

BUKU AJAR

MODUL PEMBELAJARAN PENALARAN MATEMATIKA PADA PEMBUATAN ALGORITMA KOMPUTASI DASAR

Oleh:

Dr. Sulis Janu Hartati, MT.

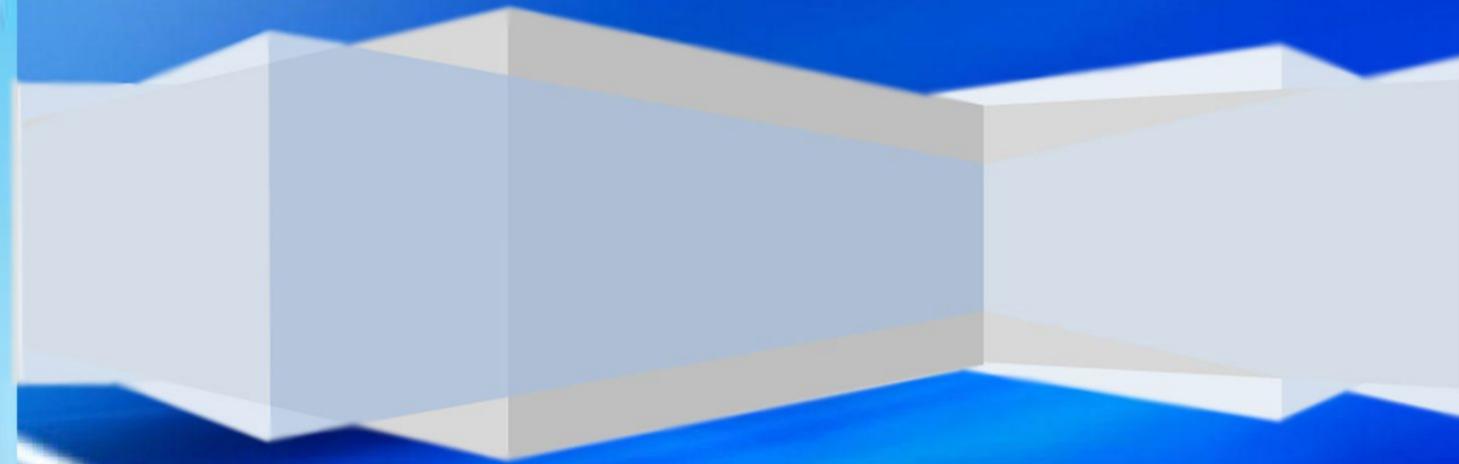
Endra Rahmawati, S.Kom, M.Kom.

Anik Vega Vitianingsih, S.Kom., MT.

Neny Kurniati, S.Kom., MT.

Dr. Nur Sayidah, SE., M.Si., AK.

Dr. Muhajir, M.Ed.



PENERBIT
UNITOMO PRESS
Jl. Semolowaru No. 84
Surabaya, 60283
E - mail : press@unitomo.ac.id
Telp. (031) 5925970
Fax. (031) 5938935



UNITOMO PRESS

BUKU AJAR

MODUL PEMBELAJARAN PENALARAN MATEMATIKA PADA PEMBUATAN ALGORITMA KOMPUTASI DASAR

Oleh:

Dr. Sulis Janu Hartati, MT.

Endra Rahmawati, S.Kom, M.Kom.

Anik Vega Vitianingsih, S.Kom., MT.

Neny Kurniati, S.Kom., MT.

Dr. Nur Sayidah, SE., M.Si., AK.

Dr. Muhajir, M.Ed.

2018

Dibiayai oleh:

Direktorat Riset dan Pengabdian Masyarakat
Direktorat Jendral Penguatan Riset dan Pengembangan
Kementrian Riset, Teknologi, dan Pendidikan Tinggi
sesuai dengan Kontrak Penelitian Tahun Anggaran 2018
Nomor: 120/SP2H/LT/DRPM/2018, Tanggal 30 Januari


UNITOMO PRESS

BUKU AJAR

MODUL PEMBELAJARAN PENALARAN MATEMATIKA PADA PEMBUATAN ALGORITMA KOMPUTASI DASAR

Penulis :

Dr. Sulis Janu Hartati, MT.

Endra Rahmawati, S.Kom, M.Kom.

Anik Vega Vitianingsih, S.Kom., MT.

Neny Kurniati, S.Kom., MT.

Dr. Nur Sayidah, SE., M.Si., AK.

Dr. Muhajir, M.Ed.

© 2018

Diterbitkan Oleh:



Jl. Semolowaru No. 84

Surabaya 60283

Jawa Timur, Indonesia

E-Mail : press@unitomo.ac.id

Telp : (031) 592 5970

Fax : (031) 593 8935

Cetakan pertama, Agustus 2018

Hak Cipta dilindungi undang – undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara apapun tanpa izin tertulis dari penerbit.

PRAKATA

Dengan mengucapkan syukur Alhamdulillahirabbil 'alamin kami panjatkan kehadiran ya Rabbi, Tuhan Yang Maha Rahman dan Rahim, atas segala karuniaNya sehingga Bahan Ajar atau Modul Pembelajaran yang merupakan luaran penelitian pada skim Sosial, Humaniora, dan Pendidikan untuk tahun **Kedua** dapat diselesaikan. Tanpa ijin dan ridloNya sangat mustahil Dokumen ini dapat diselesaikan tepat waktu.

Pada kesempatan ini tak lupa kami sampaikan rasa terimakasih dan penghargaan yang setinggi-tingginya kepada DRPM, Dirjen Penguatan Riset dan Pengembangan, Kemen Ristek Dikti dan Kopertis Wilayah VII yang telah memberikan kesempatan kepada kami. Sungguh dana penelitian ini sangat bermanfaat bagi kami, khususnya dalam hal meningkatkan kemampuan meneliti.

Kami juga haturkan penghargaan yang setinggi-tingginya kepada semua pihak yang telah memberi bantuan dan dukungan sehingga purwarupa ini dapat diselesaikan.

- 1) Rektor Universitas Dr. Soetomo Surabaya, yang telah memberikan kesempatan untuk melakukan penelitian. Tanpa dukungan beliau, mustahil penelitian ini dapat diselesaikan tepat pada waktunya.
- 2) Ketua POLTEK Surabaya, selaku mitra dalam penelitian, yang telah memberikan kepercayaan kepada kami. Tanpa kepercayaan tersebut, penelitian ini tidak bisa terlaksana.
- 3) Ketua Lembaga Penelitian Universitas Dr. Soetomo Surabaya, serta Dekan Fakultas Keguruan dan Ilmu Pendidikan, atas kesempatan dan kepercayaannya pada peneliti.
- 4) Para mahasiswa Program Studi Teknik Informatika angkatan 2016/2017, yang bersedia menjadi subjek penelitian.
- 5) Semua anggota tim penelitian, atas kerjasama dan pengertiannya.

Akhir kata, kami sampaikan rasa terimakasih kepada semua pihak, yang telah membantu dan mendukung kegiatan penelitian ini. Tanpa bantuan dan dukungan dari berbagai pihak, mustahil penelitian ini dapat dilaksanakan.

Surabaya, Juli 2018
Tim Peneliti

DAFTAR ISI

PRAKATA.....	iii
DAFTAR ISI	iv
DAFTAR GAMBAR.....	vi
MODUL 1 KOMPONEN DASAR PENGOLAHAN DATA	1
Sub Kompetensi	1
Tujuan Pembelajaran	1
Waktu Pelaksanaan Perkuliahan	1
Media yang Digunakan	1
Komponen Pengolahan Data.....	1
Rangkuman.....	17
Latihan Soal.....	17
MODUL 2 PERNYATAAN MATEMATIKA DAN LOGIKA MATEMATIKA	18
Sub Kompetensi	18
Tujuan Pembelajaran	18
Waktu Pelaksanaan Perkuliahan	18
Media yang Digunakan	18
Metode Penyajian Algoritma	18
Pernyataan Matematika Untuk Komputasi	25
Logika Matematika Untuk Komputasi	29
Rangkuman.....	31
Latihan Soal.....	32
MODUL 3 VARIABEL DAN DATA.....	33
Sub Kompetensi	33
Tujuan Pembelajaran	33
Waktu Pelaksanaan Perkuliahan	33
Media yang Digunakan	33
Variabel dan Tipe Data	33
Klasifikasi Variabel Berdasar Komponen Pengolah Data.....	37
Rangkuman.....	43
Latihan Soal.....	43
MODUL 4 PERCABANGAN.....	44
Sub Kompetensi	44
Tujuan Pembelajaran	44
Waktu Pelaksanaan Perkuliahan	44
Media yang Digunakan	44
Makna Simbol Flowchart Percabangan.....	44
Klasifikasi Proses Percabangan Pada Pengolahan Data.....	46
Rangkuman.....	48
Latihan Soal.....	48
MODUL 5 PERULANGAN 1	50
Sub Kompetensi	50
Tujuan Pembelajaran	50
Waktu Pelaksanaan Perkuliahan	50
Media yang Digunakan	50
Konsep Dasar Perulangan	50
Penalaran Sigma Dalam Komputasi	54
Rangkuman.....	59
Latihan Soal.....	59

MODUL 6 PERULANGAN2	60
Sub Kompetensi	60
Tujuan Pembelajaran	60
Waktu Pelaksanaan Perkuliahan	60
Media yang Digunakan	60
Penalaran Phi (Multiplikator)	60
Penalaran Faktorial Dalam Komputasi	64
Rangkuman	68
Latihan Soal	68
MODUL 7 MEMBUAT PSEUDOCODE DARI FLOWCHART DAN VA-	
RIABEL ARRAY 1 DIMENSI.....	69
Sub Kompetensi	69
Tujuan Pembelajaran	69
Waktu Pelaksanaan Perkuliahan	69
Media yang Digunakan	69
Aturan Penulisan Pseudocode	69
Variabel Array 1 Dimensi	71
Rangkuman	76
Latihan Soal	77

DAFTAR GAMBAR

Gambar 1. 1 Komponen Dasar Pengolahan Data antara Input, Proses, dan Output	2
Gambar 1. 2 Tampilan Awal Sistem Menu Aplikasi Dasar Algoritma-Komputasi	3
Gambar 1. 3 Form Login	4
Gambar 1. 4 Form Registrasi	5
Gambar 1. 5 Menu Masukan dan Keluaran dari Setiap Form	5
Gambar 1. 6 Komponen Input Form Operasi Penjumlahan	6
Gambar 1. 7 Komponen Output Form Operasi Penjumlahan	6
Gambar 1. 8 Komponen Input Form Operasi Pengurangan	6
Gambar 1. 9 Komponen Output Form Operasi Pengurangan	6
Gambar 1. 10 Komponen Input Form Operasi Perkalian	7
Gambar 1. 11 Komponen Output Form Operasi Perkalian	7
Gambar 1. 12 Komponen Input Menghitung Nilai Akhir	7
Gambar 1. 13 Komponen Output Menghitung Nilai Akhir	8
Gambar 1. 14 Komponen Input Menghitung Rata-rata Dua Variabel	8
Gambar 1. 15 Komponen Output Menghitung Rata-rata Dua Variabel	8
Gambar 1. 16 Komponen Input Menghitung Luas Lingkaran	9
Gambar 1. 17 Komponen Output Menghitung Luas Lingkaran	9
Gambar 1. 18 Komponen Input Menghitung Akar Persamaan Kuadrat	9
Gambar 1. 19 Komponen Output Menghitung Akar Persamaan Kuadrat	10
Gambar 1. 20 Komponen Input Operasi Pembagian	10
Gambar 1. 21 Komponen Output Operasi Pembagian	10
Gambar 1. 22 Komponen Input Membuat Kounter Dasar 1	11
Gambar 1. 23 Komponen Output Membuat Kounter Dasar 1	11
Gambar 1. 24 Komponen Input Membuat Kounter Dasar 2	11
Gambar 1. 25 Komponen Output Membuat Kounter Dasar 2	11
Gambar 1. 26 Komponen Input Akumulasi Penjumlahan	12
Gambar 1. 27 Komponen Output Akumulasi Penjumlahan	12
Gambar 1. 28 Komponen Input Akumulasi Perkalian	12
Gambar 1. 29 Komponen Output Akumulasi Perkalian	12
Gambar 1. 30 Komponen Input Faktorial	13

Gambar 1. 31 Komponen Output Faktorial	13
Gambar 1. 32 Komponen Input Operasi Penjumlahan Matriks	13
Gambar 1. 33 Komponen Output Operasi Penjumlahan Matriks	14
Gambar 1. 34 Komponen Input Operasi Pengurangan Matriks	14
Gambar 1. 35 Komponen Output Operasi Pengurangan Matriks	14
Gambar 1. 36 Komponen Input Logika Dasar	15
Gambar 1. 37 Komponen Output Logika Dasar	15
Gambar 1. 38 Komponen Input Kombinasi Dua Relasi.....	15
Gambar 1. 39 Komponen Output Kombinasi Dua Relasi	15
Gambar 1. 40 Komponen Input Konsep Perulangan.....	16
Gambar 1. 41 Komponen Output Konsep Perulangan	16
Gambar 1. 42 Komponen Input Konsep Perulangan Array.....	16
Gambar 1. 43 Komponen Output Konsep Perulangan Array.....	16
Gambar 2. 1 Contoh Flowchart dan Pseudocode Untuk Hitung Gaji Karyawan.....	19
Gambar 2. 2 Simbol Flowchart untuk Pembuatan Algoritma	20
Gambar 2. 3 Struktur Penulisan Pseudocode.....	20
Gambar 2. 4 Toolbar Menu Flowchart dan Pseudocode dari Form Pernyataan Matematika.....	21
Gambar 2. 5 Form Operasi Penjumlahan.....	21
Gambar 2. 6 Kebutuhan Variabel pada Form Operasi Penjumlahan.....	21
Gambar 2. 7 Flowchart dari Form Operasi Penjumlahan	22
Gambar 2. 8 Pseudocode dari Form Operasi Penjumlahan	22
Gambar 2. 9 Form Operasi Pengurangan	23
Gambar 2. 10 Kebutuhan Variabel pada Form Operasi Pengurangan	23
Gambar 2. 11 Flowchart dari Form Operasi Pengurangan.....	23
Gambar 2. 12 Pseudocode dari Form Operasi Pengurangan	24
Gambar 2. 13 Form Operasi Perkalian	24
Gambar 2. 14 Kebutuhan Variabel pada Form Operasi Perkalian	24
Gambar 2. 15 Flowchart dari Form Operasi Perkalian.....	25
Gambar 2. 16 Pseudocode dari Form Operasi Perkalian.....	25
Gambar 2. 17 Persamaan yang mewakili Penjumlahan, Pengurangan, dan Perkalian.....	26
Gambar 2. 18 Menu Aritmetika Dasar	26

Gambar 2. 19 Bentuk pernyataan matematika pada Form Operasi Penjumlahan	27
Gambar 2. 20 Bentuk pernyataan matematika pada Form Operasi Pengurangan.....	28
Gambar 2. 21 Bentuk pernyataan matematika pada Form Operasi Perkalian	28
Gambar 2. 22 Contoh Logika Matematika.....	29
Gambar 2. 23 Menu Logika Matematika.....	29
Gambar 2. 24 Bentuk logika matematika pada Form Logika Dasar	30
Gambar 2. 25 Bentuk Logika Matematika pada Form Kombinasi Dua Variabel	31
Gambar 3. 1 Perbedaan Operator dan Operand	34
Gambar 3. 2 Aplikasi Dasar-dasar Algoritma Komputasi bagian Menu Menghitung Nilai Akhir.....	37
Gambar 3. 3 Sub Menu dari Menu Aplikasi Aritmetika	38
Gambar 3. 4 Pseudocode dan Flowchart Perhitungan Nilai Akhir.....	39
Gambar 3. 5 Kode Program Perhitungan Nilai Akhir.....	40
Gambar 3. 6 Aplikasi Dasar-dasar Algoritma Komputasi bagian Menu Menghitung Rata-rata Dua Variabel.....	40
Gambar 3. 7 Pseudocode dan Flowchart Perhitungan Rata-rata Dua Buah Variabel.....	42
Gambar 4. 1 Simbol Flowchart Proses Percabangan	45
Gambar 4. 2 Flowchart Proses Percabangan Pembelian Baju	45
Gambar 4. 3 Flowchart Proses Percabangan Nilai Huruf	46
Gambar 4. 4 Flowchart Proses Percabangan Pembagian Dua Buah Bilangan	48
Gambar 5. 1 Flowchart Dasar Perulangan For	51
Gambar 5. 2 Model Penalaran Pembuatan Kounter Dasar 1	52
Gambar 5. 3 Tabel Kebenaran Pembuatan Kounter Dasar 1	53
Gambar 5. 4 Flowchart Perulangan Pembuatan Kounter Dasar 1	53
Gambar 5. 5 Model Penalaran Pembuatan Akumulasi Penjumlahan	56
Gambar 5. 6 Tabel Kebenaran Pembuatan Akumulasi Penjumlahan.....	56
Gambar 5. 7 Flowchart Perulangan Pembuatan Akumulasi Penjumlahan.....	57
Gambar 5. 8 Flowchart Perulangan Pembuatan Akumulasi Penjumlahan.....	57

han(Lanjutan).....	58
Gambar 6. 1 Model Penalaran PembuatanAkumulasiPerkalian	63
Gambar 6. 2 Tabel Kebenaran Pembuatan AkumulasiPerkalian	63
Gambar 6. 3 Flowchart Perulangan PembuatanAkumulasiPerkalian	64
Gambar 6. 4 Aplikasi Dasar Algoritma –MenghitungFaktorial	65
Gambar 6. 6 Aplikasi Dasar Algoritma – MenghitungFaktorial (Lanjutan).....	67
Gambar 6. 7 Model PenalaranPerhitunganFaktorial	66
Gambar 6. 8 Tabel KebenaranPerhitunganFaktorial.....	67
Gambar 6. 9 Flowchart PerulanganPerhitunganFaktorial	67
Gambar 7. 1 StrukturPenulisanPseudocode.....	70
Gambar 7. 2 Penulisan Pseudocode dengan Notasi Algoritmik pada Proses Sekuensial untuk Proses Penjumlahan dan Pengu- rangan 2buahbilangan	71
Gambar 7. 3 Aplikasi Dasar Algoritma – Menghitung PenjumlahanMa- triksBaris.....	73
Gambar 7. 4 Model Penalaran Perhitungan PenjumlahanMatriksBaris.....	74
Gambar 7. 5 Aplikasi Dasar Algoritma – Perhitungan PenguranganMa- triksBaris	74
Gambar 7. 6 Kebutuhan Variabel Untuk PenguranganMatriksBaris.....	75
Gambar 7. 7 Model Penalaran PenguranganMatriks Baris.....	76

MODUL 1

KOMPONEN DASAR PENGOLAHAN DATA

Sub Kompetensi

- Mahasiswa dapat menjelaskan konsep dasar pengolahan data dengan bantuan stimulus dari luar berupa media pembelajaran penalaran matematika.

Tujuan Pembelajaran

- Mampu memahami Konsep Dasar Pengolahan Data.
- Mampu menjelaskan perbedaan Komponen Dasar Pengolahan Data.

Waktu Pelaksanaan Perkuliahan

- Pertemuan Minggu ke-1.

Media yang Digunakan

- Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Sistem Menu dan Modul Login.

Komponen Pengolahan Data

Data merupakan sekumpulan fakta yang dapat diolah sesuai dengan kebutuhan pengguna. Pengolahan Data merupakan proses manipulasi data ke dalam bentuk yang lebih berarti/bermakna sehingga dapat menghasilkan sebuah informasi. Seperti halnya komputer, yang mempunyai sejumlah proses pengolahan data pendukung agar dapat menghasilkan keluaran dengan tepat dan benar, pada Algoritma juga mempunyai 3 komponen utama sebagai penentu keberhasilan pengolahan data yaitu komponen input, proses, dan output. Hubungan Komponen Pengolahan Data dapat dilihat pada Gambar 1.1.



Gambar 1. 1 Komponen Dasar Pengolahan Data antara Input, Proses, dan Output.

Keterangan :

- Input : Data inputan (masukan), baik berupa deklarasi variabel dan tipe data yang akan digunakan selama proses berjalan, maupun variabel yang nilainya berasal dari inputan user (pengguna).
- Proses : Tahap pemrosesan data, baik menggunakan logika proses maupun model matematika yang sesuai dengan kebutuhan pengguna.
- Output : Data output (keluaran) yang dihasilkan berdasarkan inputan dan proses yang terjadi dalam sebuah pengolahan data dan algoritma.

Sebuah proses komputasi dapat dikatakan berjalan dengan baik jika dapat menghasilkan (beberapa) nilai output dari (beberapa) nilai input yang diberikan oleh pengguna. Dengan adanya 3 komponen pengolahan data dan algoritma tersebut, pembuatan algoritma harus melalui 3 tahap yang harus dilakukan, yaitu sebagai berikut :

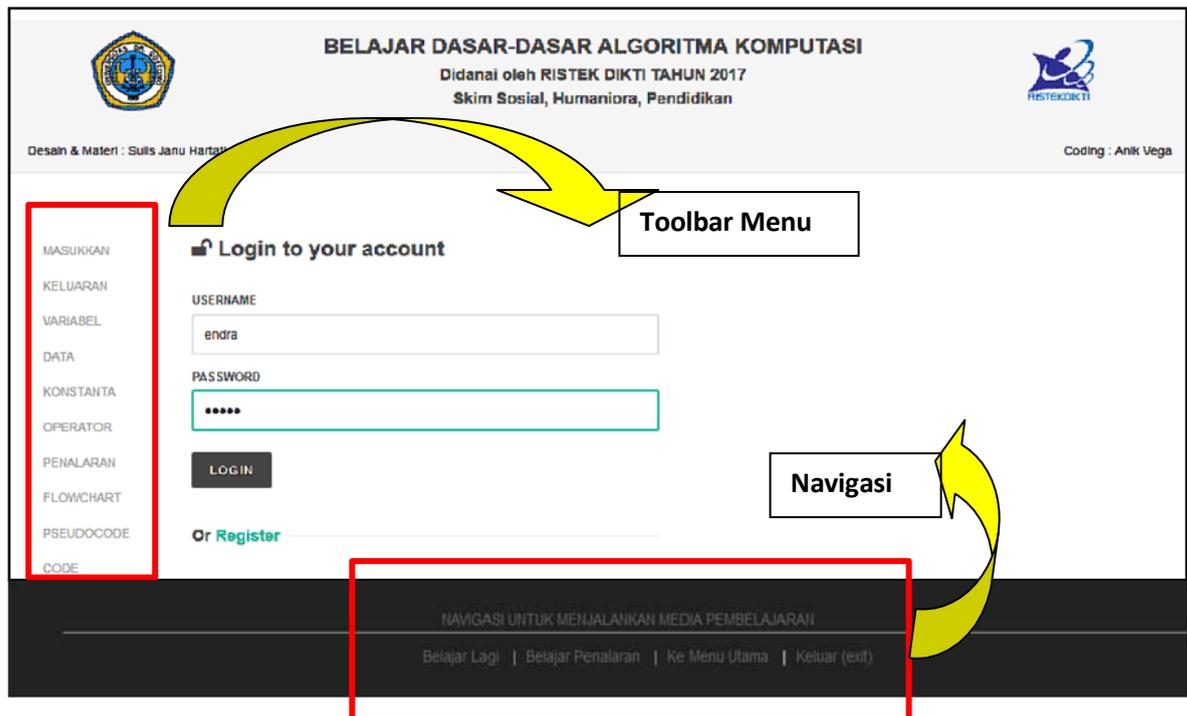
- a. Identifikasi semua variabel, tipe data, dan konstanta yang dibutuhkan sebagai data input.
- b. Tentukan proses yang tepat sebagai solusi untuk menyelesaikan permasalahan.
- c. Tampilkan semua hasil akhir pengolahan data dan algoritma.

