop
- § push = 67
- § push = 55
- § pop
- § pop

b) Data = Yogyakarta

Lakukan operasi :

- push = Kota P
- push = elajar &
- push = Bud

- push = aya
- pop
- pop

c) Data = melakukan kebaikan

Lakukan operasi :

- pop
- pop
- pop
- push = ya

d) Data = Belajar yang Rajin

Lakukan operasi :

- push = biar sukses
- push =
- pop
- pop

e) Data = Teknik informatika

Lakukan operasi :

- pop
- pop
- pop
- pop
- push = oke

2) Diberikan data sebagai berikut yang diimplementasikan pada suatu queue atau antrian :

a) Data = shofwatul

Lakukan operasi :

§ Enqueue = ma

§ Dequeue

§ Dequeue

§ Enqueue = iyun

§ Enqueue = da ri

b) Data = sekolah

Lakukan operasi :

- Dequeue

- Enqueue = jangan

- Dequeue

c) Data = universi

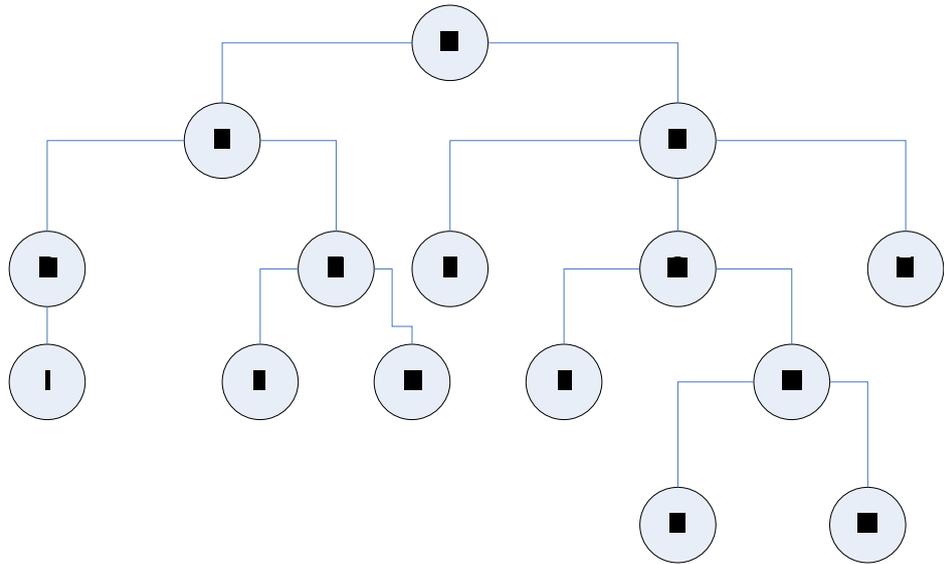
Lakukan operasi :

- Enqueue = tas

- Dequeue

- Dequeue

3) Jelaskan dan sebutkan komponen dari tree yang termasuk tingkat (level), derajat (degree), daun (leaf), tinggi (height) atau kedalaman (depth) dan ancestor pada gambar tree dibawah ini



Gambar 11. Gambar Soal 3)

4) Apa yang anda ketahui tentang graph dan bagaimana prinsip dari graph secara umum ?

f. Kunci Jawaban tes formatif 3

1) Operasi pop adalah operasi menghapus data pada suatu stack. Operasi push adalah operasi menyisipkan data pada stack.

a) Data = 081822828998

∅ Operasi pop → 08182282899

∅ Operasi push (67) → 0818228289967

∅ Operasi push (55) → 081822828996755

∅ Operasi pop → 08182282899675

∅ Operasi pop → 0818228289967

b) Data = yogyakarta

∅ Operasi push(Kota P)

→ Yogyakarta Kota P

∅ Operasi push(elajar &)

→ Yogyakarta Kota Pelajar &

- Ø Operasi push(Bud)
 - à Yogyakarta Kota Pelajar &Bud
- Ø Operasi push(aya)
 - à Yogyakarta Kota Pelajar &Budaya
- Ø Operasi pop
 - à Yogyakarta Kota Pelajar &Buday
- Ø Operasi pop
 - à Yogyakarta Kota Pelajar &Buda

c) Data = melakukan kebaikan

- Ø Operasi pop à melakukan kebaika
- Ø Operasi pop à melakukan kebaik
- Ø Operasi pop à melakukan kebai
- Ø Operasi pop à melakukan keba
- Ø Operasi push(ya) à melakukan kebaya

d) Data = Belajar yang Rajin

- Ø Operasi push(biar sukses)
 - à Belajar yang Rajinbiar sukses
- Ø Operasi push(...)
 - à Belajar yang Rajinbiar sukses....
- Ø Operasi pop
 - à Belajar yang Rajinbiar sukses...
- Ø Operasi pop
 - à Belajar yang Rajinbiar sukses..