Logika dapat diartikan sebagai penalaran atau sesuatu yang masuk akal. Penalaran adalah salah satu bentuk pemikiran. Berdasarkan arti kata tersebut, logika dapat diartikan sebagai ilmu yang menggunakan prinsip-prinsip berlogika agar dapat berfikir secara epat dan benar menurut kaidah/aturan tertentu. Cara berfikir logis dan sistematis inilah yang menjadi dasar penting pembuatan algoritma. Oleh karena itu, Algoritma dapat didefinisikan sebagai urutan instruksi spesifik secara bertahap yang harus dilakukan untuk menyelesaikan masalah.

Contoh Implementasi Konsep Pengolahan Data dapat dilihat pada Aplikasi Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Sistem Menu dan Form Login.

- **Pada Tampilan Utama Aplikasi (Sistem Menu)**

Menu aplikasi didesain dengan mempertimbangkan pedoman interaksi manusia dan komputer. Beberapa hal diperhatikan adalah: warna latar pada monitor, warna dan font huruf, konsistensi penempatan navigasi dan tool bar, serta konsistensi penempatan identitas aplikasi, dan area belajar. Lihat Gambar 1.2.



Gambar 1. 2 Tampilan Awal Sistem Menu Aplikasi Dasar Algoritma Komputasi.

Beberapa Toolbar Menu Pilihan diletakkan pada bagian kiri aplikasi dan fungsinya adalah sebagai berikut :

- 1) Tool bar masukan berfungsi untuk belajar model masukan.
- 2) Tool bar keluaran berfungsi untuk belajar model keluaran
- 3) Tool bar variabel berfungsi untuk belajar tentang variable yang ada pada model masukan dan keluaran, serta proses.
- 4) Tool bar data berfungsi untuk belajar tentang data yang ada pada model masukan dan keluaran, serta proses.
- 5) Tool bar konstanta berfungsi untuk belajar tentang konstanta yang ada pada model masukan dan keluaran, serta proses.

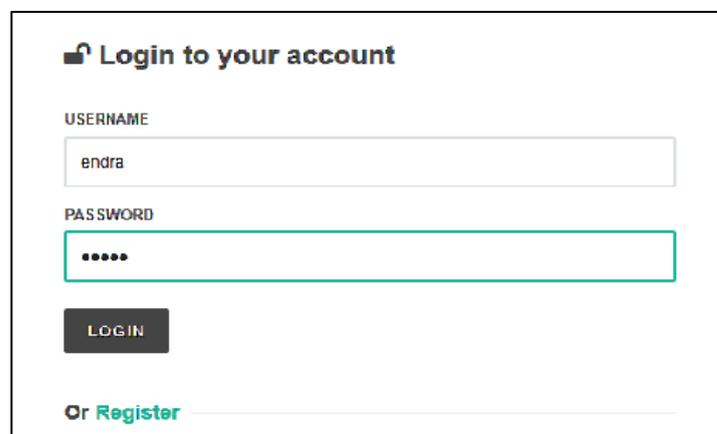
- 6) Tool bar operator berfungsi untuk belajar tentang operator yang ada pada komponen proses.
- 7) Tool bar penalaran berfungsi untuk belajar tentang penalaran yang ada pada proses transformasi masukan menjadi keluaran.
- 8) Tool bar flowchart berfungsi untuk belajar tentang flowchart yang mentransformasikan masukan menjadi keluaran.
- 9) Tool bar pseudo code berfungsi untuk belajar tentang pseudo code yang mentransformasikan masukan menjadi keluaran.

Sedangkan navigasi untuk menjalankan aplikasi tersedia empat pilihan, yaitu: belajar lagi, belajar penalaran, kembali ke menu utama, serta exit. Fungsi masing-masing navigasi adalah sebagai berikut.

- 1) Pilihan belajar lagi, berfungsi untuk mengarahkan pengguna jika menginginkan belajar lagi untuk materi pada tool bar.
- 2) Pilihan belajar penalaran, berfungsi untuk mengarahkan pengguna jika menginginkan belajar penalaran pada proses transformasi.
- 3) Pilihan kembali ke menu utama, berfungsi untuk mengarahkan pengguna untuk kembali ke menu utama.
- 4) Pilihan exit, berfungsi untuk mengarahkan pengguna keluar dari aplikasi.

- **Pada Form Login**

Pengguna diharuskan memasukkan data username dan password terlebih dahulu pada teksfield yang telah disediakan. Kemudian tekan tombol Login. Apabila belum terdaftar sebagai anggota, sebaiknya melakukan Register terlebih dahulu. Lihat Gambar 1.3 dan 1.4.



The image shows a login form with the following elements:

- Title: **Login to your account**
- Label: **USERNAME**
- Input field: Contains the text "endra".
- Label: **PASSWORD**
- Input field: Contains five dots (masked password).
- Button: **LOGIN**
- Link: **Or Register**

Gambar 1. 3 Form Login

Register a new account

NAME *

USERNAME *

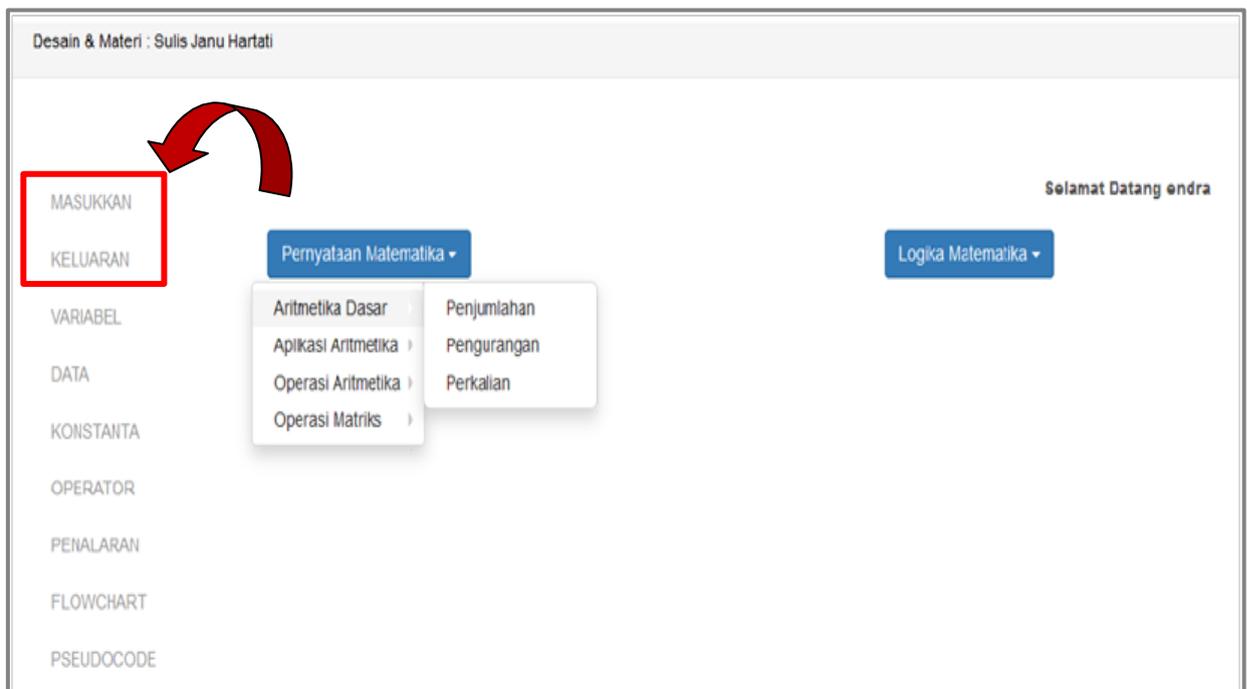
PASSWORD *

REGISTER NOW

Gambar 1. 4 Form Registrasi

Jika Anda belum terdaftar, isikan data nama, username, dan password, kemudian tekan tombol Register Now. Setelah register, pengguna dapat menginputkan data username dan password dengan benar, sehingga dapat masuk dan mengakses seluruh sistem menu pada aplikasi dasar logika komputasi.

Komponen input dan output pada Menu Form Aplikasi Dasar-Dasar Algoritma Komputasi dapat dilakukan dengan cara memilih Menu Pernyataan Matematika serta Logika Matematika, dan **tekan toolbar menu pada sebelah kiri bernama Masukan dan Keluaran.**



Gambar 1. 5 Menu Masukan dan Keluaran dari Setiap Form.

- **Menu Pernyataan Matematika**

1. **Pilih dan klik Menu Aritmetika Dasar – Penjumlahan**

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



The screenshot shows a web form titled "Operasi Penjumlahan". It contains two input fields labeled "BILANGAN KE-1" and "BILANGAN KE-2". Below these fields is a blue button labeled "Tampilkan Hasil". A red rectangular box highlights the two input fields.

Gambar 1. 6 Komponen Input Form Operasi Penjumlahan.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.



The screenshot shows the same "Operasi Penjumlahan" form. The input fields are now empty. A red rectangular box highlights the blue "Tampilkan Hasil" button and the area below it, which is the output area.

Gambar 1. 7 Komponen Output Form Operasi Penjumlahan.

2. **Pilih dan klik Menu Aritmetika Dasar – Pengurangan**

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input



The screenshot shows a web form titled "Operasi Pengurangan". It contains two input fields labeled "BILANGAN KE-1" and "BILANGAN KE-2". Below these fields is a blue button labeled "Tampilkan Hasil". A red rectangular box highlights the two input fields.

Gambar 1. 8 Komponen Input Form Operasi Pengurangan.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output



The screenshot shows the same "Operasi Pengurangan" form. The input fields are now empty. A red rectangular box highlights the blue "Tampilkan Hasil" button and the area below it, which is the output area.

Gambar 1. 9 Komponen Output Form Operasi Pengurangan.

3. Pilih dan klik Menu Aritmetika Dasar – Perkalian

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



The screenshot shows a web form titled "Operasi Perkalian". It contains two input fields labeled "BILANGAN KE-1" and "BILANGAN KE-2". Below these fields is a blue button labeled "Tampilkan Hasil". A red rectangular box highlights the two input fields.

Gambar 1. 10 Komponen Input Form Operasi Perkalian.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

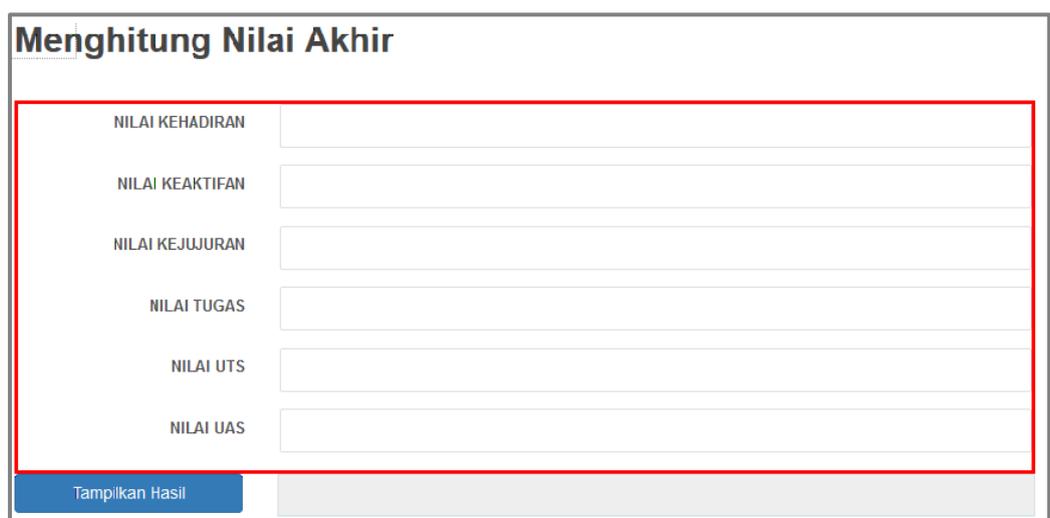


The screenshot shows the same "Operasi Perkalian" form. The "Tampilkan Hasil" button is now highlighted with a red rectangular box, indicating it has been clicked to display the output.

Gambar 1. 11 Komponen Output Form Operasi Perkalian.

4. Pilih dan klik Menu Aplikasi Aritmetika – Menghitung Nilai Akhir

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



The screenshot shows a web form titled "Menghitung Nilai Akhir". It contains six input fields labeled "NILAI KEHADIRAN", "NILAI KEAKTIFAN", "NILAI KEJUJURAN", "NILAI TUGAS", "NILAI UTS", and "NILAI UAS". Below these fields is a blue button labeled "Tampilkan Hasil". A red rectangular box highlights all six input fields.

Gambar 1. 12 Komponen Input Menghitung Nilai Akhir.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

The screenshot shows a web application window titled "Menghitung Nilai Akhir". It contains six input fields for the following categories: NILAI KEHADIRAN, NILAI KEAKTIFAN, NILAI KEJUJURAN, NILAI TUGAS, NILAI UTS, and NILAI UAS. At the bottom of the window, there is a blue button labeled "Tampilkan Hasil" and a grey area to its right. A red rectangular box highlights the bottom portion of the window, including the button and the grey area.

Gambar 1. 13 Komponen Output Menghitung Nilai Akhir.

5. Pilih dan klik Menu Aplikasi Aritmetika – Menghitung Rata – rata Dua Variabel.

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.

The screenshot shows a web application window titled "Rata-Rata Dua Variabel". It contains two input fields labeled "BILANGAN KE-1" and "BILANGAN KE-2". At the bottom, there is a blue button labeled "Tampilkan Hasil" and a grey area to its right. A red rectangular box highlights the input fields and the button area.

Gambar 1. 14 Komponen Input Menghitung Rata-rata Dua Variabel.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

The screenshot shows a web application window titled "Rata-Rata Dua Variabel". It contains two input fields labeled "BILANGAN KE-1" and "BILANGAN KE-2". At the bottom, there is a blue button labeled "Tampilkan Hasil" and a grey area to its right. A red rectangular box highlights the bottom portion of the window, including the button and the grey area.

Gambar 1. 15 Komponen Output Menghitung Rata-rata Dua Variabel.

6. Pilih dan klik Menu Aplikasi Aritmetika – Menghitung Luas Lingkaran

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



Gambar 1. 16 Komponen Input Menghitung Luas Lingkaran.

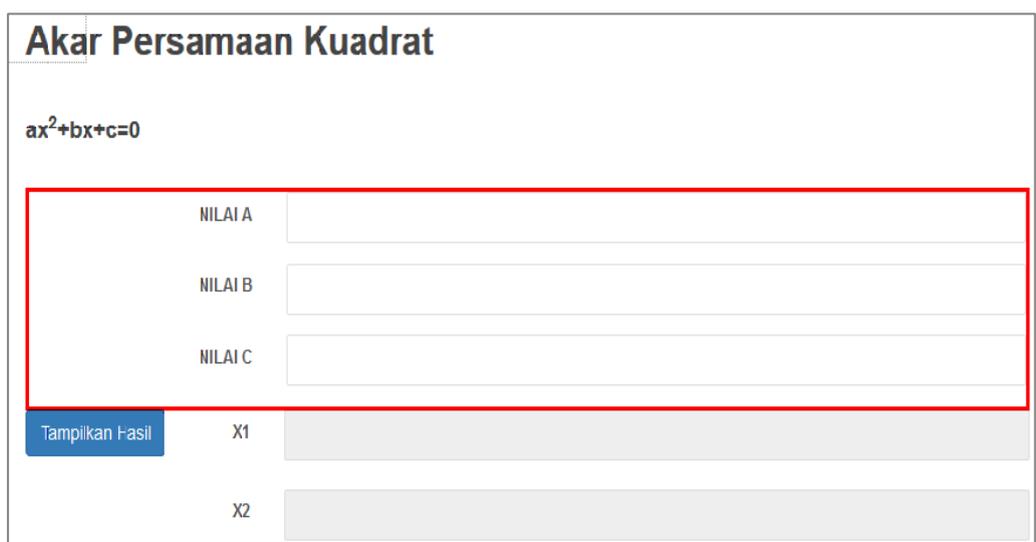
- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.



Gambar 1. 17 Komponen Output Menghitung Luas Lingkaran.

7. Pilih dan klik Menu Aplikasi Aritmetika – Menghitung Akar Persamaan Kuadrat.

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



Gambar 1. 18 Komponen Input Menghitung Akar Persamaan Kuadrat.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

Akar Persamaan Kuadrat

$ax^2+bx+c=0$

NILAI A

NILAI B

NILAI C

Tampilkan Hasil X1

X2

Gambar 1. 19 Komponen Output Menghitung Akar Persamaan Kuadrat.

8. Pilih dan klik Menu Operasi Aritmetika – Operasi Pembagian.

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.

Operasi Pembagian

BILANGAN KE-1

BILANGAN KE-2

Tampilkan Hasil

Gambar 1. 20 Komponen Input Operasi Pembagian.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

Operasi Pembagian

BILANGAN KE-1

BILANGAN KE-2

Tampilkan Hasil

Gambar 1. 21 Komponen Output Operasi Pembagian.

9. Pilih dan klik Menu Operasi Aritmetika – Membuat Kounter – Kounter Dasar 1.

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



The screenshot shows a window titled "Membuat Kounter Dasar 1". Inside, there is a single input field highlighted with a red border. To the right of the input field is a blue button labeled "Tampilkan Hasil". Below the input field is a grey rectangular area.

Gambar 1. 22 Komponen Input Membuat Kounter Dasar 1.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

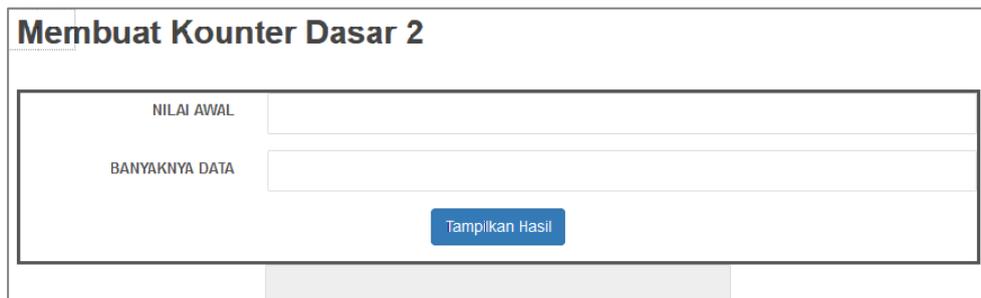


The screenshot shows a window titled "Membuat Kounter Dasar 1". Inside, there is a single output field highlighted with a red border. Above the output field is a blue button labeled "Tampilkan Hasil". To the left of the output field is a grey rectangular area.

Gambar 1. 23 Komponen Output Membuat Kounter Dasar 1.

10. Pilih dan klik menu Operasi Aritmetika – Membuat Kounter – Kounter Dasar 2.

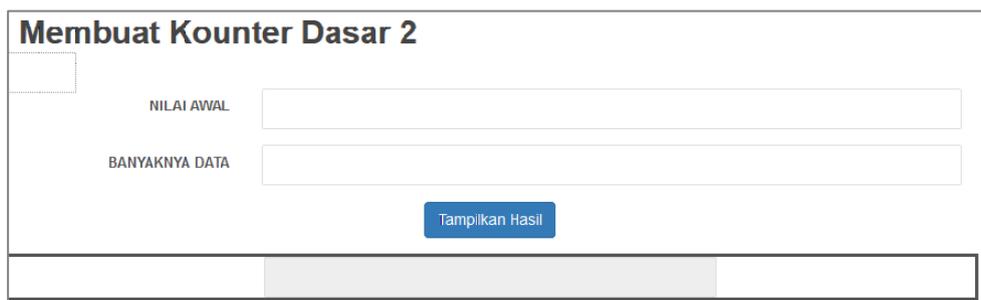
- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



The screenshot shows a window titled "Membuat Kounter Dasar 2". Inside, there are two input fields. The first is labeled "NILAI AWAL" and the second is labeled "BANYAKNYA DATA". Below the input fields is a blue button labeled "Tampilkan Hasil". Below the button is a grey rectangular area.

Gambar 1. 24 Komponen Input Membuat Kounter Dasar 2.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

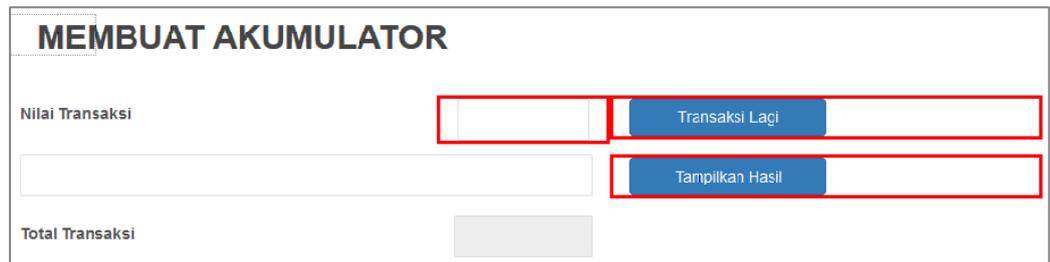


The screenshot shows a window titled "Membuat Kounter Dasar 2". Inside, there are two output fields. The first is labeled "NILAI AWAL" and the second is labeled "BANYAKNYA DATA". Above the output fields is a blue button labeled "Tampilkan Hasil". To the left of the output fields is a grey rectangular area.

Gambar 1. 25 Komponen Output Membuat Kounter Dasar 2.

11. Pilih dan klik Menu Operasi Aritmetika – Akumulasi Penjumlahan

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



The screenshot shows a web interface titled "MEMBUAT AKUMULATOR". It features a form with the following elements: a label "Nilai Transaksi" followed by an input field, a blue button labeled "Transaksi Lagi", a second input field, a blue button labeled "Tampilkan Hasil", and a label "Total Transaksi" followed by a greyed-out output field. Red rectangular boxes highlight the first input field, the "Transaksi Lagi" button, the second input field, and the "Tampilkan Hasil" button.

Gambar 1. 26 Komponen Input Akumulasi Penjumlahan.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

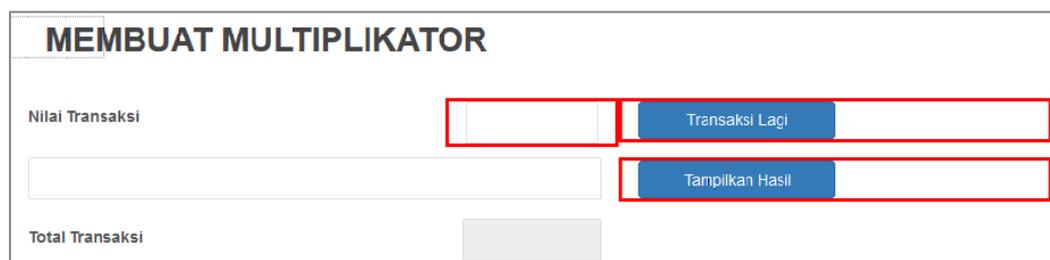


The screenshot shows the same "MEMBUAT AKUMULATOR" interface. In this view, the input fields and buttons are not highlighted. Instead, the second input field and the "Total Transaksi" output field are highlighted with red rectangular boxes, indicating they are the active output components.

Gambar 1. 27 Komponen Output Akumulasi Penjumlahan.

12. Pilih dan klik Menu Operasi Aritmetika – Akumulasi Perkalian

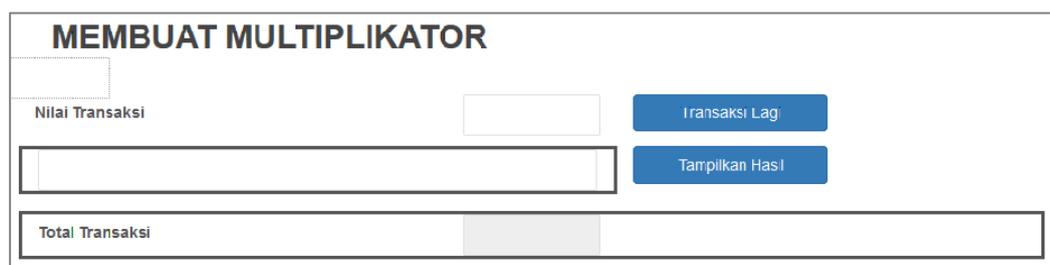
- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



The screenshot shows a web interface titled "MEMBUAT MULTIPLIKATOR". It features a form with the following elements: a label "Nilai Transaksi" followed by an input field, a blue button labeled "Transaksi Lagi", a second input field, a blue button labeled "Tampilkan Hasil", and a label "Total Transaksi" followed by a greyed-out output field. Red rectangular boxes highlight the first input field, the "Transaksi Lagi" button, the second input field, and the "Tampilkan Hasil" button.

Gambar 1. 28 Komponen Input Akumulasi Perkalian.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

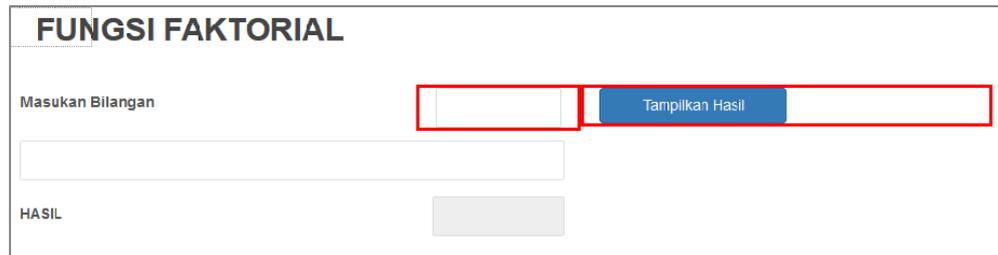


The screenshot shows the same "MEMBUAT MULTIPLIKATOR" interface. In this view, the input fields and buttons are not highlighted. Instead, the second input field and the "Total Transaksi" output field are highlighted with red rectangular boxes, indicating they are the active output components.

Gambar 1. 29 Komponen Output Akumulasi Perkalian.

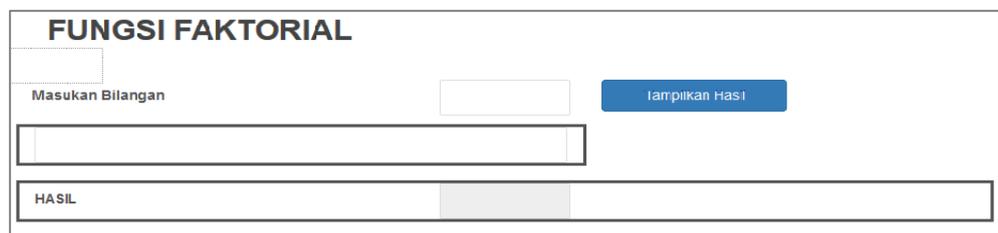
13. Pilih dan klik Menu Operasi Aritmetika – Faktorial

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



Gambar 1. 30 Komponen Input Faktorial.

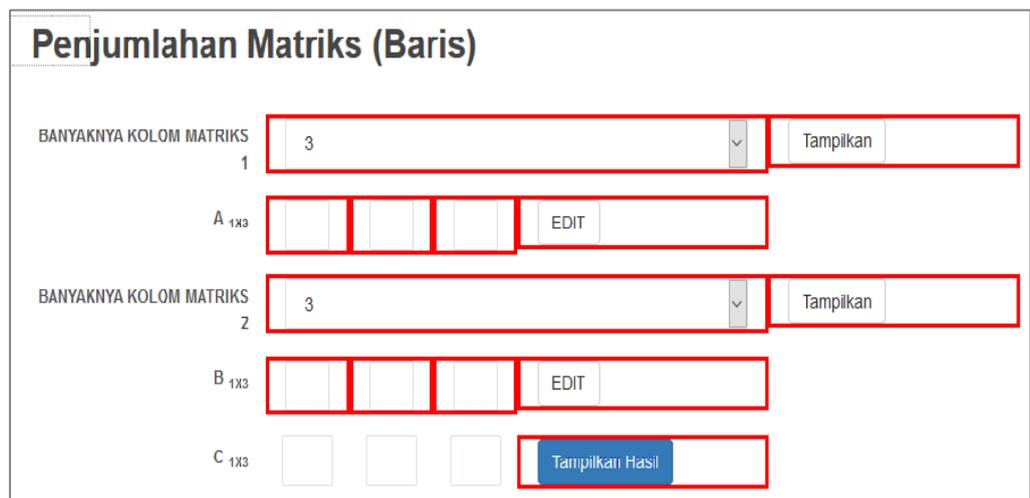
- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.



Gambar 1. 31 Komponen Output Faktorial.

14. Pilih dan klik Menu Operasi Matriks – Operasi Penjumlahan

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



Gambar 1. 32 Komponen Input Operasi Penjumlahan Matriks.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

The screenshot shows the 'Penjumlahan Matriks (Baris)' interface. It features two matrix input sections. The first section is for matrix A, with 'BANYAKNYA KOLOM MATRIKS 1' set to 3 and a 'Tampilkan' button. Below it is a 1x3 grid of input boxes for matrix A and an 'EDIT' button. The second section is for matrix B, with 'BANYAKNYA KOLOM MATRIKS 2' set to 3 and a 'Tampilkan' button. Below it is a 1x3 grid of input boxes for matrix B and an 'EDIT' button. At the bottom, there is a 1x3 grid of input boxes for matrix C and a blue 'Tampilkan Hasil' button.

Gambar 1. 33 Komponen Output Operasi Penjumlahan Matriks.

15. Pilih dan klik Menu Operasi Matriks – Operasi Pengurangan

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.

The screenshot shows the 'Pengurangan Matriks (Baris)' interface. It features two matrix input sections. The first section is for matrix A, with 'BANYAKNYA KOLOM MATRIKS 1' set to 3 and a 'Tampilkan' button. Below it is a 1x3 grid of input boxes for matrix A and an 'EDIT' button. The second section is for matrix B, with 'BANYAKNYA KOLOM MATRIKS 2' set to 3 and a 'Tampilkan' button. Below it is a 1x3 grid of input boxes for matrix B and an 'EDIT' button. At the bottom, there is a 1x3 grid of input boxes for matrix C and a blue 'Tampilkan Hasil' button. Red boxes highlight the input fields and buttons in each section.

Gambar 1. 34 Komponen Input Operasi Pengurangan Matriks.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

The screenshot shows the 'Pengurangan Matriks (Baris)' interface. It features two matrix input sections. The first section is for matrix A, with 'BANYAKNYA KOLOM MATRIKS 1' set to 3 and a 'Tampilkan' button. Below it is a 1x3 grid of input boxes for matrix A and an 'EDIT' button. The second section is for matrix B, with 'BANYAKNYA KOLOM MATRIKS 2' set to 3 and a 'Tampilkan' button. Below it is a 1x3 grid of input boxes for matrix B and an 'EDIT' button. At the bottom, there is a 1x3 grid of input boxes for matrix C and a blue 'Tampilkan Hasil' button. A red box highlights the 1x3 grid of input boxes for matrix C.

Gambar 1. 35 Komponen Output Operasi Pengurangan Matriks.

- **Menu Logika Matematika**

1. **Pilih dan Klik Menu Logika Dasar**

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.

LOGIKA DASAR

X1 ADALAH SIMBOL PERNYATAAN 1: $a = 0$
X2 ADALAH SIMBOL PERNYATAAN 2: $a > 0$

MASUKKAN NILAI A

X1	X2	NOT X1	NOT X2	X1 and X2	X1 or X2
FALSE	TRUE	TRUE	FALSE	FALSE	TRUE

Gambar 1. 36 Komponen Input Logika Dasar.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

LOGIKA DASAR

X1 ADALAH SIMBOL PERNYATAAN 1: $a = 0$
X2 ADALAH SIMBOL PERNYATAAN 2: $a > 0$

MASUKKAN NILAI A

X1	X2	NOT X1	NOT X2	X1 and X2	X1 or X2
FALSE	TRUE	TRUE	FALSE	FALSE	TRUE

Gambar 1. 37 Komponen Output Logika Dasar.

2. **Pilih dan klik Menu Kombinasi Dua Relasi**

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.

KOMBINASI DUA BUAH RELASI

MASUKKAN TIGA BUAH BILANGAN

NILAI TERKECIL ADALAH

Gambar 1. 38 Komponen Input Kombinasi Dua Relasi.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.

KOMBINASI DUA BUAH RELASI

MASUKKAN TIGA BUAH BILANGAN

NILAI TERKECIL ADALAH

Gambar 1. 39 Komponen Output Kombinasi Dua Relasi.

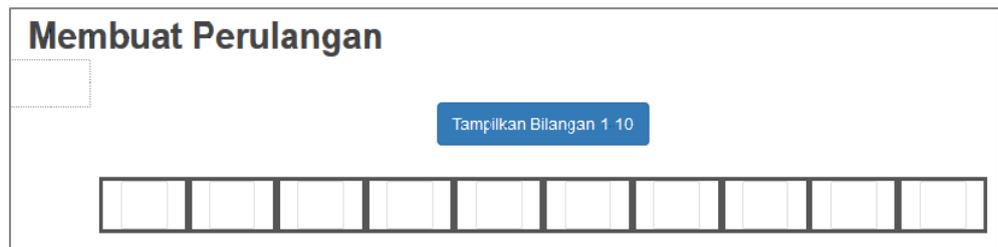
3. Pilih dan klik Menu Konsep Perulangan

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



Gambar 1. 40 Komponen Input Konsep Perulangan.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.



Gambar 1. 41 Komponen Output Konsep Perulangan.

4. Pilih dan klik Menu Konsep Perulangan Array

- a. Klik Toolbar Menu Masukan untuk menampilkan Komponen Input.



Gambar 1. 42 Komponen Input Konsep Perulangan Array.

- b. Klik Toolbar Menu Keluaran untuk menampilkan Komponen Output.



Gambar 1. 43 Komponen Output Konsep Perulangan Array.

Rangkuman

- ❖ Algoritma adalah urutan instruksi spesifik secara bertahap yang harus dilakukan dalam menyelesaikan masalah.
- ❖ Komponen Pengolahan Data terdiri dari 3 elemen utama, yaitu :
 - Input
 - Proses
 - Output

Latihan Soal

1. Buat Rincian Data Input dan Output yang dibutuhkan untuk mendesain sebuah Form Pendaftaran Mahasiswa Baru. Anda dapat menggunakan contoh Formulir Pendaftaran Mahasiswa Baru di kampus Anda sebagai bukti pendukung nyata.
2. Buat Rincian Data Input dan Output yang dibutuhkan untuk mendesain sebuah Form Pembayaran Listrik. Anda dapat menggunakan contoh Bukti Pembayaran Listrik di Rumah Anda sebagai bukti pendukung nyata.

MODUL 2

PERNYATAAN MATEMATIKA DAN LOGIKA MATEMATIKA

Sub Kompetensi

- Mahasiswa dapat menjelaskan konsep dasar pengolahan data dengan bantuan stimulus dari luar berupa media pembelajaran penalaran matematika.

Tujuan Pembelajaran

- Mahasiswa dapat memaknai algoritma yang disajikan dalam bentuk flowchart.
- Mahasiswa dapat memaknai algoritma yang disajikan dalam bentuk pseudocode.
- Mampu menyelesaikan pernyataan matematika untuk komputasi.
- Mampu menggunakan logika matematika untuk komputasi.

Waktu Pelaksanaan Perkuliahan

- Pertemuan Minggu ke-1.

Media yang Digunakan

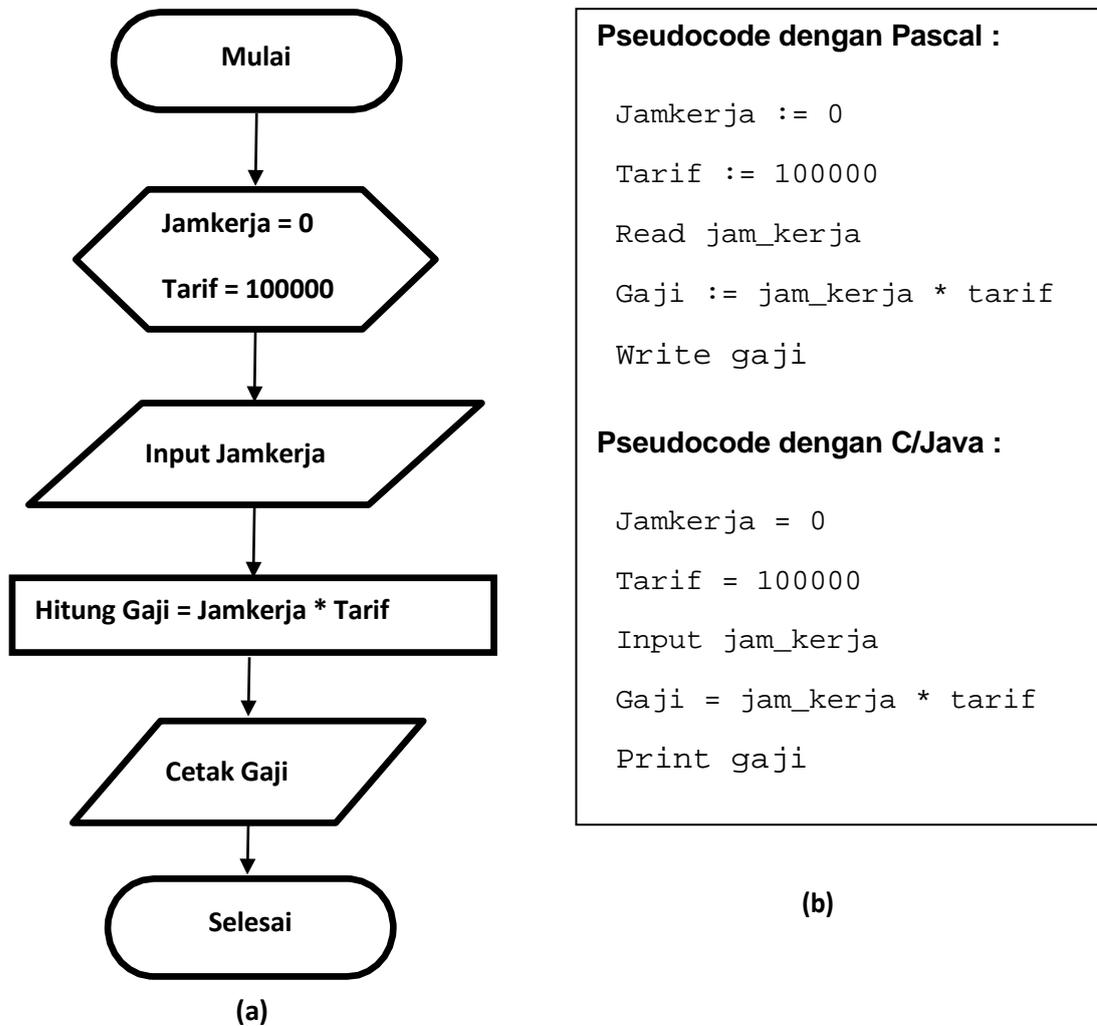
- Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika - Aritmatika Dasar (Penjumlahan, Pengurangan, Perkalian).

Metode Penyajian Algoritma

Sebuah algoritma dapat disajikan menggunakan dua teknik yaitu Teknik Tulisan dan Teknik Gambar. Teknik tulisan biasanya menggunakan *pseudocode*, sedangkan teknik gambar biasanya menggunakan diagram alir (*flowchart*).

Flowchart didefinisikan sebagai suatu bagan terurut yang menggambarkan alur terjadinya suatu proses, dengan menggunakan simbol – simbol tertentu yang telah disepakati sesuai dengan aturan komputasi. Sedangkan pseudocode didefinisikan sebagai bentuk penyajian algoritma dengan menggunakan struktur bahasa tertentu, seperti C dan Pascal.

Pseudocode berasal dari kata *pseudo* yang berarti imitasi/mirip/menyerupai. Sedangkan *code* berarti kode program. Dari asal kata tersebut, dapat dikatakan bahwa pseudocode menggunakan struktur penulisan yang mirip dengan bahasa pemrograman tertentu. Contoh Flowchart dan Pseudocode dapat dilihat pada Gambar 2.1.

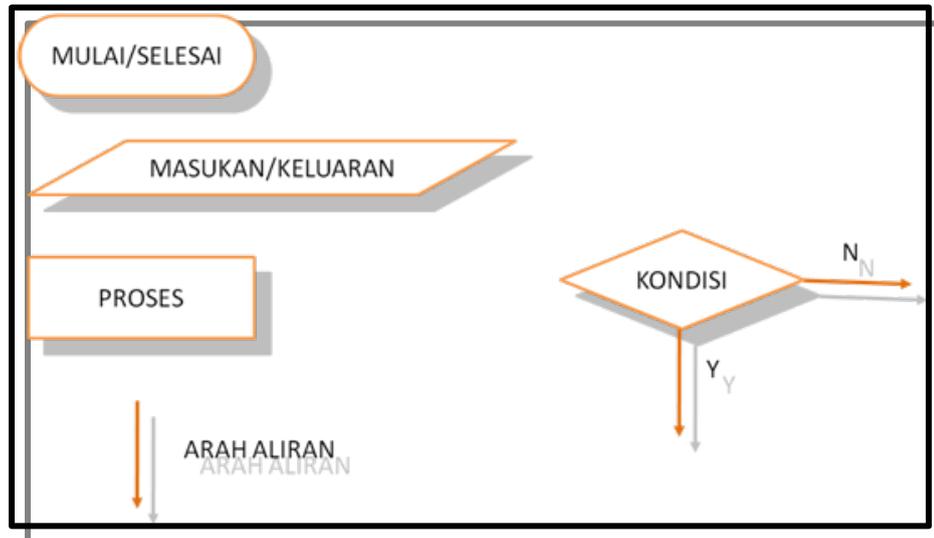


Gambar 2. 1 Contoh Flowchart dan Pseudocode Untuk Hitung Gaji Karyawan.

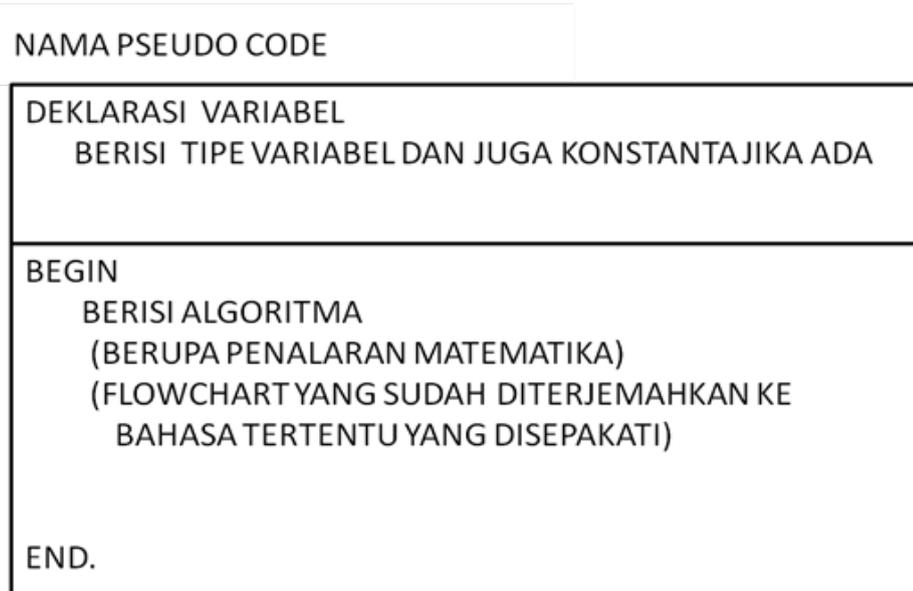
Dengan adanya flowchart dan pseudocode tersebut dalam menyajikan algoritma, ada beberapa manfaat yang dapat diperoleh, yaitu sebagai berikut :

- a. Memudahkan penelusuran alur proses bisnis tertentu.
- b. Mempercepat proses pencarian lokasi kesalahan yang terjadi dalam sebuah pemrograman.
- c. Membantu proses pembuatan Dokumentasi.

Beberapa simbol flowchart yang digunakan secara umum dan struktur penulisan pseudocode dapat dilihat pada Gambar 2.2 dan 2.3.

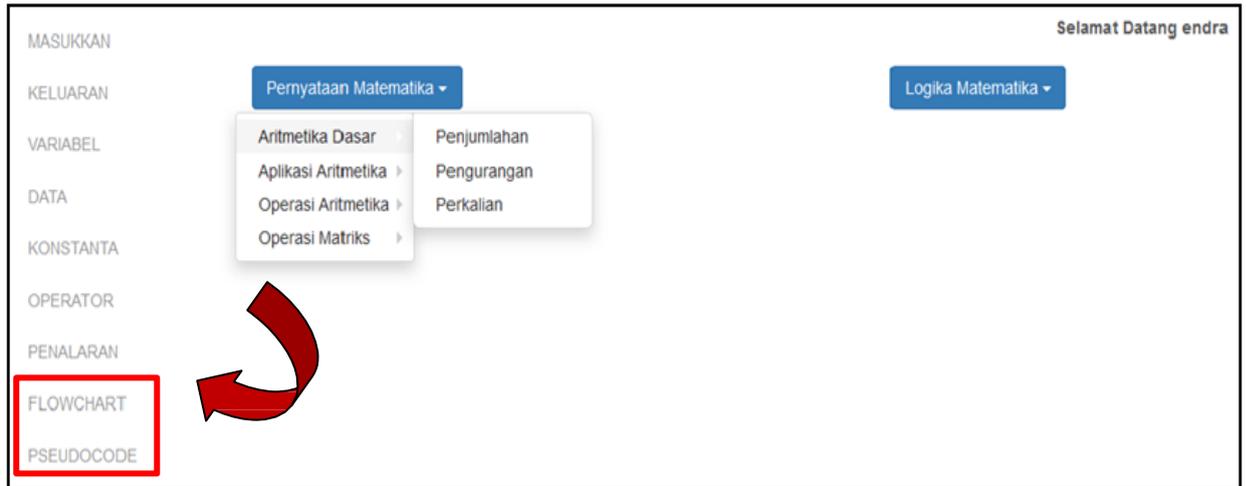


Gambar 2. 2 Simbol Flowchart untuk Pembuatan Algoritma.