e) Data = Teknik informatika

- Ø Operasi pop à Teknik informatik
- Ø Operasi pop à Teknik informati
- Ø Operasi pop à Teknik informat
- Ø Operasi pop à Teknik informa

Ø Operasi push(oke) à Teknik informaoke

2) Antrian (queue) adalah suatu kumpulan data yang mana penambahan elemen hanya bisa dilakukan pada suatu ujung (disebut dengan sisi belakang atau rear), dan penghapusan (pengambilan elemen) dilakukan lewat ujung lain (disebut dengan sisi depan atau front). Enqueue adalah Istilah yang digunakan apabila seseorang masuk dalam sebuah antrian. Sedangkan Dequeue adalah istilah yang sering dipakai bila seseorang keluar dari antrian .

a) Data = shofwatul

- Ø Operasi enqueue(ma) à shofwatulma
- Ø Operasi dequeue à hofwatulma
- Ø Operasi dequeue à ofwatulma
- Ø Operasi enqueue(iuyun) à ofwatulmaiuyun
- Ø Operasi enqueue(da ri) à ofwatulmaiuyunda ri

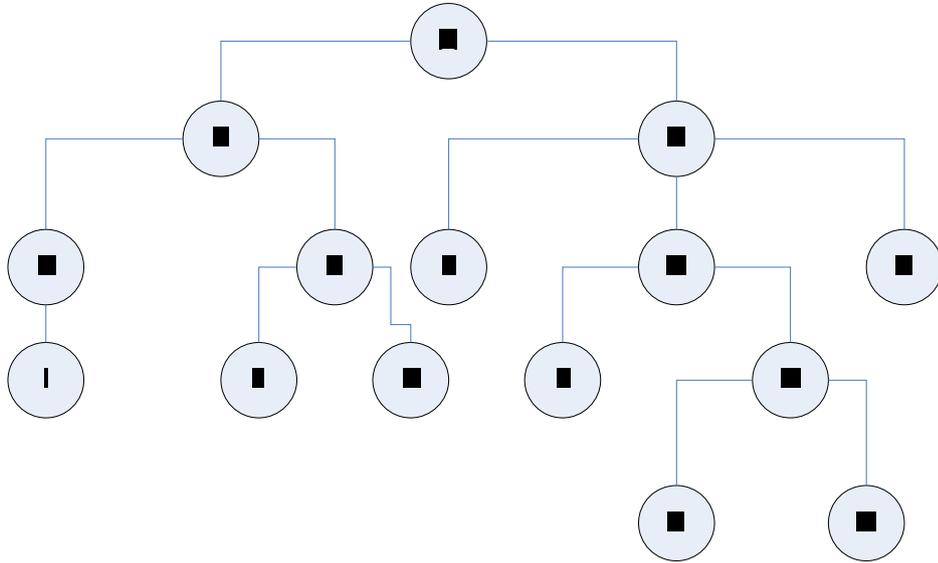
b) Data = sekolah

- Ø Operasi dequeue à ekolah
- Ø Operasi enqueue(jangan) à ekolahjangan
- Ø Operasi dequeue à kolahjangan

c) Data = universi

- Operasi enqueue(tas) à universitas
- Operasi enqueue à niversitas
- Operasi dequeue à iversitas

- 3) Tree adalah bentuk struktur data tak linear yang mempunyai sifat-sifat dan ciri-ciri khusus.