Gambar 2. 3 Struktur Penulisan Pseudocode.

Adapun **contoh bentuk penyajian Algoritma dalam bentuk Flowchart dan Pseducode** dapat dilihat dengan memilih setiap menu pada Pernyataan Matematika, kemudian **klik toolbar menu Flowchart dan Pseducode yang terletak pada bagian kiri.**



Gambar 2. 4 Toolbar Menu Flowchart dan Pseudocode dari Form Pernyataan Matematika.

- **Menu Pernyataan Matematika – Aritmatika Dasar**

1. **Pilih dan klik Menu Penjumlahan**

Pada Form Operasi Penjumlahan terdapat 2 teksfield yang harus diisi pertama kali yaitu Bilangan ke-1 dan Bilangan ke-2. Setelah mengisi kedua bilangan tersebut, Anda dapat klik tombol Tampilkan Hasil. Maka akan tampil hasil penjumlahan kedua bilangan tersebut. Klik Toolbar Variabel, Flowchart, dan Pseudocode pada bagian kiri aplikasi.

Operasi Penjumlahan

BILANGAN KE-1	<input style="width: 90%;" type="text" value="10"/>
BILANGAN KE-2	<input style="width: 90%;" type="text" value="7"/>
<input style="width: 100%;" type="button" value="Tampilkan Hasil"/>	<input style="width: 90%;" type="text" value="17"/>

Gambar 2. 5 Form Operasi Penjumlahan

ADA 3 VARIABEL YANG TERLIHAT DILAYAR, YAITU :

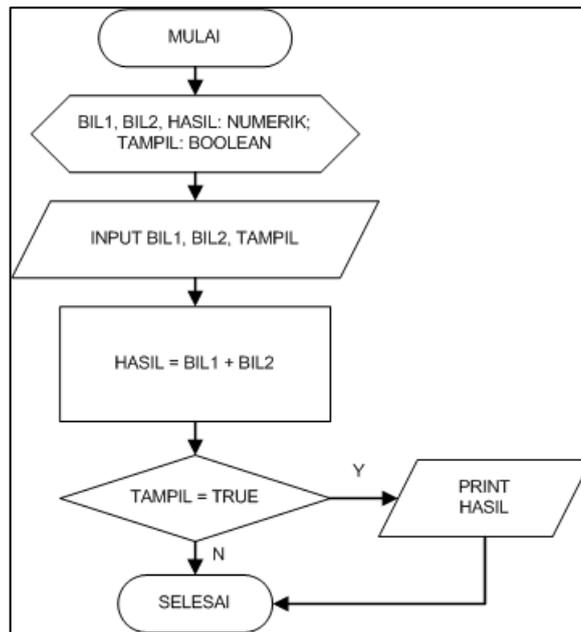
- (1) BILANGAN KE-1 (SIMBOL BIL1)
- (2) BILANGAN KE-2 (SIMBOL BIL-2)
- (3) TAMPILKAN HASIL (SIMBOL TAMPIL)

ADA 1 VARIABEL YANG TIDAK TERLIHAT YAITU :

HASIL PENJUMLAHAN (SIMBOL HASIL, SELENGKAPNYA ADALAH: $HASIL = BIL1 + BIL2$)

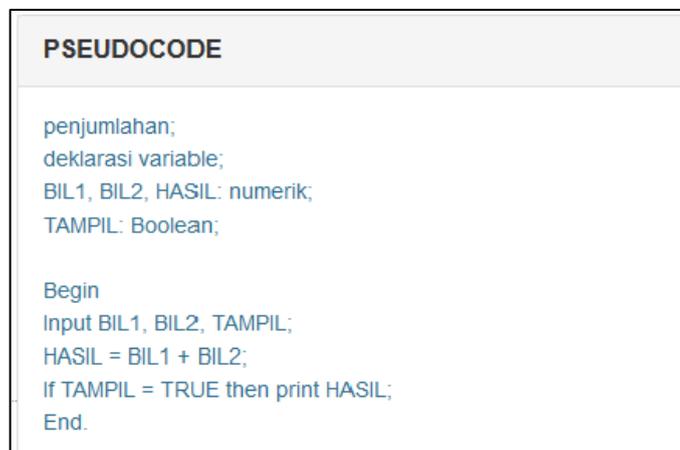
Gambar 2. 6 Kebutuhan Variabel pada Form Operasi Penjumlahan.

a. Flowchart



Gambar 2. 7 Flowchart dari Form Operasi Penjumlahan

b. Pseudocode



Gambar 2. 8 Pseudocode dari Form Operasi Penjumlahan

2. Pilih dan Klik Menu Pengurangan

Pada Form Operasi Pengurangan terdapat 2 teksfield yang harus diisi pertama kali yaitu Bilangan ke-1 dan Bilangan ke-2. Setelah mengisi kedua bilangan tersebut, Anda dapat klik tombol Tampilkan Hasil. Maka akan tampil hasil

pengurangan kedua bilangan tersebut. Klik Toolbar Variabel, Flowchart, dan Pseudocode pada bagian kiri aplikasi.

Operasi Pengurangan

BILANGAN KE-1	<input type="text" value="20"/>
BILANGAN KE-2	<input type="text" value="7"/>
<input type="button" value="Tampilkan Hasil"/>	<input type="text" value="13"/>

Gambar 2. 9 Form Operasi Pengurangan

ADA 3 VARIABEL YANG TERLIHAT DILAYAR, YAITU :

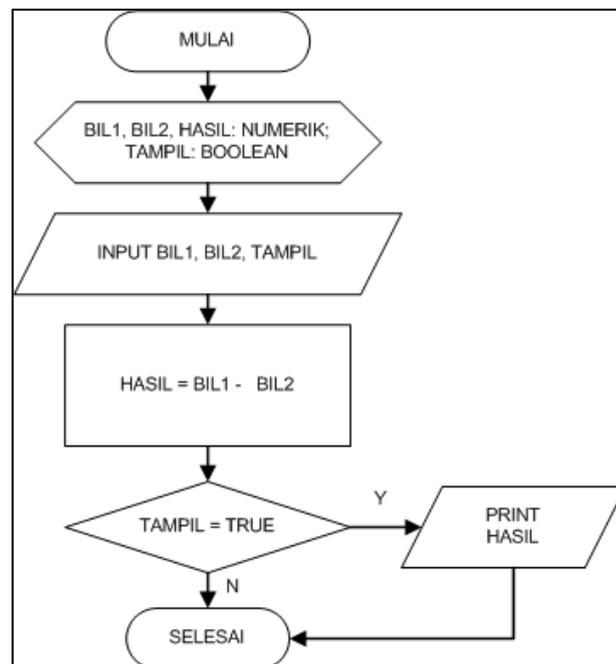
- (1) BILANGAN KE-1 (SIMBOL BIL1)
- (2) BILANGAN KE-2 (SIMBOL BIL-2)
- (3) TAMPILKAN HASIL (SIMBOL TAMPIL)

ADA 1 VARIABEL YANG TIDAK TERLIHAT YAITU :

HASIL PENGURANGAN (SIMBOL HASIL, SELENGKAPNYA ADALAH: $HASIL = BIL1 - BIL2$)

Gambar 2. 10 Kebutuhan Variabel pada Form Operasi Pengurangan.

a. Flowchart



Gambar 2. 11 Flowchart dari Form Operasi Pengurangan.

b. Pseudocode

```
PSEUDOCODE

pengurangan;
deklarasi variable;
BIL1, BIL2, HASIL: numerik;
TAMPIL: Boolean;

Begin
Input BIL1, BIL2, TAMPIL;
HASIL = BIL1 - BIL2;
If TAMPIL = TRUE then print HASIL;
End.
```

Gambar 2. 12 Pseudocode dari Form Operasi Pengurangan.

3. Pilih dan Klik Menu Perkalian

Pada Form Operasi Perkalian terdapat 2 teksfield yang harus diisi pertama kali yaitu Bilangan ke-1 dan Bilangan ke-2. Setelah mengisi kedua bilangan tersebut, Anda dapat klik tombol Tampilkan Hasil. Maka akan tampil hasil perkalian kedua bilangan tersebut. Klik Toolbar Variabel, Flowchart, dan Pseudocode pada bagian kiri aplikasi.

Operasi Perkalian

BILANGAN KE-1: 5

BILANGAN KE-2: 9

Tampilkan Hasil: 45

Gambar 2. 13 Form Operasi Perkalian.

ADA 3 VARIABEL YANG TERLIHAT DILAYAR, YAITU :

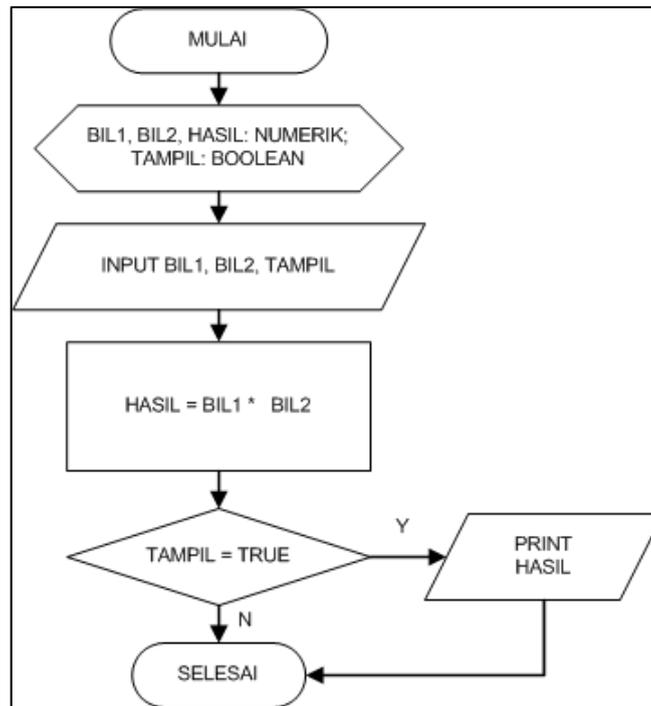
- (1) BILANGAN KE-1 (SIMBOL BIL1)
- (2) BILANGAN KE-2 (SIMBOL BIL-2)
- (3) TAMPILKAN HASIL (SIMBOL TAMPIL)

ADA 1 VARIABEL YANG TIDAK TERLIHAT YAITU :

HASIL PERKALIAN (SIMBOL HASIL, SELENGKAPNYA ADALAH: $HASIL = BIL1 * BIL2$)

Gambar 2. 14 Kebutuhan Variabel pada Form Operasi Perkalian.

1. Flowchart



Gambar 2. 15 Flowchart dari Form Operasi Perkalian.

2. Pseudocode

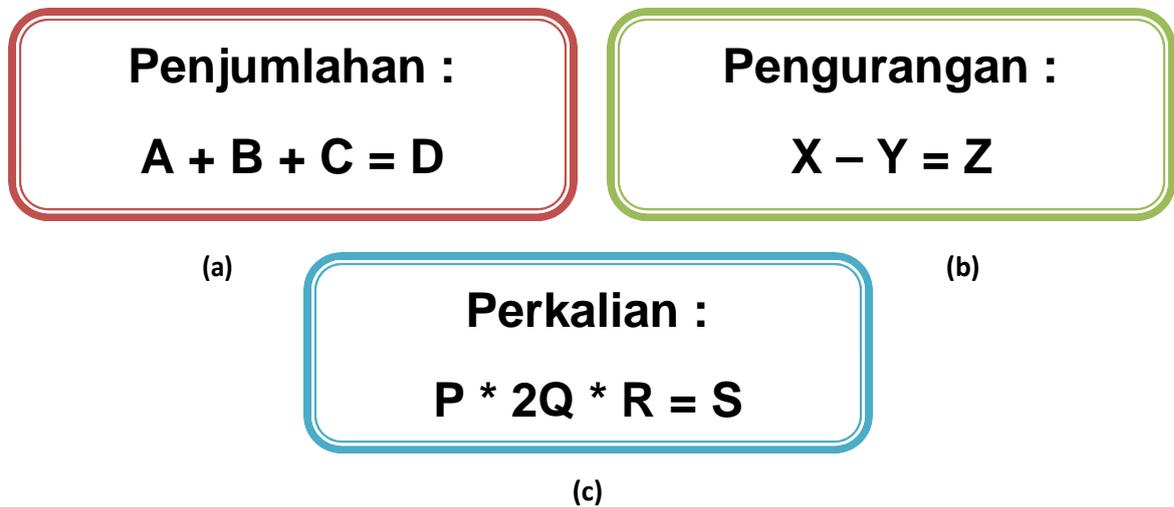


Gambar 2. 16 Pseudocode dari Form Operasi Perkalian.

Pernyataan Matematika Untuk Komputasi

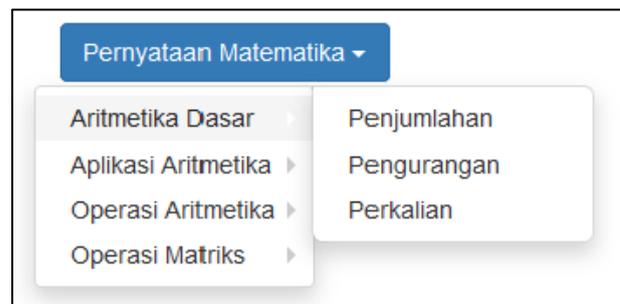
Pernyataan Matematika untuk Komputasi yang dimaksud adalah persamaan matematika dasar untuk perhitungan, seperti penjumlahan, pengurangan, dan

perkalian. Seperti yang diketahui oleh mayoritas peserta didik, operasi perhitungan dasar tersebut adalah landasan utama bagi seseorang untuk dapat menguasai operasi aritmatika. Berikut ini contoh beberapa persamaan yang mewakili perhitungan dasar untuk operasi Penjumlahan, Pengurangan, dan Perkalian :



Gambar 2. 17 Persamaan yang mewakili Penjumlahan, Pengurangan, dan Perkalian.

Pada Aplikasi Dasar Algoritma, Menu Pernyataan Matematika dibagi menjadi beberapa menu seperti yang terlihat pada Gambar 2.18. Pada masing-masing menu tersebut mengandung beberapa pernyataan matematika yang dapat dipelajari untuk mewakili pengolahan Komputasi, yaitu Operasi Penjumlahan, Operasi Pengurangan, dan Operasi Perkalian.



Gambar 2. 18 Menu Aritmetika Dasar.

1. Pilih dan Klik Menu Penjumlahan

Bentuk pernyataan matematika yang digunakan pada Form Operasi Penjumlahan tersebut adalah sebagai berikut :

$$\text{Hasil} = \text{Bil 1} + \text{Bil 2}$$

Operasi Penjumlahan

BILANGAN KE-1	<input type="text" value="10"/>
BILANGAN KE-2	<input type="text" value="7"/>
<input type="button" value="Tampilkan Hasil"/>	<input type="text" value="17"/>

$$\text{Hasil} = \text{Bil 1} + \text{Bil 2}$$

$$\begin{aligned}\text{Hasil} &= 10 + 7 \\ \text{Hasil} &= 17\end{aligned}$$

Gambar 2. 19 Bentuk pernyataan matematika pada Form Operasi Penjumlahan.

2. Pilih dan Klik Menu Pengurangan

Bentuk pernyataan matematika yang digunakan pada Form Operasi Pengurangan tersebut adalah sebagai berikut :

$$\text{Hasil} = \text{Bil 1} - \text{Bil 2}$$

Operasi Pengurangan

BILANGAN KE-1	<input type="text" value="20"/>
BILANGAN KE-2	<input type="text" value="7"/>
<input type="button" value="Tampilkan Hasil"/>	<input type="text" value="13"/>

$$\text{Hasil} = \text{Bil 1} - \text{Bil 2}$$

$$\begin{aligned}\text{Hasil} &= 20 - 7 \\ \text{Hasil} &= 13\end{aligned}$$

Gambar 2. 20 Bentuk pernyataan matematika pada Form Operasi Pengurangan.

3. Pilih dan klik Menu Perkalian

Bentuk pernyataan matematika yang digunakan pada Form Operasi Perkalian tersebut adalah sebagai berikut :

$$\text{Hasil} = \text{Bil 1} * \text{Bil 2}$$

Operasi Perkalian	
BILANGAN KE-1	<input type="text" value="5"/>
BILANGAN KE-2	<input type="text" value="9"/>
<input type="button" value="Tampilkan Hasil"/>	<input type="text" value="45"/>

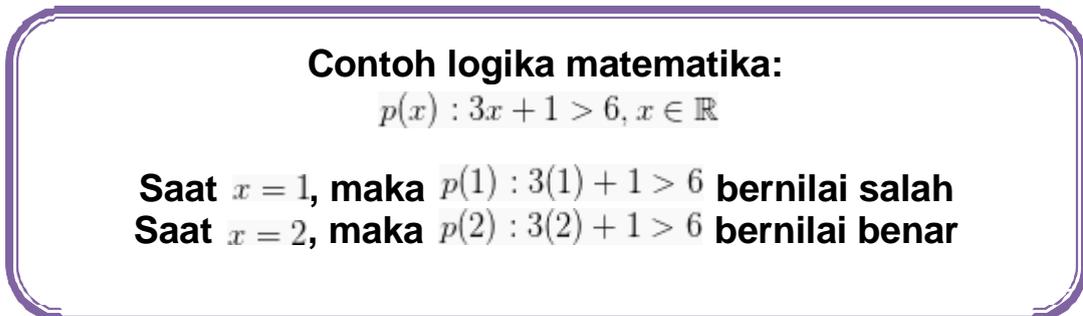
$$\text{Hasil} = \text{Bil 1} * \text{Bil 2}$$

$$\begin{aligned}\text{Hasil} &= 5 * 9 \\ \text{Hasil} &= 45\end{aligned}$$

Gambar 2. 21 Bentuk pernyataan matematika pada Form Operasi Perkalian.

Logika Matematika Untuk Komputasi

Logika matematika adalah cabang logika dan matematika yang mengandung kajian matematis logika dan aplikasi kajian ini pada bidang-bidang lain di luar matematika. Logika matematika berhubungan erat dengan ilmu komputer dan logika filosofis. Contoh Logika Matematika dapat dilihat pada Gambar 2.3.



Gambar 2. 22 Contoh Logika Matematika

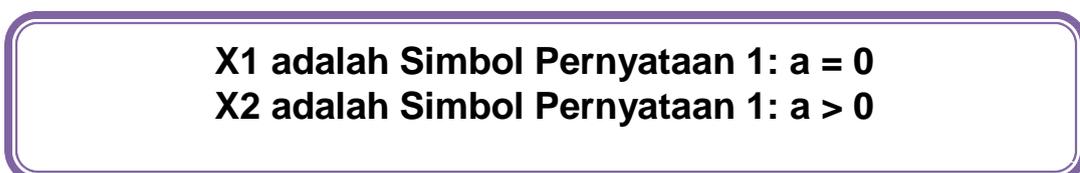
Pada Aplikasi Dasar Algoritma, Menu Logika Matematika dibagi menjadi beberapa menu seperti yang terlihat pada Gambar 2.23. Pada masing-masing menu tersebut mengandung beberapa logika matematika yang dapat dipelajari untuk mewakili pengolahan Komputasi, yaitu Logika Dasar dan Kombinasi Dua Relasi.



Gambar 2. 23 Menu Logika Matematika.

1. Pilih dan klik Menu Logika Dasar

Bentuk Logika Matematika yang digunakan pada Form Logika Dasar adalah sebagai berikut :



LOGIKA DASAR

X1 ADALAH SIMBOL PERNYATAAN 1: $a = 0$
X2 ADALAH SIMBOL PERNYATAAN 2: $a > 0$

MASUKKAN NILAI A

5

Tampilkan Hasil

X1	X2	NOT X1	NOT X2	X1 and X2	X1 or X2
FALSE	TRUE	TRUE	FALSE	FALSE	TRUE

X1 adalah Simbol Pernyataan 1: $a = 0$

X2 adalah Simbol Pernyataan 1: $a > 0$

Jika A diisi dengan angka 5, maka :

1. X1 ($5 = 0$) bernilai **false/salah**.
2. X2 ($5 > 0$) bernilai **true/benar**.
3. Not X1 (Not ($5 = 0$)) \rightarrow (Not (false)) \rightarrow **True**
4. Not X2 (Not ($5 > 0$)) \rightarrow (Not (true)) \rightarrow **False**
5. X1 AND X2 \rightarrow ($5 = 0$) AND ($5 > 0$) \rightarrow False AND True \rightarrow **False**
6. X1 OR X2 \rightarrow ($5 = 0$) OR ($5 > 0$) \rightarrow False OR True \rightarrow **True**

Gambar 2. 24 Bentuk logika matematika pada Form Logika Dasar.

2. Pilih dan klik Menu Kombinasi Dua Relasi

Bentuk Logika Matematika yang digunakan pada Form Kombinasi Dua Relasi adalah sebagai berikut :

Mencari Nilai Terkecil dari 3 buah bilangan:

**Jika $bil[1] < bil[2]$, maka terkecil = $bil[1]$;
Jika tidak, maka terkecil = $bil[2]$;**

**Jika $bil[3] < terkecil$, maka minimal = $bil[3]$;
Jika tidak, maka minimal = terkecil;**

KOMBINASI DUA BUAH RELASI

MASUKKAN TIGA BUAH BILANGAN

6

10

1

Tampilkan Nilai Terkecil

NILAI TERKECIL ADALAH

1

Mencari Nilai Terkecil dari 3 buah bilangan:

bil[1] = 6;
bil[2] = 10;
bil[3] = 1;

Jika bil[1] < bil[2], maka terkecil = bil[1];
Jika tidak, maka terkecil = bil[2];
Jika 6 < 10, maka terkecil = 6

Jika bil[3] < terkecil, maka minimal = bil[3];
Jika tidak, maka minimal = terkecil;
Jika 1 < 6, maka minimal = 1

Sehingga nilai terkecil adalah 1.

Gambar 2. 25 Bentuk Logika Matematika pada Form Kombinasi Dua Variabel.

Rangkuman

- ❖ Flowchart didefinisikan sebagai suatu bagan terurut yang menggambarkan alur terjadinya suatu proses, dengan menggunakan simbol – simbol tertentu yang telah disepakati sesuai dengan aturan komputasi.
- ❖ Pseudocode didefinisikan sebagai bentuk penyajian algoritma dengan menggunakan struktur bahasa tertentu, seperti C dan Pascal.
- ❖ Pernyataan Matematika untuk Komputasi yang dimaksud adalah persamaan matematika dasar untuk perhitungan, seperti penjumlahan, pengurangan, dan perkalian.
- ❖ Logika matematika adalah cabang logika dan matematika yang mengandung kajian matematis logika.

Latihan Soal

1. Pelajari Aplikasi Dasar Algoritma Menu Pernyataan Matematika - Aritmatika Dasar (Penjumlahan, Pengurangan, Perkalian).
 - a. Pada setiap modul sebutkan variabel yang digunakan beserta tipenya, serta fungsinya.
 - b. Pada setiap modul, identifikasi komponen pengolahan datanya.
 - c. Pada setiap komponen proses sebutkan operator yang digunakan.
2. Buatlah persamaan matematika untuk menghitung nilai akhir mahasiswa, dengan aturan: nilai akhir merupakan jumlahan dari 20% nilai UTS, 20% nilai UAS, 20% nilai kehadiran, 40% nilai keaktifan kelas dan tugas.
3. Buatlah persamaan matematika untuk menghitung nilai rata-rata dari dua mahasiswa.
4. Buatlah persamaan matematika untuk menghitung luas lingkaran.
5. Buatlah persamaan matematika untuk menghitung akar persamaan kuadrat

berikut ini: $ax^2 + bx + c = 0$

MODUL 3

VARIABEL DAN DATA

Sub Kompetensi

- Mahasiswa dapat menjelaskan perbedaan antara variabel, konstanta, data serta operator dengan bantuan stimulus dari luar berupa media pembelajaran penalaran matematika.

Tujuan Pembelajaran

- Mahasiswa dapat mengetahui jenis variabel dan konstanta.
- Mahasiswa dapat menjelaskan hubungan variabel dan operator.

Waktu Pelaksanaan Perkuliahan

- Pertemuan Minggu ke-2 dan 3.

Media yang Digunakan

- Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika – Aplikasi Aritmatika (Menghitung Nilai Akhir, Rata-rata dua variabel, Luas Lingkaran, Akar Persamaan Kuadrat).

Variabel dan Tipe Data

Variabel merupakan suatu lokasi memori komputer yang digunakan untuk menampung dan menyimpan data yang akan diolah. Ada beberapa ciri-ciri variabel, yaitu sebagai berikut :

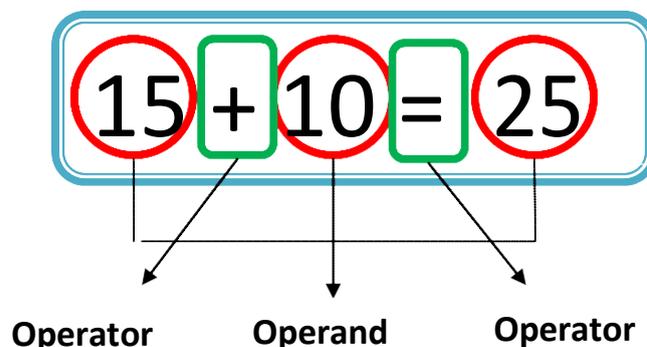
- a. Penamaannya bersifat UNIK, artinya harus berbeda dengan variabel lainnya yang digunakan pada algoritma, contoh : `bilangan1`, `bilangan2`, `hasil`, `luas`, `panjang`, `lebar`, `nama`, `alamat`, dan lain sebagainya.
- b. Satu variabel hanya diperuntukkan menyimpan satu jenis data saja.
- c. Tidak tergantung pada besarnya data.
- d. Nilainya bisa berubah-ubah.

Selain variabel yang selalu berubah, ada juga variabel yang nilainya tetap, yang biasa disebut sebagai **Konstanta**. Konstanta hampir sama dengan variabel. Konstanta merupakan variabel yang mempunyai besaran atau nilai yang tetap selama program dijalankan. Contoh : phi, panjangmaksimal, username, password, dan lain sebagainya.

Data yang akan diprogram memiliki tipe yang bermacam-macam. Tipe data pada algoritma yang paling sering digunakan adalah sebagai berikut :

- **Integer** : bilangan bulat, bukan nilai pecahan (1, 2, 500, 1000)
- **Real** : bilangan real, bilangan dengan digit desimal dibelakang koma (3.5, 5.0, 20.75)
- **Karakter** : data alfanumerik (A, B, *, @, 9, ?)
- **String** : untaian karakter (“STIKOM”, “SISTEM INFORMASI”)
- **Boolean** : hanya terdiri dari 2 nilai yaitu true dan false.

Ekspresi adalah pernyataan atau sebuah proses yang mentransformasikan nilai menjadi hasil (keluaran) yang diinginkan melalui proses perhitungan (komputasi). Ekspresi terdiri dari **Operator dan Operand**. Operand adalah nilai yang diberikan, dapat berupa variabel, konstanta, nilai, dan nilai balik dari fungsi. Sedangkan operator merupakan penghubung antar operand. Perhatikan Gambar 3.1 berikut ini.



Gambar 3. 1 Perbedaan Operator dan Operand.

Adapun pembagian ekspresi pada logika lagoritma dibagi menjadi 3, yaitu :

1. Ekspresi Aritmatika

Ekspresi aritmatika merupakan ekspresi yang digunakan untuk proses perhitungan. Misalnya untuk penjumlahan, pengurangan, perkalian,

pembagian, termasuk menampung hasil (keluaran) dari proses perhitungan tersebut.

➤ **Operator yang digunakan** : ^, *, /, DIV, MOD, +, -, =, ←

Untuk lebih jelas mengenai prioritas dalam penggunaan operator, lihat pada Tabel 3.1.

➤ **Operand yang digunakan** : numerik (angka)

➤ **Hasil (Keluaran) berupa** : numerik (angka)

➤ **Contoh** :

- $A * B$
- $x = (k * i) \text{ mod } 2$
- $\text{hasil} = \text{bil1} - \text{bil2}$

Tabel 3. 1 Prioritas Penggunaan Operator Aritmatika

Prioritas	Operator	Operasi	Tipe Operand	Tipe Hasil
1	^	Pangkat	Real, real Integer, real Real, integer	Real Integer real
2	*	Perkalian	Real,real Integer, integer Real, integer	Real Integer real
	/	Pembagian real	Real, real Integer, integer Real, integer	Real Real Real
	DIV	Pembagian bulat	Integer, integer	integer
	MOD	Sisa Pembagian	Integer, integer	Integer
3	+	Penjumlahan	Real,real Integer, integer Real, integer	Real Integer real
	-	Pengurangan	Real,real Integer, integer Real, integer	Real Integer Real
4	= atau ←	Pemuatan Nilai	Integer Real	Integer Real

2. Ekspresi Relasi

Ekspresi relasi merupakan ekspresi yang digunakan pada algoritma yang mengandung proses percabangan, tepatnya pada kondisi yang berlaku sebagai syarat/aturan dari sebuah permasalahan.

- **Operator yang digunakan** : <, >, <>, =, >=, <=, NOT, AND, OR

Operator Relasi digunakan untuk membandingkan hubungan antara 2 buah operand dan hasil yang didapatkan merupakan keluaran bertipe boolean, true n false.

- **Operand yang digunakan** : numerik, string
- **Hasil (Keluaran) berupa** : Boolean (true, false)
- **Contoh** :

A = 10

B = 30

C = 10

A <> C = false

Untuk lebih jelasnya, perhatikan hasil output penggunaan operator relasi untuk ketiga bilangan tersebut pada Tabel 3.2.

Tabel 3. 2 Penggunaan Operator Relasi dan Hasil Output

Operator	Arti	Contoh	Hasil Output
=	Sama dengan	A= B	false
>	Lebih dari	B > A	true
<	Kurang dari	B < C	false
>=	Lebih dari atau sama dengan	A >=C	true
<=	Kurang dari atau sama dengan	A <=B	true
<>	Tidak sama dengan	A<>C	false

3. Ekspresi String

Ekspresi String digunakan untuk menggabungkan 2 buah string atau lebih. Penggabungan string dilakukan menggunakan operator +.

- **Operator yang digunakan** : +
- **Operand yang digunakan** : string

- **Hasil (Keluaran) berupa :** string
- **Contoh :**
 - kata1 ← “Logika”
 - kata2 ← “Pemrograman Dasar”
 - kata1 + kata2 = “LogikaPemrograman Dasar”

Klasifikasi Variabel Berdasar Komponen Pengolah Data

Penggunaan Variabel memang dapat diklasifikasikan berdasarkan komponen pengolah data. Masih ingat Modul 1 ? Komponen Pengolahan Data terdiri dari 3 komponen yaitu komponen input, proses, dan output. Pada sub bab ini diharapkan Anda dapat memilah dan mengklasifikasikan variabel yang digunakan pada algoritma, mana saja yang termasuk variabel input, variabel proses, dan variabel output. Untuk memudahkan proses klasifikasi variabel, Anda dapat membuka Aplikasi Dasar-dasar Algoritma Komputasi bagian Menu Menghitung Nilai Akhir, seperti pada Gambar 3.2.

Selamat Datang endra

Pernyataan Matematika ▾ Logika Matematika ▾

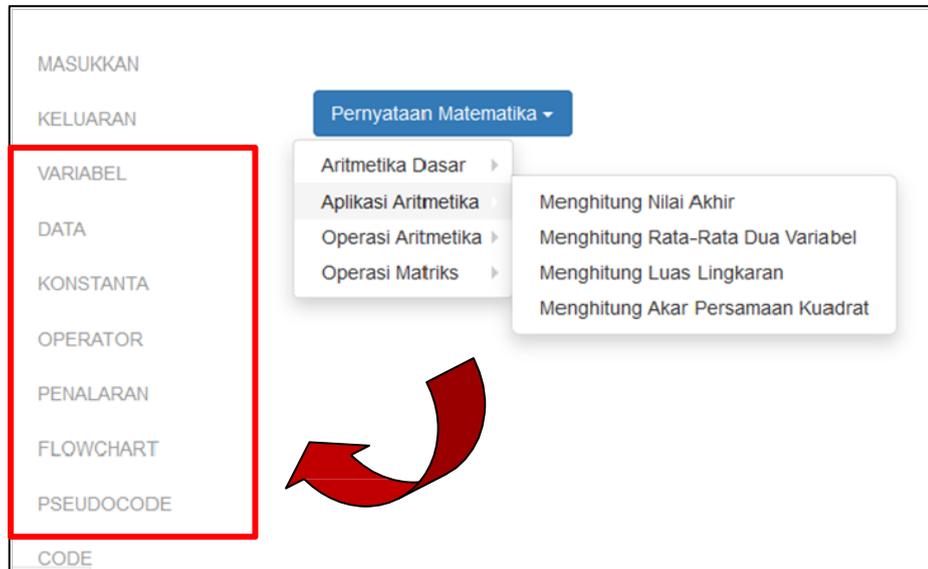
- Aritmetika Dasar ▾
- Aplikasi Aritmetika ▾
 - Menghitung Nilai Akhir
 - Menghitung Rata-Rata Dua Variabel
 - Menghitung Luas Lingkaran
 - Menghitung Akar Persamaan Kuadrat
- Operasi Aritmetika ▾
- Operasi Matriks ▾

Menghitung Nilai Akhir

NILAI KEHADIRAN	<input type="text" value="80"/>
NILAI KEAKTIFAN	<input type="text" value="75"/>
NILAI KEJUJURAN	<input type="text" value="90"/>
NILAI TUGAS	<input type="text" value="60"/>
NILAI UTS	<input type="text" value="85"/>
NILAI UAS	<input type="text" value="90"/>
<input type="button" value="Tampilkan Hasil"/>	84

Gambar 3. 2 Aplikasi Dasar-dasar Algoritma Komputasi bagian Menu Menghitung Nilai Akhir.

Anda dapat **memanfaatkan Toolbar Menu pada bagian sebelah kiri** untuk mengakses Variabel, Data, Konstanta, Operator, Penalaran, Flowchart, dan Pseudocode.



Gambar 3. 3 Sub Menu dari Menu Aplikasi Aritmetika.

- a. Jika **Toolbar Variabel di klik**, maka akan muncul daftar variabel yang dibutuhkan untuk Menghitung Nilai Akhir, seperti Gambar di bawah ini.

ADA 7 VARIABEL YANG TERLIHAT DILAYAR, YAITU :

- (1) Nilai Akhir (nAkhir)
- (2) Nilai Keaktifan (nAktif)
- (3) Nilai Kejujuran (nJujur)
- (4) Nilai Tugas (nTgs)
- {5} Nilai UTS (nUTS)
- (6) Nilai UAS (nUAS)
- (7) Tampil

Ada 1 variabel tidak terlihat di monitor:
Hasil perhitungan nilai akhir, diberi symbol nAkhir:

Aturan menghitung nAkhir adalah:

$$nAkhir = 0.1*nHadir + 0.1*nAktif + 0.5*nJujur + 0.1*nTgs + 0.1*nUts + 0.1* nUas$$

- b. Langkah selanjutnya adalah **klik Toolbar Operator**.

OPERATOR YANG DIGUNAKAN ADALAH : =, +, *
 DIGUNAKAN UNTUK MENGHITUNG NILAI AKHIR DENGAN ATURAN SEPERTI BERIKUT INI :

$$nAkhir = 0.1*nHadir + 0.1*nAktif + 0.5*nJujur + 0.1*nTgs + 0.1*nUts + 0.1* nUas$$

c. Langkah selanjutnya adalah **klik Toolbar Penalaran**.

PENALARAN YANG DIGUNAKAN ADALAH PENALARAN INDUKSI SEBAB AKIBAT

BENTUK PENALARAN
JIKA TAMPIL = TRUE, MAKA CETAK HASIL

Tabel kebenaran

Nama Variable	Masukan	Nilai
TAMPIL	Klik	True
TAMPIL	Tidak Klik	False

d. Kemudian **klik Toolbar Pseudocode dan Flowchart** untuk menampilkan bentuk Pseudocode dari Menghitung Nilai Akhir.

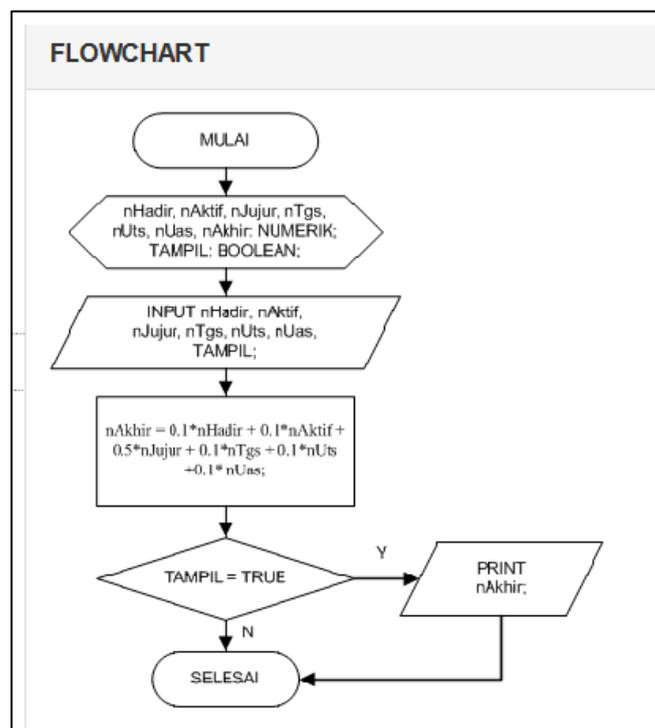
PSEUDOCODE

```

nilai_akhir;
deklarasi variable;
nHadir, nAktif, nJujur, nTgs, nUts, nUas, nAkhir: numerik;
TAMPIL: Boolean;

Begin
Input nHadir, nAktif, nJujur, nTgs, nUts, nUas, TAMPIL;
nAkhir = 0.1*nHadir + 0.1*nAktif + 0.5*nJujur + 0.1*nTgs + 0.1*nUts + 0.1* nUas;
If TAMPIL = TRUE then print nAkhir;
End.

```



Gambar 3. 4 Pseudocode dan Flowchart Perhitungan Nilai Akhir.

- e. Kemudian **klik Toolbar Code** untuk mengetahui bentuk **script/program aplikasi untuk Menghitung Nilai Akhir** menggunakan bahasa pemrograman Java.

```
CODE

public class NilaiAkhir {
    public static void main(String[] args) {
        float nHadir = 100;
        float nAktif = 100;
        float nJujur = 70;
        float nTgs = 100 ;
        float nUts = 90;
        float nUas = 80;
        float nAkhir = 0;

        nAkhir = (float) ((0.1*nHadir) + (0.1*nAktif) + (0.5*nJujur) +
            (0.1*nTgs) + (0.1*nUts) + (0.1* nUas));
        System.out.println("Nilai Akhir : "+nAkhir);
    }
}

Yang tampil adalah code dalam bahasa JAVA.
run:
Nilai Akhir : 82.0
BUILD SUCCESSFUL (total time: 0 seconds)

Jika code dikompile dan dijalankan akan muncul tampilan seperti pada simulasi (area pembelajaran) media pembelajaran ini
```

Gambar 3. 5 Kode Program Perhitungan Nilai Akhir.

Untuk contoh lain, Anda dapat membuka Aplikasi Dasar-dasar Algoritma Komputasi dan **klik pada bagian Menu Menghitung Rata-rata Dua Variabel**, seperti yang terlihat pada Gambar 3.6.

Rata-Rata Dua Variabel	
BILANGAN KE-1	<input type="text" value="89"/>
BILANGAN KE-2	<input type="text" value="45"/>
<input type="button" value="Tampilkan Hasil"/>	<input type="text" value="67"/>

Gambar 3. 6 Aplikasi Dasar-dasar Algoritma Komputasi bagian Menu Menghitung Rata-rata Dua Variabel.

- a. Jika **Toolbar Variabel di klik**, maka muncul daftar variabel yang dibutuhkan untuk Perhitungan Rata-rata Dua Variabel, seperti Gambar di bawah ini.

ADA 3 VARIABEL YANG TERLIHAT DILAYAR, YAITU :

(1) BILANGAN KE-1 (SIMBOL BIL-1)
 (2) BILANGAN KE-2 (SIMBOL BIL-2)
 (3) TAMPILKAN HASIL (SIMBOL TAMPIL)

ADA 1 VARIABEL YANG TIDAK TERLIHAT YAITU :
 RATA-RATA DUA VARIABEL (SIMBOL RATA)
 SELENGKAPNYA ADALAH :
 $RATA = (BIL1 + BIL2) / 2$

- b. Langkah selanjutnya adalah **klik Toolbar Operator**.

OPERATOR YANG DIGUNAKAN ADALAH : =, +, /
 DIGUNAKAN UNTUK MENGHITUNG NILAI RATA-RATA DUA VARIABLE DENGAN SIMBOL :
 $RATA = (BIL1 + BIL2) / 2$

PAHAMI PERBEDAAN MAKNA DUA SIMBOL BERIKUT INI

SIMBOL	MAKNA
$RATA = (BIL1 + BIL2) / 2$	1) BIL1 + BIL2 2) hasil penjumlahan kemudian dibagi 2
$RATA = BIL1 + BIL2 / 2$	1) BIL2 / 2 2) hasil pembagian kemudian ditambah dengan BIL1

- c. Langkah selanjutnya adalah **klik Toolbar Penalaran**.

PENALARAN YANG DIGUNAKAN ADALAH PENALARAN INDUKSI SEBAB AKIBAT

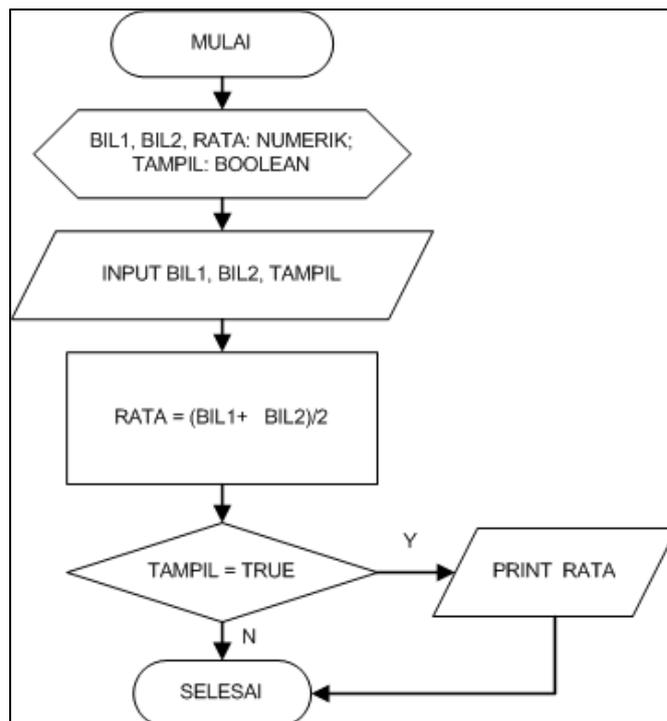
BENTUK PENALARAN
 JIKA TAMPIL = TRUE, MAKA CETAK HASIL

Tabel kebenaran

Nama Variable	Masukan	Nilai
TAMPIL	Klik	True
TAMPIL	Tidak Klik	False

- d. Kemudian **klik Toolbar Pseudocode dan Flowchart** untuk menampilkan bentuk Pseudocode dari Menghitung Rata-rata Dua Variabel.

```
PSEUDOCODE  
  
rata_dua_variable;  
deklarasi variable;  
BIL1, BIL2, RATA: numerik;  
TAMPIL: Boolean;  
  
Begin  
Input BIL1, BIL2, TAMPIL;  
RATA = (BIL1 + BIL2)/2;  
If TAMPIL = TRUE then print RATA;  
End.
```



Gambar 3. 7 Pseudocode dan Flowchart Perhitungan Rata-rata Dua Buah Variabel.

Rangkuman

- ❖ Variabel merupakan suatu lokasi memori komputer yang digunakan untuk menampung dan menyimpan data yang akan diolah. contoh : bilangan1, bilangan2, hasil, luas, panjang, lebar, nama, alamat, dan lain sebagainya.
- ❖ Konstanta merupakan variabel yang mempunyai besaran atau nilai yang tetap selama program dijalankan. Contoh : phi, panjangmaksimal, username, password, dan lain sebagainya.
- ❖ Ekpresi dibagi menjadi 3 : Aritmatika, Relasi, dan String.
- ❖ Operator ada 3 jenis yaitu Operator Aritmatika, Relasi, dan Logika.

Latihan Soal

Buka Aplikasi Dasar Algoritma dan pilih Menu Aplikasi Aritmetika – Menghitung Luas Lingkaran dan Menghitung Akar Persamaan Kuadrat. Kemudian Buatlah persamaan aljabarnya, kemudian tentukan variabel yang dibutuhkan, bentuk pseudocode dan flowchart untuk pernyataan matematika berikut ini :

1. Menghitung luas lingkaran.
2. Menghitung akar persamaan kuadrat berikut ini:

$$ax^2 + bx + c = 0$$

MODUL 4

PERCABANGAN

Sub Kompetensi

- Mahasiswa dapat menjelaskan penggunaan Konsep Percabangan pada Algoritma Pemrograman Dasar.

Tujuan Pembelajaran

- Mahasiswa dapat menjelaskan penggunaan Variabel dan Operator pada proses percabangan.
- Mahasiswa dapat menjelaskan tahapan pembuatan flowchart untuk proses percabangan.