Gambar 12. Gambar Jawaban Soal 3)

- a) tingkat (level) : tingkat 0/1 à A
 tingkat 1/2 à B, C
 tingkat 2/3 à D, E, F, G, H
 tingkat 3/4 à I, J, K, L, M
 tingkat 4/5 à N, O
- b) derajat (degree) : Simpul A mempunyai derajat 2
 Simpul B mempunyai derajat 2
 Simpul C mempunyai derajat 3
- c) daun (leaf) : Simpul F, H, I, J, K, L, N, O berderajat 0
- d) tinggi (height) atau kedalaman (depth): semua daun yang berakar pada A mempunyai tinggi atau kedalaman 4.
- e) Ancestor dari simpul L adalah adalah A, C dan G

- 4) Graph secara umum bisa didefinisikan sebagai kumpulan titik (nodes atau vertices) dan garis (arcs atau edges). Karena garis selalu diawali dengan suatu titik dan diakhiri pada titik yang lain, maka garis bisa dituliskan sebagai pasangan antara dua titik. Prinsip graph secara umum adalah mencari jalur terpendek dari semua tempat yang harus dipenuhi, sehingga bisa menghemat waktu, tenaga, maupun biaya.

g. Lembar Kerja3

Alat dan Bahan

Seperangkat alat tulis

Kesehatan dan Keselamatan Kerja

- 1) Berdo'alah sebelum memulai kegiatan belajar !
- 2) Bacalah dan pahami petunjuk praktikum pada setiap lembar kegiatan belajar!
- 3) Bila telah selesai matikan komputer sesuai dengan prosedur!

Langkah Kerja

- 1) Pelajari dan analisis data yang ada kemudian pelajari jika data tersebut menggunakan struktur data stack, queue, tree dan graph
- 2) Pelajari beberapa istilah yang ada pada struktur data stack, queue, tree dan graph.

BAB III EVALUASI

A. PERTANYAAN

1. Buatlah algoritma, flowchart kemudian translasikan kedalam bahasa pascal dan C kasus berikut : Carilah konversi suhu dari Celcius menjadi Reamur, Fahrenheit dan Kelvin
2. Buatlah algoritma dan flowchart dari kasus Konversikan nilai angka menjadiberikut : nilai huruf dengan ketentuan sebagai berikut :

Nilai angka	Nilai Huruf
0-20	E
21-40	D
41-60	C
61-80	B
81-100	A

B. KUNCI JAWABAN EVALUASI

1. Analisis :

Rumus konversi dari Celcius menjadi Reamur, Fahrenheit, dan Kelvin adalah sebagai berikut :

Ø Reamur : $\frac{4}{5}$ celcius

Ø Fahrenheit : $\frac{9}{5}$ celcius + 32

Ø Kelvin : celcius + 273

Algoritma :

Algoritma menghitung konversi suhu

Deklarasi

C : integer

K : real

F : real

R : real

Deskripsi

Read (C)

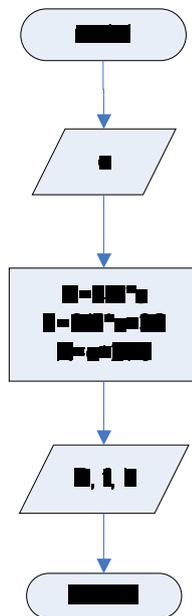
$R \leftarrow 4/5 * C$

$F \leftarrow 9/5 * C + 32$

$K \leftarrow C + 273$

Write(R,F,K)

Flowchart :



Gambar 13. Flowchart Jawaban Soal 1

Bahasa Pascal :

```
Program konversi_suhu;
Uses wincrt;
Var
    C      : integer;
    R, f, k : real;
Begin
    Write('Masukkan suhu derajat celcius : ');readln (c);
    R := 4/5+c;
    F := 9/5*c+32;
    K := c+273;
    Writeln('Reamur =',r);
    Writeln('Fahrenheit =',f);
    Writeln('Kelvin =',k);
End.
```

Bahasa C :

```
#include<stdio.h>
Main() {
    Int c;
    Float k, f, r;
    Printf ("Masukkan suhu derajat celcius : ");=
    Scanf("%d",&c);
    R = 4/5.0+c;
    F = 9/5.0*c+32;
    K = c+273;
    Printf("Reamur = %f",r);
    Printf("Fahrenheit = %f",f);
    Printf("Kelvin = %f",k);
    Return 0;}
}
```

2. Algoritma :

Algoritma konversi nilai

Deklarasi

Nilai : integer

Nilai_huruf : char

Deskripsi

Read(nilai)

If(nilai>0)and(nilai<=20)then nilai_huruf β 'E'

Else If(nilai>20)and(nilai<=40)then nilai_huruf β 'D'