Waktu Pelaksanaan Perkuliahan

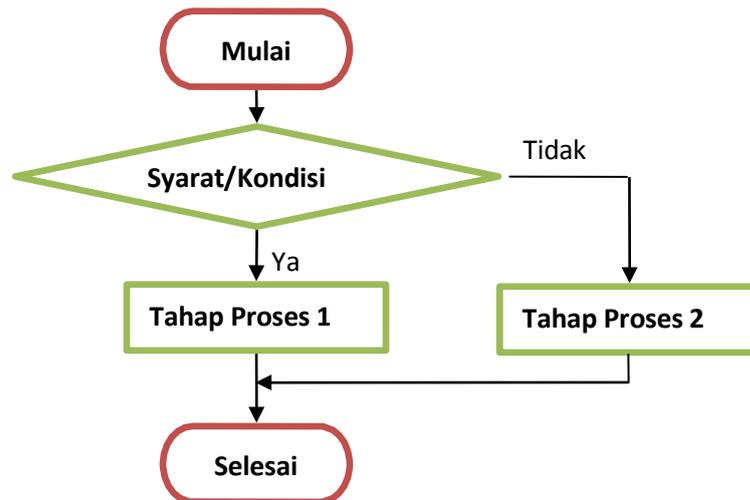
- Pertemuan Minggu ke-4.

Media yang Digunakan

- Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika – Aplikasi Aritmatika (Akar Persamaan Kuadrat).
- Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika – Operasi Aritmatika (Operasi Pembagian).

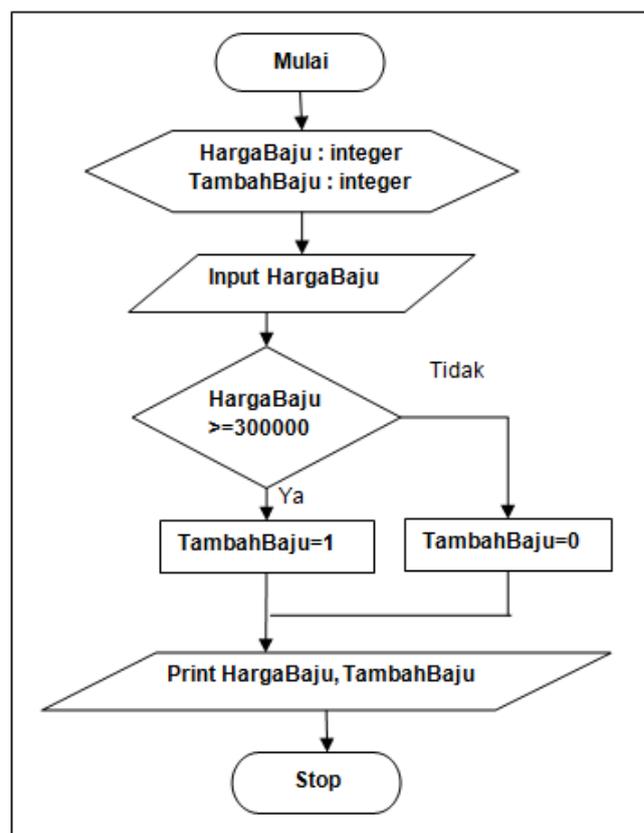
Makna Simbol Flowchart Percabangan

Pada proses percabangan, sebuah perintah/instruksi dapat dijalankan dengan syarat/kondisi tertentu harus dipenuhi terlebih dahulu. Apabila syarat/kondisi belum terpenuhi, maka perintah/instruksi tersebut tidak akan pernah dijalankan. Syarat atau kondisi yang digunakan boleh lebih dari satu kondisi (tidak terbatas). Proses percabangan juga dikenal sebagai proses pemilihan atau proses pengambilan keputusan.



Gambar 4. 1 Simbol Flowchart Proses Percabangan.

Contoh 1 : Pembelian baju di sebuah mall. Jika harga baju diatas Rp 300.000, maka pembeli akan mendapatkan 1 potong baju dengan jenis yang sama. Namun, apabila harga baju tidak melebihi harga Rp 300.000, maka pembeli tidak mendapatkan baju gratis. **Flowchart 1** : Bentuk Flowchartnya dapat dilihat pada Gambar 4.2.

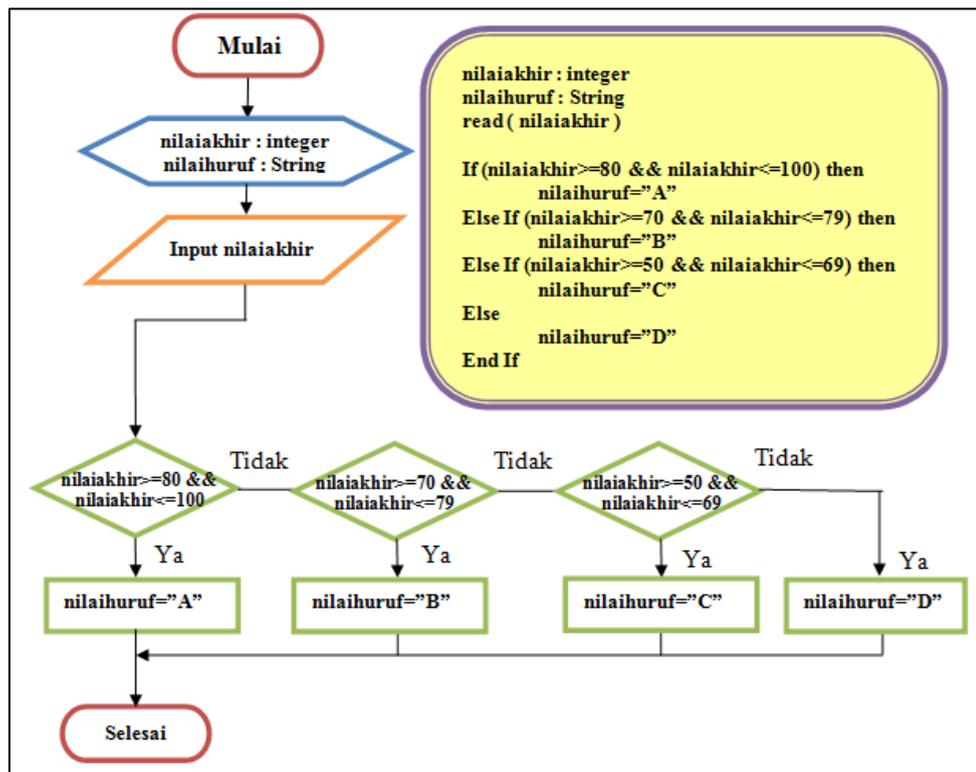


Gambar 4. 2 Flowchart Proses Percabangan Pembelian Baju.

Tabel 4. 1 Penggunaan Operator Logika pada Proses Percabangan.

Prioritas	Operator	Arti
1	NOT	Komplemen Logika
2	AND	Perbandingan secara DAN
3	OR	Perbandingan secara OR

Penggunaan Operator Logika dapat dicontohkan pada Flowchart Gambar 4.4.



Gambar 4. 3 Flowchart Proses Percabangan Nilai Huruf.

Klasifikasi Proses Percabangan Pada Pengolahan Data

Karakteristik yang membedakan operasi pembagian dengan operasi aritmetika yang lain adalah memiliki syarat tertentu, yaitu bilangan pembagi tidak boleh nol. Sehingga dari sisi pengolahan data membutuhkan penanganan yang sedikit berbeda dari operasi aritmetika lainnya.

Contoh Implementasi Konsep Percabangan dapat dilihat pada Aplikasi Belajar Dasar-dasar Algoritma Komputasi, **pilih dan klik pada Menu Pernyataan Matematika – Operasi Aritmatika (Operasi Pembagian).**

a. Anda dapat memulai untuk memilih dan **klik pada Menu Operasi Pembagian**.



Operasi Pembagian

BILANGAN KE-1	<input type="text" value="70"/>
BILANGAN KE-2	<input type="text" value="10"/>
<input type="button" value="Tampilkan Hasil"/>	<input type="text" value="7"/>

b. Klik **Toolbar Menu Variabel** pada samping sebelah kiri.

ADA 3 VARIABEL YANG TERLIHAT DILAYAR, YAITU :

- (1) BILANGAN KE-1 (SIMBOL BIL1)
- (2) BILANGAN KE-2 (SIMBOL BIL-2)
- (3) TAMPILKAN HASIL (SIMBOL TAMPIL)

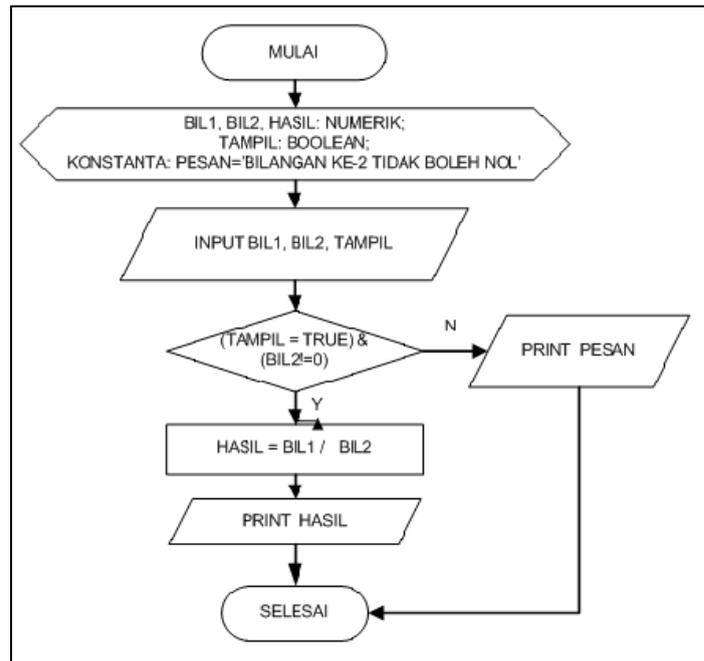
ADA 1 VARIABEL YANG TIDAK TERLIHAT YAITU :

HASIL PEMBAGIAN (SIMBOL HASIL, SELENGKAPNYA ADALAH: $HASIL = BIL1 / BIL2$)

c. Klik **Toolbar Menu Flowchart** untuk menampilkan Flowchart dari Operasi Pembagian.

Pada Operasi Pembagian ini **menggunakan kondisi percabangan, bahwa bilangan yang kedua tidak boleh sama dengan nol.**

- Jika tidak sama dengan nol, maka proses pembagian akan diteruskan.
- Jika sama dengan nol, maka akan muncul pesan bahwa proses pembagian tidak dapat dilakukan, karena proses pembagian dengan pembagi nol akan menghasilkan nilai tak terhingga.



Gambar 4. 4 Flowchart Proses Percabangan Pembagian Dua Buah Bilangan.

Rangkuman

- ❖ Pada proses percabangan, sebuah perintah/instruksi dapat dijalankan dengan syarat/kondisi tertentu harus dipenuhi terlebih dahulu.
- ❖ Syarat atau kondisi yang digunakan boleh lebih dari satu kondisi (tidak terbatas).
- ❖ Operator yang biasanya digunakan pada proses percabangan adalah Operator Logika NOT, AND, dan OR.

Latihan Soal

Buatlah algoritma berupa flowchart untuk kasus dibawah ini :

1. Balita adalah anak-anak yang berusia antara 0 sampai dengan 5 tahun, selain range usi tersebut, berarti bukan balita.
2. Predikat memuaskan HANYA diberikan pada lulusan dengan indeks prestasi antara 3.5 sampai dengan 3.75.
3. Pengemudi di jalan tol dengan kecepatan kurang dari 60 km/jam atau lebih dari 100km/jam akan ditilang.

4. Keluarga sejahtera adalah keluarga yang pendapatan tiap bulan antara 10 juta sampai dengan 15 juta rupiah.
5. Syarat peserta bakti sosial adalah membawa uang minimal Rp. 100.000,00 atau membawa pakaian layak pakai sebanyak-banyaknya 10 potong.
6. Pengemudi yang baik adalah pengemudi yang menjaga kecepatan kendaraan di jalan tol antara 60 sampai dengan 100 km.

MODUL 5

PERULANGAN 1

Sub Kompetensi

- Mahasiswa dapat menjelaskan penggunaan Konsep Perulangan pada Algoritma Pemrograman Dasar.

Tujuan Pembelajaran

- Mahasiswa dapat menjelaskan tahapan pembuatan flowchart untuk proses perulangan.
- Mahasiswa dapat membedakan percabangan dan perulangan.
- Mahasiswa dapat menjelaskan contoh penerapan perulangan pada Kounter dan Akumulator .

Waktu Pelaksanaan Perkuliahan

- Pertemuan Minggu ke-5.

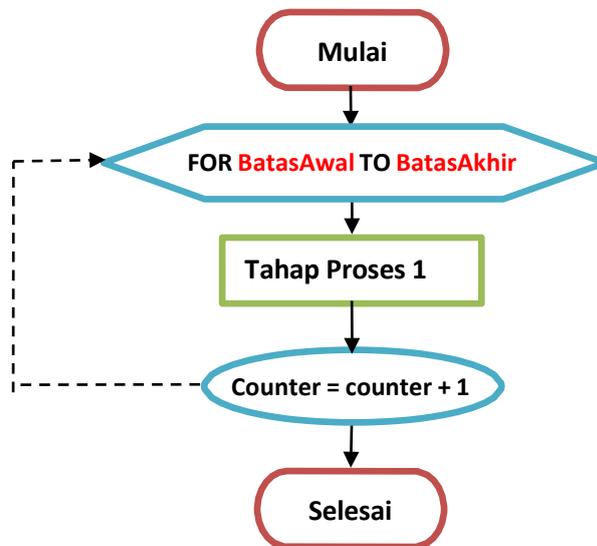
Media yang Digunakan

- Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika – Operasi Aritmatika (Kounter Dasar 1 dan 2, Akumulasi Penjumlahan).

Konsep Dasar Perulangan

Dalam sebuah algoritma, tahapan penyelesaian sebuah permasalahan dapat berupa struktur berurutan (*sequence*), struktur percabangan (*selection*), dan struktur perulangan (*loopinh*). Ketiga jenis langkah tersebut membentuk struktur dasar dari algoritma. Proses perulangan adalah sebuah proses program dimana setiap perintah/instruksi dapat dijalankan secara berulang dengan kondisi tertentu. Kondisi tersebut digunakan sebagai batas awal, batas akhir perulangan, dan Kounter. Kounter adalah sebuah proses otomatis yang biasa digunakan dalam aktivitas sehari-hari. Contoh counter adalah membuat nomer urut, misal 1, 2, 3, dst. Pada symbol matematika kounter digunakan untuk membuat indeks. Contoh x_i , dengan i

mulai dari 1, 2, 3, dan seterusnya. Untuk Flowchart dasar Perulangan dapat dilihat pada Gambar 5.1.



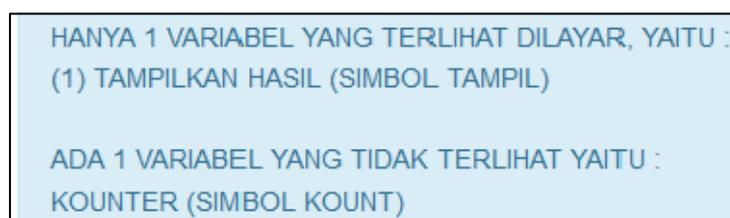
Gambar 5. 1 Flowchart Dasar Perulangan For.

Pada Aplikasi Belajar Dasar-dasar Algoritma Komputasi : **Menu Pernyataan Matematika – Operasi Aritmatika – Membuat Kounter (Kounter Dasar 1)** digunakan untuk menampilkan bilangan secara urut mulai dari 1 sampai dengan 10.

- a. Anda dapat memulai untuk memilih dan **klik pada Menu Membuat Kounter (Kounter Dasar 1)**.



- b. Klik **Toolbar Menu Variabel** pada samping sebelah kiri.



c. Klik **Toolbar Menu Data** pada samping sebelah kiri.

ADA 10 DATA YANG TERLIHAT DILAYAR, YAITU :
1 2 3 4 5 6 7 8 9 10

ADA 1 DATA YANG TIDAK TERLIHAT YAITU :
TRUE (NILAI TAMPIL)

d. Klik **Toolbar Menu Operator** pada samping sebelah kiri.

OPERATOR YANG DIGUNAKAN ADALAH : =, +
DIGUNAKAN UNTUK MENGHITUNG KOUNT, DENGAN SIMBOL :
KOUNT = KOUNT + 1

e. Klik **Toolbar Menu Penalaran** pada samping sebelah kiri.

Model Penalaran yang digunakan pada Pembuatan Kounter Dasar 1 dapat dilihat pada Gambar 5.2. Sedangkan Tabel Kebenarannya dapat dilihat pada Gambar 5.3. Pada tabel kebenaran terlihat proses perulangan yang terjadi mulai dari angka = 1 hingga angka = 10 dapat tercetak di Layar Aplikasi.

Operasi Kounter Dasar 1

PENALARAN YANG DIGUNAKAN ADALAH PENALARAN INDUKSI SEBAB AKIBAT

ADA 2 PENALARAN YANG DIGUNAKAN

PENALARAN 1: TERLIHAT DI LAYAR
JIKA TAMPIL = TRUE, MAKA LAKUKAN PROSES CETAK KOUNTER

Tabel kebenaran

Nama Variable	Masukan	Nilai
TAMPIL	Tekan	True
TAMPIL	Tidak Tekan	False

PENALARAN 2: TIDAK TERLIHAT DI LAYAR
PENALARAN 2 DIGUNAKAN UNTUK MENGHITUNG NILAI KOUNTER

JIKA KOUNT < 11 MAKA CETAK KOUNT;
KOUNT=KOUNT+1;
PERIKSA ULANG NILAI KOUNT;

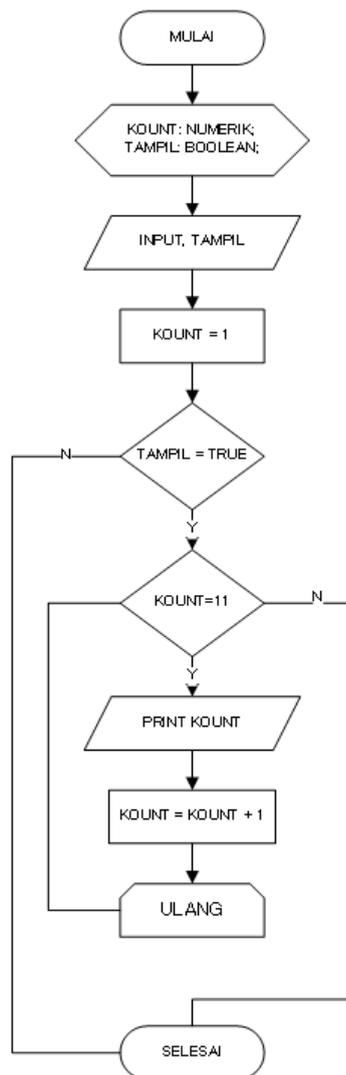
Gambar 5. 2 Model Penalaran Pembuatan Kounter Dasar 1.

Tabel kebenaran berikut terjadi jika nilai TAMPIL=True.
 Nilai Kount awal adalah 1

Proses Ke	Nilai Relasi Kount<11	Terlihat di layar	Perubahan Nilai Kount=Kount+1
1	True	1	2
2	True	1 2	3
3	True	1 2 3	4
4	True	1 2 3 4	5
5	True	1 2 3 4 5	6
6	True	1 2 3 4 5 6	7
7	True	1 2 3 4 5 6 7	8
8	True	1 2 3 4 5 6 7 8	9
9	True	1 2 3 4 5 6 7 8 9	10
10	True	1 2 3 4 5 6 7 8 9 10	11
11	False	1 2 3 4 5 6 7 8 9 10	

Gambar 5. 3 Tabel Kebenaran Pembuatan Kounter Dasar 1.

f. Klik **Toolbar Menu Penalaran** pada samping sebelah kiri.



Gambar 5. 4 Flowchart Perulangan Pembuatan Kounter Dasar 1.

Pada Gambar 5.4 menunjukkan bagaimana struktur flowchart perulangan Pembuatan Kounter Dasar 1. Cobalah Anda membuka Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika – Operasi Aritmatika (Kounter Dasar 2). Kemudian buatlah struktur Flowchart yang tepat untuk Pembuatan Kounter Dasar 2 tersebut.

Penalaran Sigma Dalam Komputasi

Operasi matematika menekankan pada pemikiran tentang model aljabar untuk symbol-simbol matematika yang tidak ada pada keyboard, seperti sigma (Σ) dan phi (Π). Symbol sigma (Σ) mewakili penjumlahan berulang atau akumulasi penjumlahan (Akumulator). Sedangkan symbol phi (Π) mewakili akumulasi perkalian.

Akumulator adalah proses penjumlahan berulang dengan menambahkan secara berurutan hasil penjumlahan dengan bilangan berikutnya. Symbol secara matematisnya adalah sebagai berikut.

$$S = \sum_{i=1}^n x_i$$

Nilai variable S diperoleh dengan cara menjumlahkan nilai x dari 1 sampai ke n.

Contoh 1, untuk n=5 dan nilai x adalah sebagai berikut.

n	x
1	10
2	35
3	25
4	20
5	40

Maka

$$S = \sum_{i=1}^n x_i = 10 + 35 + 25 + 20 + 40 = 130$$

Contoh 2, untuk n=7 dan nilai x adalah sebagai berikut.

n	x
1	10
2	35
3	25
4	20
5	40
6	10
7	10

Maka :

$$S = \sum_{i=1}^n x_i = 10 + 35 + 25 + 20 + 40 + 10 + 10 = 150$$

$i=1$

Contoh implementasi Akumulasi Penjumlahan pada Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika – Operasi Aritmatika (Akumulasi Penjumlahan) , dapat dilihat pada Gambar di bawah ini.

- a. Anda dapat memulai untuk memilih dan **klik pada Menu Operasi Matematika (Akumulasi Penjumlahan-Akumulator)**.

MEMBUAT AKUMULATOR

Nilai Transaksi

Total Transaksi

- b. Klik **Toolbar Menu Variabel** pada samping sebelah kiri.

ADA 4 VARIABEL YANG TERLIHAT DILAYAR, YAITU nTrans, tLagi, tampilHasil, dan totTrans.

c. Klik **Toolbar Menu Data dan Operator** pada samping sebelah kiri.

<p>TERDAPAT 2 DATA YANG TERLIHAT DILAYAR, YAITU :</p> <p>(1) 50 (Nilai nTrans)</p> <p>(2) 50 (totTrans)</p> <p>ADA 2 DATA YANG TIDAK TERLIHAT YAITU : TRUE (NILAI tLagi, dan NILAI tampilHasil)</p>	<p>OPERATOR YANG DIGUNAKAN ADALAH : =, + DIGUNAKAN UNTUK MENGHITUNG totTrans, DENGAN SIMBOL</p> <p>totTrans = totTrans + nTrans;</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------

d. Klik **Toolbar Menu Penalaran** pada samping sebelah kiri.

Model Penalaran yang digunakan pada Pembuatan Akumulasi Penjumlahan dapat dilihat pada Gambar 5.5. Sedangkan Tabel Kebenarannya dapat dilihat pada Gambar 5.6. Pada tabel kebenaran terlihat proses perhitungan total transaksi hingga dapat tercetak di Layar Aplikasi.

Membuat Akumulator

PENALARAN YANG DIGUNAKAN ADALAH PENALARAN INDUKSI SEBAB AKIBAT

ADA 2 PENALARAN YANG TERLIHAT DI LAYAR

1. JIKA tLagi = TRUE, MAKA
CETAK NTRANS, totTrans
ULANG PENGISIAN NILAI TRANSAKSI

Tabel kebenaran

Nama Variable	Masukan	Nilai
tLagi	Tekan	True
tLagi	Tidak Tekan	False

Perubahan nilai variable nTrans dan totTrans adalah sebagai berikut.

Proses Ke	nTrans	tLagi	totTrans=totTrans+nTrans	Terlihat di Layar (nTrans)	Terlihat di Layar (totTrans)
1.	50	True	50	50	50
2.	100	True	150	50 100	150

Gambar 5. 5 Model Penalaran Pembuatan Akumulasi Penjumlahan.

2. JIKA tampilHasil = TRUE, MAKA
CETAK NTRANS, totTrans

Tabel kebenaran

Nama Variable	Masukan	Nilai
tampilHasil	Tekan	True
tampilHasil	Tidak Tekan	False

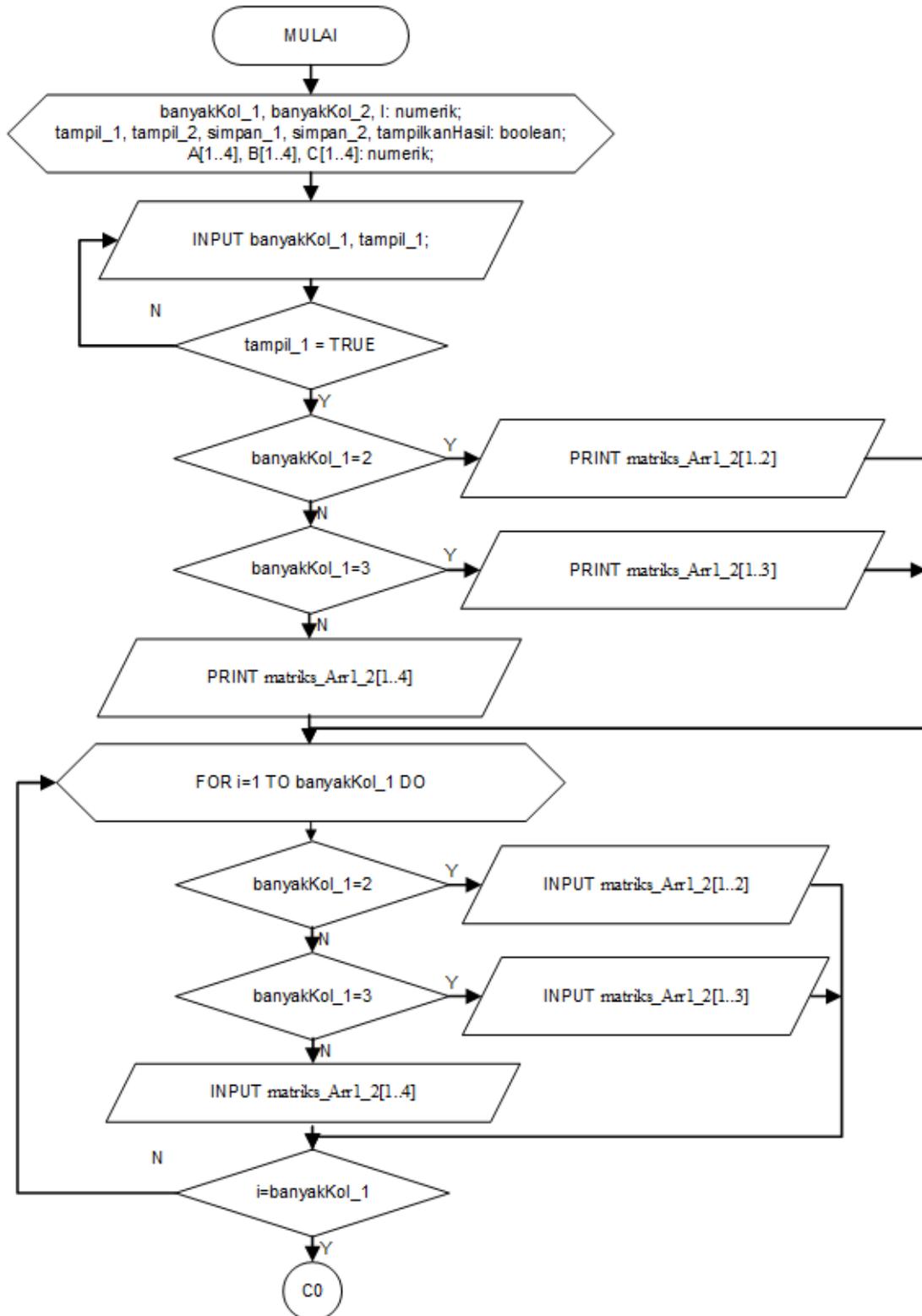
Perubahan nilai variable nTrans dan totTrans adalah sebagai berikut.

Proses Ke	nTrans	tLagi	totTrans=totTrans+nTrans	Terlihat di Layar (nTrans)	Terlihat di Layar (totTrans)
1.	50	True	50	50	50

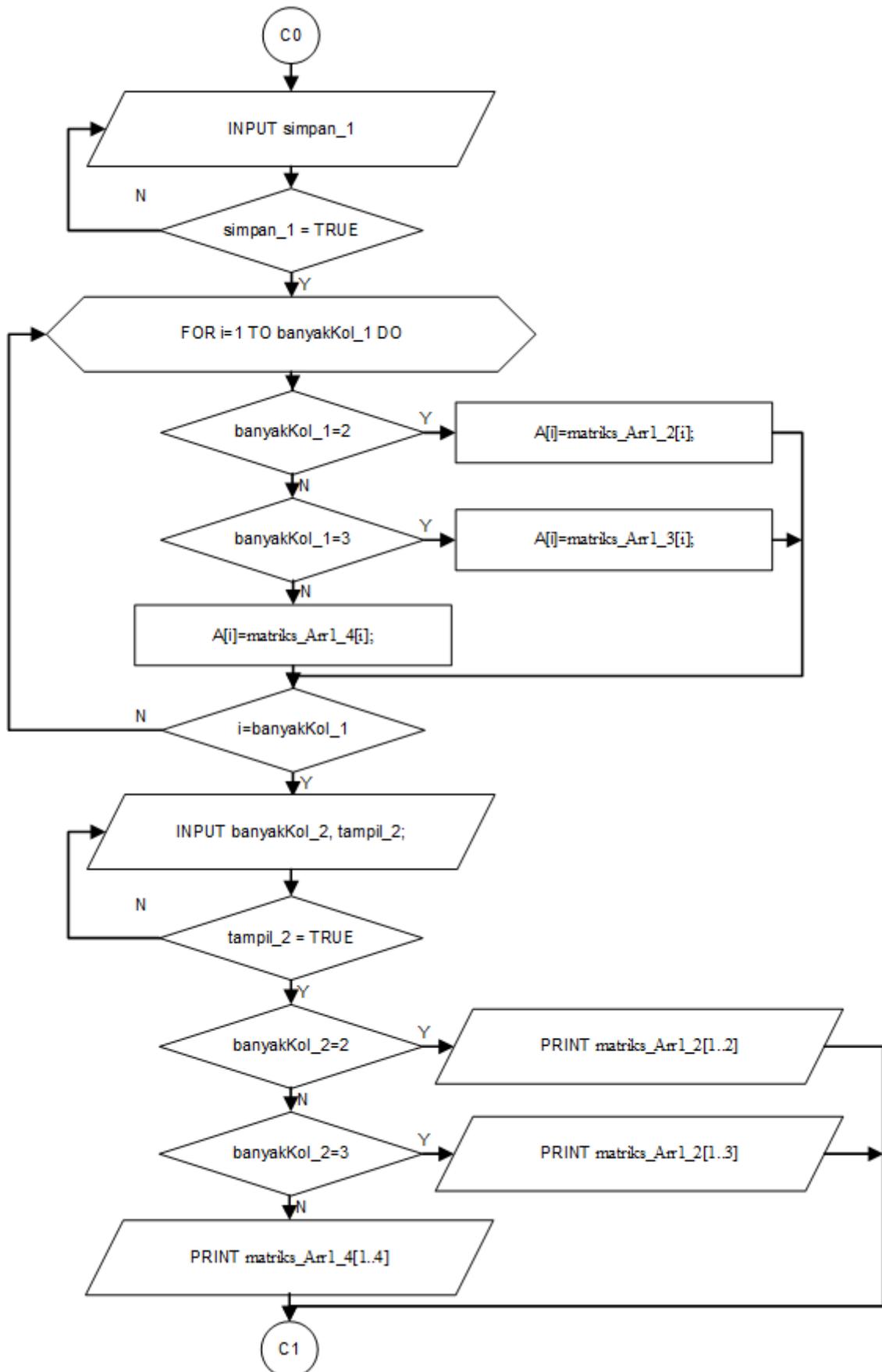
Gambar 5. 6 Tabel Kebenaran Pembuatan Akumulasi Penjumlahan.

e. Klik **Toolbar Menu Flowchart** pada samping sebelah kiri.

Pada Gambar 5.7 dan Gambar 5.8 menunjukkan bagaimana struktur flowchart perulangan Pembuatan Akumulasi Penjumlahan.



Gambar 5. 7 Flowchart Perulangan Pembuatan Akumulasi Penjumlahan.



Gambar 5. 8 Flowchart Perulangan Pembuatan Akumulasi Penjumlahan (Lanjutan).

Rangkuman

- ❖ Proses perulangan adalah sebuah proses program dimana setiap perintah/instruksi dapat dijalankan secara berulang dengan kondisi tertentu.
- ❖ Kounter adalah sebuah proses otomatis yang biasa digunakan dalam aktivitas sehari-hari. Contoh counter adalah membuat nomer urut, misal 1, 2, 3, dst.
- ❖ Operasi matematika menekankan pada pemikiran tentang model aljabar untuk symbol-simbol matematika yang tidak ada pada keyboard, seperti sigma (Σ) dan phi (Π).
- ❖ Akumulator adalah proses penjumlahan berulang dengan menambahkan secara berurutan hasil penjumlahan dengan bilangan berikutnya.

Latihan Soal

1. Buatlah flowchart dengan inputan berupa tombol, yaitu jika tombol ditekan sebanyak 10 kali maka akan mengirim data serial ke komputer "TOMBOL TELAH DITEKAN 10 KALI".
2. Tampilkan bilangan 1 sampai dengan 100 secara otomatis.
3. Buatlah deret bilangan genap yang terletak antara -4 sampai dengan 50.
4. Buatlah autentifikasi login aplikasi.

MODUL 6 PERULANGAN 2

Sub Kompetensi

- Mahasiswa dapat menjelaskan penggunaan Konsep Perulangan pada Algoritma Pemrograman Dasar.

Tujuan Pembelajaran

- Mahasiswa dapat menjelaskan tahapan pembuatan flowchart untuk proses perulangan.
- Mahasiswa dapat membedakan percabangan dan perulangan.
- Mahasiswa dapat menjelaskan contoh penerapan perulangan pada Akumulasi Perkalian dan Faktorial.

Waktu Pelaksanaan Perkuliahan

- Pertemuan Minggu ke-6.

Media yang Digunakan

- Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika – Operasi Aritmatika (Akumulasi Perkalian dan Faktorial).

Penalaran Phi (Multiplikator)

Multiplikator adalah proses perkalian berulang secara berurutan, hasil perkalian dikalikan dengan dengan berikutnya. Symbol secara matematisnya adalah sebagai berikut.

$$S = \prod_{i=1}^n x_i$$

Nilai variable S diperoleh dengan cara melakukan perkalian nilai x dari 1 sampai ke-n. Contoh 1, untuk n=5 dan nilai x adalah sebagai berikut.

n	x
1	10
2	35
3	25
4	20
5	40

Maka :

$$S = \prod_{i=1}^n x_i = 10 * 35 * 25 * 20 * 40 = 7.000.000$$

Contoh 2, untuk n=7 dan nilai x adalah sebagai berikut :

n	x
1	2
2	-3
3	2
4	0
5	40
6	10
7	10

Maka :

$$S = \prod_{i=1}^n x_i = 2 * (-3) * 2 * 0 * 40 * 10 * 10 = 0$$

Contoh implementasi Akumulasi Perkalian pada Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika – Operasi Aritmatika (Akumulasi Perkalian) , dapat dilihat pada Gambar 6.1.

- a. Anda dapat memulai untuk memilih dan **klik pada Menu Operasi Matematika (Akumulasi Perkalian-Multiplikator)**.

MEMBUAT MULTIPLIKATOR

Nilai Transaksi	<input type="text" value="10"/>	<input type="button" value="Transaksi Lagi"/>
	<input type="text" value="5 7 10"/>	<input type="button" value="Tampilkan Hasil"/>
Total Transaksi	<input type="text" value="350"/>	

- b. Klik **Toolbar Menu Variabel** pada samping sebelah kiri.

ADA 4 VARIABEL YANG TERLIHAT DILAYAR, YAITU :
nTrans, tLagi, tampilHasil, dan hasil.

- c. Klik **Toolbar Menu Data** pada samping sebelah kiri.

TERDAPAT 2 DATA YANG TERLIHAT DILAYAR, YAITU :
(1) 50 (Nilai nTrans)
(2) 50 (hasil)

ADA 2 DATA YANG TIDAK TERLIHAT YAITU :
TRUE (NILAI tLagi, dan NILAI tampilHasil)

- d. Klik **Toolbar Menu Operator** pada samping sebelah kiri.

OPERATOR YANG DIGUNAKAN ADALAH : =, *
DIGUNAKAN UNTUK MENGHITUNG hasil, DENGAN SIMBOL :

hasil = hasil * nTrans;

- e. Klik **Toolbar Menu Penalaran** pada samping sebelah kiri.

Model Penalaran yang digunakan pada Pembuatan Akumulasi Perkalian dapat dilihat pada Gambar 6.1. Sedangkan Tabel Kebenarannya dapat dilihat pada Gambar 6.2. Pada tabel kebenaran terlihat proses perhitungan perkalian transaksi hingga dapat tercetak di Layar Aplikasi.

Membuat Multiplikator

PENALARAN YANG DIGUNAKAN ADALAH PENALARAN INDUKSI SEBAB AKIBAT

ADA 2 PENALARAN YANG TERLIHAT DI LAYAR

1. JIKA tLagi = TRUE, MAKA
 CETAK nTrans, hasil
 ULANG PENGISIAN NILAI TRANSAKSI

Tabel kebenaran

Nama Variable	Masukan	Nilai
tLagi	Tekan	True
tLagi	Tidak Tekan	False

Perubahan nilai variable nTrans dan totTrans adalah sebagai berikut.

Proses Ke	nTrans	tLagi	hasil=hasil*nTrans	Terlihat di Layar (nTrans)	Terlihat di Layar (hasil)
1.	50	True	50	50	50
2.	100	True	150	50 100	5000

Gambar 6. 1 Model Penalaran Pembuatan Akumulasi Perkalian.

2. JIKA tampilHasil = TRUE, MAKA
 CETAK nTrans, hasil

Tabel kebenaran

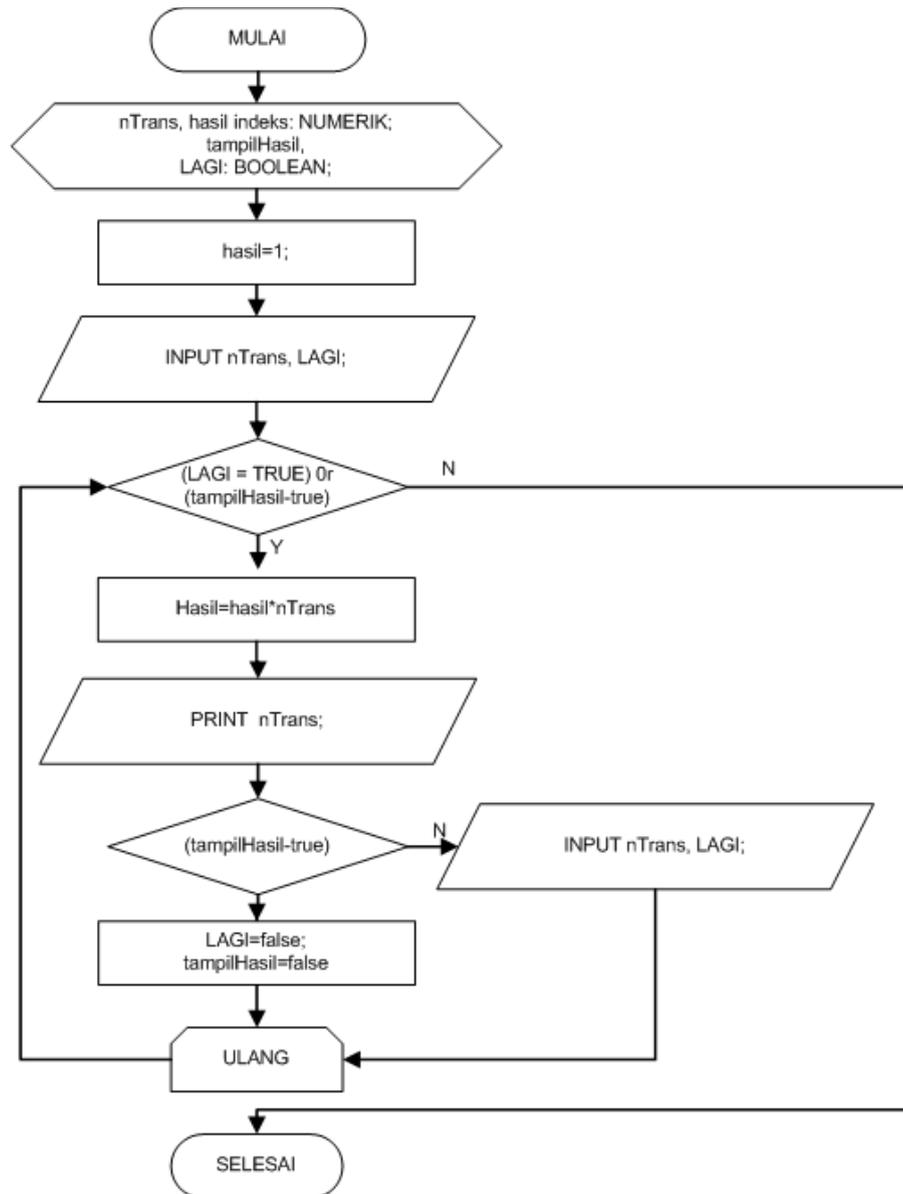
Nama Variable	Masukan	Nilai
tampilHasil	Tekan	True
tampilHasil	Tidak Tekan	False

Perubahan nilai variable nTrans dan totTrans adalah sebagai berikut.

Proses Ke	nTrans	tampilHasil	hasil=hasil*nTrans	Terlihat di Layar (nTrans)	Terlihat di Layar (hasil)
1.	50	True	50	50	50

Gambar 6. 2 Tabel Kebenaran Pembuatan Akumulasi Perkalian.

f. Klik **Toolbar Menu Flowchart** pada samping sebelah kiri.



Gambar 6. 3 Flowchart Perulangan Pembuatan Akumulasi Perkalian.

Penalaran Faktorial Dalam Komputasi

Factorial adalah proses perkalian berulang secara berurutan, hasil perkalian dikalikan dengan bilangan berikutnya, dengan nilai bilangan otomatis, dimulai dari nol sampai dengan n (sesuai kebutuhan). Symbol secara matematisnya adalah sebagai berikut.

$$fakto(n) = \prod_{i=1}^n i$$

Nilai fungsi $fakto(n)$ diperoleh dengan cara melakukan perkalian bilangan 1 sampai dengan n.

Contoh fungsi $fakto(n)$:

n	$fakto(n)$
1	1
2	1*2=2
3	1*2*3=6
4	1*2*3*4=24
...
n	1*2*3*...*n

Contoh implementasi Perhitungan Faktorial pada Aplikasi Belajar Dasar-dasar Algoritma Komputasi : **Menu Pernyataan Matematika – Operasi Aritmatika (Perhitungan Faktorial).**

- Anda dapat memulai untuk memilih dan **klik pada Menu Operasi Matematika (Perhitungan Faktorial).**

FUNGSI FAKTORIAL

Masukan Bilangan Tampilkan Hasil

6*5*4*3*2*1

HASIL 720

Gambar 6. 4 Aplikasi Dasar Algoritma – Menghitung Faktorial

- a. Klik **Toolbar Menu Variabel** pada samping sebelah kiri.

ADA 3 VARIABEL YANG TERLIHAT DILAYAR, YAITU :
n, tampilHasil, fak.

- b. Klik **Toolbar Menu Data** pada samping sebelah kiri.

TERDAPAT 2 DATA YANG TERLIHAT DILAYAR, YAITU :
(1) 3 (Nilai n)
(2) 6 (hasil factorial 3)

ADA 1 DATA YANG TIDAK TERLIHAT YAITU :
TRUE (NILAI tampilHasil)

- c. Klik **Toolbar Menu Operator** pada samping sebelah kiri.

OPERATOR YANG DIGUNAKAN ADALAH : =, *
DIGUNAKAN UNTUK MENGHITUNG fungsi faktorial, DENGAN SIMBOL :

fakto = fakto *n;

- d. Klik **Toolbar Menu Penalaran** pada samping sebelah kiri.

Fungsi Faktorial

PENALARAN YANG DIGUNAKAN ADALAH PENALARAN INDUKSI SEBAB AKIBAT

ADA 1 PENALARAN YANG TERLIHAT DI LAYAR

1. JIKA tampilHasil = TRUE, MAKA
 CETAK n, fakto;

Tabel kebenaran

Nama Variable	Masukan	Nilai
tampilHasil	Tekan	True
tampilHasil	Tidak Tekan	False

ADA 1 PENALARAN YANG TIDAK TERLIHAT DI LAYAR

1. JIKA $n > 0$, MAKA
 Fakto=fakto*n;
 CETAK n,"*";
 n=n-1; ulang periksa nilai n (apakah $n > 0$);
 jika tidak demikian maka cetak "=", fakto.

Gambar 6. 5 Model Penalaran Perhitungan Faktorial.