Else If(nilai>40)and(nilai<=60)then nilai_huruf β 'C'

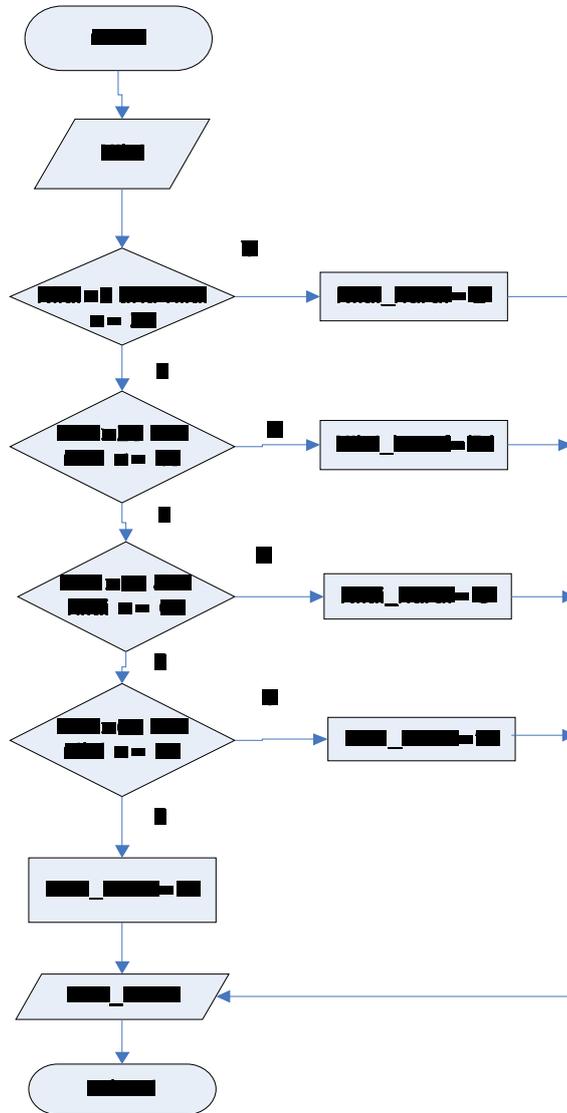
Else If(nilai>60)and(nilai<=80)then nilai_huruf β 'B'

Else nilai_huruf β 'A'

Endif

Write(nilai_huruf)

Flowchart :



Gambar 13. Flowchart Jawaban Soal 2

C. KRITERIA KELULUSAN

Aspek	Skor (1-10)	Bobot	Nilai	Keterangan
Soal 1		4		Syarat lulus nilai minimal 70 dan skor setiap aspek minimal 7
Soal 2		4		
Ketepatan waktu		2		
Nilai Akhir				

Kategori kelulusan:

70 – 79 : Memenuhi kriteria minimal. Dapat bekerja dengan bimbingan.

80 – 89 : Memenuhi kriteria minimal. Dapat bekerja tanpa bimbingan.

90 – 100 : Di atas kriteria minimal. Dapat bekerja tanpa bimbingan.

BAB IV

PENUTUP

Setelah menyelesaikan modul ini dan mengerjakan semua tugas serta evaluasi maka berdasarkan kriteria penilaian, peserta diklat dapat dinyatakan lulus/ tidak lulus. Apabila dinyatakan lulus maka dapat melanjutkan ke modul berikutnya sesuai dengan alur peta kedudukan modul, sedangkan apabila dinyatakan tidak lulus maka peserta diklat harus mengulang modul ini dan tidak diperkenankan mengambil modul selanjutnya.

DAFTAR PUSTAKA

- Hartanto, Budi. (2003). Pembuatan Program C Secara Mudah. . Andi Offset. Yogyakarta
- Hendrowati, Retno & Hariyanto, Bambang. (2000). Logika Matematika. Informatika. Bandung
- Kadir, Abdul. (2001). Pemrograman Dasar Turbo C. Andi Offset. Yogyakarta
- Munir, Rinaldi. (1999). Algoritma dan Pemrograman dalam Bahasa Pascal dan C. Informatika. Bandung
- Santosa, Insap. (1992). Struktur Data Menggunakan Turbo Pascal 6.0. Andi Offset. Yogyakarta
- Sanjaya, Dwi. (2001). Struktur Data di Planet Pascal. J&J Learning. Yogyakarta