Tabel kebenaran

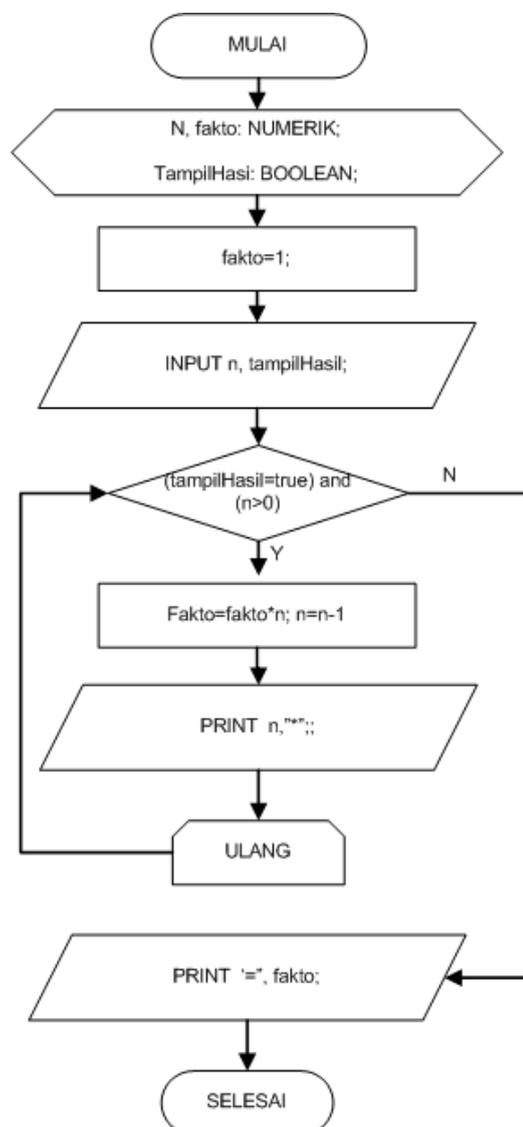
Nama Variable	Perubahan nilai n	Nilai
n	$n > 0$	True
n	$n = 0$	False

Perubahan nilai variable dan fakto adalah sebagai berikut.

Proses Ke	n	fakto
1.	3	3
2.	2	6
3.	1	6

Gambar 6. 6 Tabel Kebenaran Perhitungan Faktorial.

e. Klik **Toolbar Menu Flowchart** pada samping sebelah kiri.



Gambar 6. 7 Flowchart Perulangan Perhitungan Faktorial.

Rangkuman

- ❖ Multiplikator adalah proses perkalian berulang secara berurutan, hasil perkalian dikalikan dengan dengan berikutnya.
- ❖ Factorial adalah proses perkalian berulang secara berurutan, hasil perkalian dikalikan dengan bilangan berikutnya, dengan nilai bilangan otomatis, dimulai dari nol sampai dengan n (sesuai kebutuhan).

Latihan Soal

Pada sebuah toko swalayan, Ani menemukan promo baru yang ditawarkan oleh pihak swalayan. Promo tersebut diberlakukan untuk pembelian shampoo dengan jumlah genap, minimal pembelian 2 botol shampoo akan diberikan gratis 1 botol shampoo yang sama. Promo ini berlaku untuk kelipatan. Buatlah daftar untuk mencatat berapa jumlah shampoo gratis yang harus diberikan kepada customer apabila dibatasi pembelian shampoo maksimal adalah 10 botol.

2 botol	4 botol	6 botol	8 botol	10 botol	12 botol	14 botol	16 botol	18 botol	20 botol
1 botol	2 botol	3 botol	4 botol	5 botol	6 botol	7 botol	8 botol	9 botol	10 botol

Buatlah Flowchart dan Pseudocode untuk permasalahan diatas menggunakan konsep Proses Perulangan.

MODUL 7

MEMBUAT PSEUDOCODE DARI FLOWCHART DAN VARIABEL ARRAY 1 DIMENSI

Sub Kompetensi

- Mahasiswa dapat memberikan contoh pengembangan algoritma menggunakan Pseudocode.

Tujuan Pembelajaran

- Mahasiswa dapat menjelaskan tahapan pembuatan pseudocode untuk semua proses algoritma.
- Mahasiswa dapat menjelaskan struktur penulisan pseudocode untuk Array 1 Dimensi.
- Mahasiswa dapat menjelaskan pembuatan persamaan aljabar secara eksplisit.

Waktu Pelaksanaan Perkuliahan

- Pertemuan Minggu ke-6.

Media yang Digunakan

- Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika – Operasi Matriks (Operasi Penjumlahan dan Operasi Pengurangan).

Aturan Penulisan Pseudocode

Pseudocode didefinisikan sebagai bentuk penyajian algoritma dengan menggunakan struktur bahasa tertentu. Pseudocode berasal dari kata *pseudo* yang berarti imitasi/mirip/menyerupai. Sedangkan *code* berarti kode program. Dari asal kata tersebut, dapat diartikan kode yang mirip dengan kode pemrograman

sebenarnya. Kode program yang paling sering digunakan dalam penulisan pseudocode adalah struktur bahasa pascal dan C.

Contoh algoritma dalam struktur Bahasa Indonesia :

Baca data nilai_tugas dan nilai_ujian

Hitung total adalah nilai_tugas ditambah nilai_ujian

Tampilkan total

Pseudocode dengan Pascal :

Read (nilai_tugas, nilai_ujian)

total := nilai_tugas + nilai_ujian

Write total

Pseudocode dengan C/Java :

Input (nilai_tugas, nilai_ujian)

total = nilai_tugas + nilai_ujian

Print total

Judul : (Judul_Algoritma)

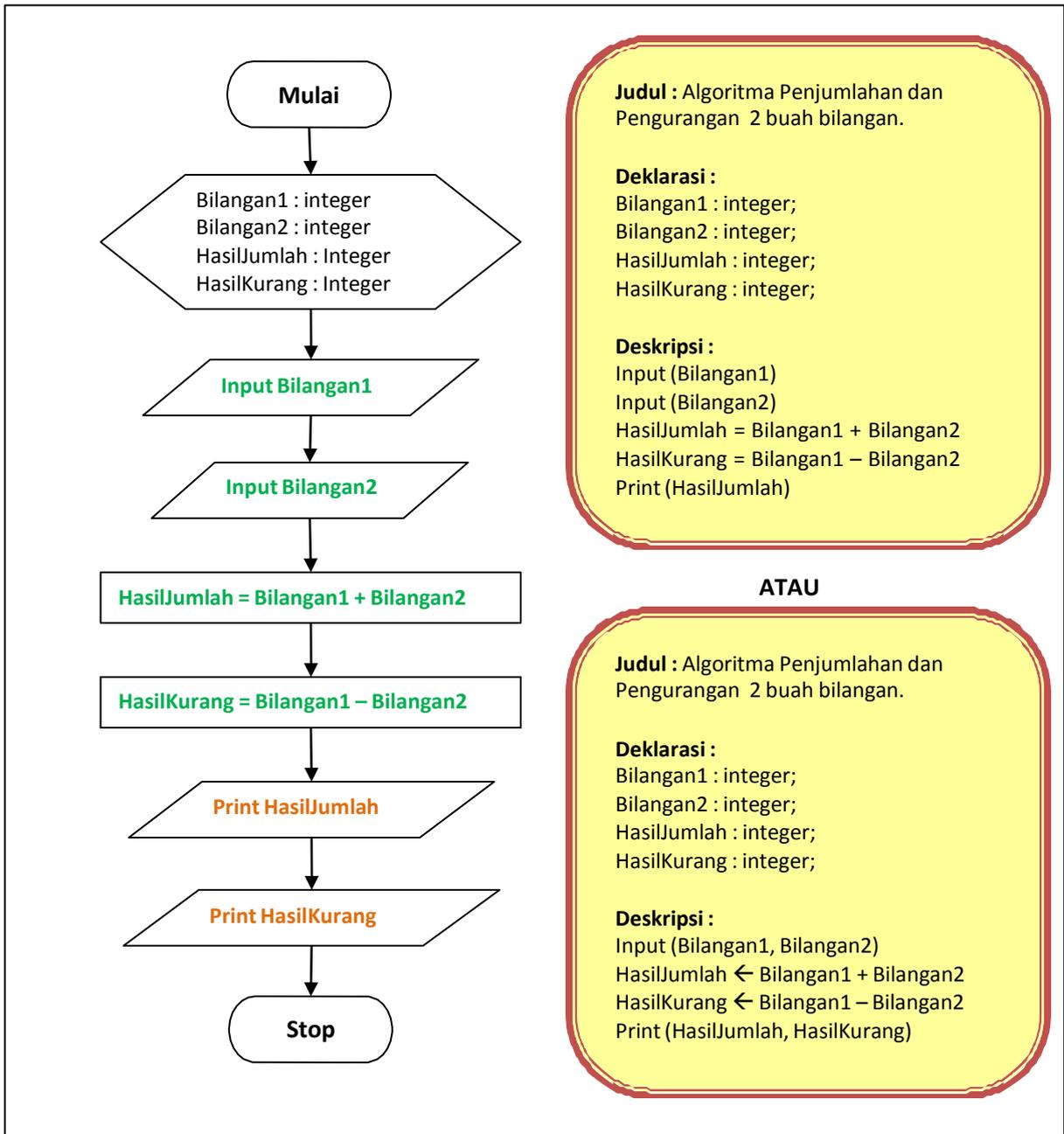
Deklarasi :

(Tulis semua variabel dan konstanta yang dibutuhkan, termasuk Tipe data variabel dan konstanta)

Deskripsi :

(Tulis setiap tahapan/proses yang dilakukan di dalam algoritma, mulai dari awal sampai akhir. Sesuaikan penulisan deskripsi dengan apa yang ditulis di dalam symbol flowchart atau sesuaikan dengan penulisan notasi algoritmik)

Gambar 7. 1 Struktur Penulisan Pseducode



Gambar 7. 2 Penulisan Pseudocode dengan Notasi Algoritmik pada Proses Sekuensial untuk Proses Penjumlahan dan Pengurangan 2 buah bilangan.

Variabel Array 1 Dimensi

Operasi matriks yang dipelajari adalah operasi dasar, meliputi: penjumlahan dan pengurangan. Sedangkan operasi perkalian tidak dibahas karena penekanan pada materi ini adalah konsep variable array, khususnya array 1 dimensi.

Konsep variable array penting untuk dipahami karena dalam proses komputasi dikenal dua klasifikasi variable, yaitu sederhana dan tidak. Variable sederhana sudah

dibahas pada contoh-contoh sebelumnya. Variabel sederhana adalah variable yang hanya dapat memuat satu data pada saat yang bersamaan. Sedangkan variable array adalah variable yang dapat memuat lebih dari satu data. Diskusi pada contoh ini meliputi penggunaan variabel array satu dimensi (diberi symbol 1-D).

Untuk mengenalkan konsep variable array 1-D, pada pembahasan diawali dengan review tentang matriks baris dan kolom, beserta operasi penjumlahan dan pengurangan terhadap matriks tersebut. Matriks baris adalah matriks yang berukuran 1 baris dan n kolom. Diberi notasi huruf besar diikuti dengan indeks jumlah baris kali jumlah kolom (misal $A_{1 \times n}$). Matriks kolom adalah matriks yang berukuran m baris dan 1 kolom (misal $B_{m \times 1}$).

Berikut ini adalah beberapa contoh matriks baris dan kolom.

1) Contoh matriks baris.

$$a). A_{1 \times 3} = [1 \quad 0 \quad 20]; b) B_{1 \times 5} = [2 \quad -5 \quad 11 \quad 3 \quad 0]$$

2) Contoh matriks kolom.

$$a). A_{2 \times 1} = \begin{bmatrix} 5 \\ -1 \end{bmatrix}; b). D_{4 \times 1} = \begin{bmatrix} 0 \\ 4 \\ 1 \\ 1 \end{bmatrix}$$

7

Hal penting yang harus dipahami oleh mahasiswa pada operasi penjumlahan dan pengurangan matriks adalah, bahwa dua buah atau lebih matriks dapat dijumlahkan atau dikurangkan jika ukuran kedua matriks tersebut sama. Beberapa contoh berikut ini dapat dijadikan landasan untuk memahami operasi penjumlahan dan pengurangan pada matriks.

- 1) Matriks $A_{1 \times 4}$ bisa dijumlah dengan matriks $B_{1 \times 4}$, hasilnya adalah matriks lain yang berukuran juga 1×4 . Notasi: $C_{1 \times 4} = A_{1 \times 4} + B_{1 \times 4}$.
- 2) Matriks $A_{1 \times 4}$ **tidak bisa** dijumlah dengan matriks $B_{1 \times 5}$.
- 3) Matriks $A_{3 \times 1}$ bisa dikurangkan dengan matriks $B_{3 \times 1}$, hasilnya adalah matriks lain yang berukuran juga 3×1 . Notasi: $C_{3 \times 1} = A_{3 \times 1} + B_{3 \times 1}$.
- 4) Matriks $A_{2 \times 1}$ **tidak bisa** dikurangkan dengan matriks $B_{3 \times 1}$

Contoh implementasi Variabel Array 1D pada Aplikasi Belajar Dasar-dasar Algoritma Komputasi : Menu Pernyataan Matematika – Operasi Matriks (Operasi Penjumlahan dan Pengurangan Matriks).

- a. Anda dapat memulai untuk memilih dan **klik pada Menu Operasi Matriks (Penjumlahan Matriks)**.

Penjumlahan Matriks (Baris)

BANYAKNYA KOLOM MATRIKS 1: Tampilkan

A _{1x3} EDIT

BANYAKNYA KOLOM MATRIKS 2: Tampilkan

B _{1x3} EDIT

C _{1x3} Tampilkan Hasil

- b. Klik **Toolbar Menu Variabel** pada samping sebelah kiri.

ADA 3 VARIABEL ARRAY YANG TERLIHAT DILAYAR, YAITU
 (1) A[1..4]: VARIABEL A DENGAN PANJANG ARRAY 4.
 (2) B[1..4]: VARIABEL B DENGAN PANJANG ARRAY 4.
 (3) C[1..4]: VARIABEL C DENGAN PANJANG ARRAY 4.

ADA 2 VARIABEL SEDERHANA YANG TERLIHAT YAITU:
 banyakKol_1 dan banyakKol_2

ADA 5 VARIABEL BOOLEAN YANG TERLIHAT YAITU:
 tampil_1, tampil_2, simpan_1, simpan_2, dan tampilkanHasil.

PIKIRKAN VARIABEL YANG TIDAK TERLIHAT DI LAYAR KARENA HANYA DIGUNAKAN UNTUK PROSES

- c. Klik **Toolbar Menu Operator** pada samping sebelah kiri.

OPERATOR YANG DIGUNAKAN ADALAH : + DAN = ;

Digunakan untuk menghitung operasi penjumlahan dua buah matriks berikut ini.

$C[i] = A[i]+B[i]$; dengan i adalah indeks variable array

- d. Klik **Toolbar Menu Penalaran** pada samping sebelah kiri.

Model Penalaran yang digunakan pada Perhitungan Penjumlahan Matriks Baris dapat dilihat pada Gambar 7.3. Proses penjumlahan matriks baris pada intinya menggunakan Proses Perulangan. Hal ini dikarenakan setiap nilai yang berada pada matriks mempunyai indeks. Indeks inilah yang akan berulang dan proses penjumlahan berjalan mulai dari nilai awal baris hingga nilai akhir baris pada matriks tersebut. Nilai pertama pada Matriks A akan ditambahkan dengan nilai pertama Matriks B. Nilai kedua pada Matriks B akan ditambahkan dengan nilai kedua pada Matriks B. Hal ini akan terus berulang hingga nilai yang terakhir.

```

Operasi Penjumlahan Matriks

PENALARAN YANG DIGUNAKAN ADALAH PENALARAN INDUKSI SEBAB AKIBAT BERANTAI

BENTUK PENALARAN (untuk matriks A)
JIKA TAMPIL_1 = TRUE
    MAKA JIKA (banyakKol_1 = 2)MAKA (PRINT matriks_Arr1_2)
    ELSE JIKA (banyakKol_1 = 3)
        MAKA (PRINT matriks_Arr1_3)
        ELSE (PRINT matriks_Arr1_4);
JIKA SIMPAN_1=TRUE
    FOR i=1 TO banyakKol_1 DO
        JIKA (banyakKol_1 = 2) MAKA A[i]=matriks_Arr1_2[i]
        ELSE JIKA (banyakKol_1 = 3)
            MAKA A[i]=matriks_Arr1_2[i]
            ELSE A[i]=matriks_Arr1_2[i];
Catatan: penalaran untuk matriks B sama dengan matriks A ⇒ buatlah

```

```

PENALARAN PENJUMLAHAN.
JIKA banyakKol_1 = banyakKol_2 MAKA
    FOR i=1 TO banyakKol_1 DO C[i]=A[i]+B[i]
ELSE
    PRINT "Matriks A tidak bisa dijumlahkan dengan matriks B".

PENALARAN MENAMPILKAN HASIL
JIKA tampilkanHasil = TRUE MAKA
    FOR i=1 TO banyakKol_1 DO PRINT C[i]

```

Gambar 7. 3 Model Penalaran Perhitungan Penjumlahan Matriks Baris.

Setelah belajar penjumlahan matriks baris, sekarang saatnya belajar pengurangan matriks baris. Pada intinya, kedua proses tersebut sama persis menggunakan proses perulangan. Hanya yang membedakan adalah operator aritmatika yang digunakan.

- a. Anda dapat memulai untuk memilih dan **klik pada Menu Operasi Matriks (Pengurangan Matriks)**.

Pengurangan Matriks (Baris)

BANYAKNYA KOLOM MATRIKS 1: Tampilkan

A _{1x3}: EDIT

BANYAKNYA KOLOM MATRIKS 2: Tampilkan

B _{1x3}: EDIT

C _{1x3}: Tampilkan Hasil

Gambar 7. 4 Aplikasi Dasar Algoritma – Perhitungan Pengurangan Matriks Baris.

b. Klik **Toolbar Menu Variabel** pada samping sebelah kiri.

ADA 3 VARIABEL ARRAY YANG TERLIHAT DILAYAR, YAITU
(1) A[1..4]: VARIABEL A DENGAN PANJANG ARRAY 4.
(2) B[1..4]: VARIABEL B DENGAN PANJANG ARRAY 4.
(3) C[1..4]: VARIABEL C DENGAN PANJANG ARRAY 4.

ADA 2 VARIABEL SEDERHANA YANG TERLIHAT YAITU:
banyakKol_1 dan banyakKol_2

ADA 5 VARIABEL BOOLEAN YANG TERLIHAT YAITU:
tampil_1, tampil_2, simpan_1, simpan_2, dan tampilkanHasil.

PIKIRKAN VARIABEL YANG TIDAK TERLIHAT DI LAYAR KARENA HANYA DIGUNAKAN UNTUK PROSES

c. Klik **Toolbar Menu Data** pada samping sebelah kiri.

ADA 16 DATA YANG TERLIHAT DILAYAR, YAITU :
(1) DATA DARI VARIABEL A[1..4]
[1][2][2][1]
(2) DATA DARI VARIABEL B[1..4]
[0][0][1][0]
(3) DATA DARI VARIABEL C[1..4]
[1][2][1][1]
(4) banyakKol_1 = 4 dan banyakKol_2 = 4

ADA 5 DATA BOOLEAN YANG TIDAK TERLIHAT YAITU :
tampil_1, tampil_2, simpan_1, simpan_2, dan tampilkanHasil.

PIKIRKAN DATA YANG TIDAK TERLIHAT DI LAYAR, KARENA HANYA DIGUNAKAN UNTUK PROSES.

d. Klik **Toolbar Menu Operator** pada samping sebelah kiri.

OPERATOR YANG DIGUNAKAN ADALAH : - DAN = ;

Digunakan untuk menghitung operasi penjumlahan dua buah matriks berikut ini.

$C[i] = A[i] - B[i]$; dengan i adalah indeks variable array

e. Klik **Toolbar Menu Penalaran** pada samping sebelah kiri.

Proses pengurangan matriks baris pada intinya menggunakan Proses Perulangan. Hal ini dikarenakan setiap nilai yang berada pada matriks mempunyai indeks. Indeks inilah yang akan berulang dan proses pengurangan berjalan mulai dari nilai awal baris hingga nilai akhir baris pada matriks tersebut. Nilai pertama pada Matriks A akan dikurangi dengan nilai pertama Matriks B. Nilai kedua pada Matriks B

akan dikurangi dengan nilai kedua pada Matriks B. Hal ini akan terus berulang hingga nilai yang terakhir. Model Penalaran yang digunakan pada Perhitungan Matriks Baris dapat dilihat pada Gambar 7.5.

```

Operasi Penjumlahan Matriks

PENALARAN YANG DIGUNAKAN ADALAH PENALARAN INDUKSI SEBAB AKIBAT BERANTAI

BENTUK PENALARAN (untuk matriks A)
JIKA TAMPIL_1 = TRUE
    MAKA JIKA (banyakKol_1 = 2)MAKA (PRINT matriks_Arr1_2)
        ELSE JIKA (banyakKol_1 = 3)
            MAKA (PRINT matriks_Arr1_3)
            ELSE (PRINT matriks_Arr1_4);
JIKA SIMPAN_1=TRUE
    FOR i=1 TO banyakKol_1 DO
        JIKA (banyakKol_1 = 2) MAKA A[i]=matriks_Arr1_2[i]
            ELSE JIKA (banyakKol_1 = 3)
                MAKA A[i]=matriks_Arr1_2[i]
                ELSE A[i]=matriks_Arr1_2[i];
Catatan: penalaran untuk matriks B sama dengan matriks A ⇒ buatlah

PENALARAN PENGURANGAN.
JIKA banyakKol_1 = banyakKol_2 MAKA
    FOR i=1 TO banyakKol_1 DO C[i]=A[i] - B[i]
ELSE
    PRINT "MATRIKS A TIDAK BISA DIKURANGKAN DENGAN MATRIKS B".

PENALARAN MENAMPILKAN HASIL
JIKA tampilkanHasil = TRUE MAKA
    FOR i=1 TO banyakKol_1 DO PRINT C[i]

```

Gambar 7. 5 Model Penalaran Pengurangan Matriks Baris.

Rangkuman

- ❖ Pseudocode didefinisikan sebagai bentuk penyajian algoritma dengan menggunakan struktur bahasa tertentu.
- ❖ Pseudocode berasal dari kata *pseudo* yang berarti imitasi/mirip/menyerupai. Sedangkan *code* berarti kode program. Dari asal kata tersebut, dapat diartikan kode yang mirip dengan kode pemrograman sebenarnya.
- ❖ Variabel array adalah variabel yang dapat memuat lebih dari satu data.
- ❖ Matriks baris adalah matriks yang berukuran 1 baris dan n kolom.

Latihan Soal

Kerjakan studi kasus di bawah ini menggunakan konsep Array 1D :

- Ibu menyuruh Ani untuk berbelanja sejumlah barang di supermarket. Setibanya di supermarket, ternyata sedang diadakan diskon untuk semua jenis barang. Diskon yang diberikan setiap barang berbeda dengan barang lainnya. Berdasarkan catatan yang dititipkan ibu kepada Ani, Ani membeli sejumlah barang dengan daftar diskon barang sebagai berikut :
 - Sabun → diskon 5%
 - Beras → diskon 7%
 - Gula dan Minyak Goreng → 10%

Anda diminta untuk membuat flowchart dan pseudocode dengan hasil akhir sebagai berikut :

- a. Daftar barang yang dibeli Ani.
- b. Harga per barang
- c. Diskon per barang
- d. Jumlah beli barang (Qty)
- e. Total per jenis barang
- f. Total Belanja Ani keseluruhan