Nama : Ivana Gabby Lauretta

NIM : 103102400037

Fakultas : Informatika

Program Studi : Sains Data

Mata Kuliah : Pengenalan Pemrograman

Kelas : DS – 04 – 02

**RANGKUMAN MATA KULIAH PENGENALAN PEMROGRAMAN**

**Pengantar Pemrograman, Komputer, dan Algoritma**

* **Model Computer dan RAM**
* Piranti input

Perangkat yang digunakan computer untuk menerima data dari pengguna.

* Memori

Untuk penyimpanan data yang akan atau dihasilkan dari suatu proses.

* Processor

Eksekutor setiap perintah atau instruksi yang diberikan melalui suatu program computer.

* Piranti output

Perangkat yang digunakan untuk menampilkan data atau informasi.

* Program computer

Kumpulan instruksi atau tugas yang dieksekusi oleh computer.

* Pemrograman adalah proses membangun sebuah program computer yang dapat dieksekusi oleh computer untuk tugas tertentu.
* Program computer yaitu program computer berisi Kumpulan instruksi untuk memanfaatkan sumber daya yang ada pada computer seperti keyboard, monitor, kamera, printer, hardisk, CPU, GPU, dan lain sebagainya untuk melaksanakan tugas tertentu.
* Bahasa pemrograman adalah media berkomunikasi dengan computer yang terdiri dari *lexis* (kosakata) serta *syntax* (aturan) penyusunan kata-kata tersebut untuk menjadi sebuah *instruction* (kalimat) yang valid.
* Belajar memrogram adalah belajar tentang strategi pemecahan masalah metodologi dan sistematika pemecahan masalah tersebut kemudian menuangkannya dalam suatu notasi yang disepakati bersama. Belajar memrogram bersifat pemahaman persoalan, analisis, dan sintesis.
* Belajar bahasa pemrograman adalah belajar memakai suatu bahasa berdasarkan tata bahasa dan memanfaatkan instruksi-instruksi yang dapat dipakai. Bahasa pemgrograman bersifat keterampilan.
* **Algoritma**

Algoritma adalah urutan instruksi yang jika diikuti dapat digunakan untuk mencapai suatu tujuan tertentu atau menyelesaikan suatu masalah.

Example:

Soal

Terdapat sebuah kotak berwarna hitam di atas meja, sehingga kita tidak bisa melihat isi dari kotak. Salah satu sisinya terdapat lubang seukuran pergelangan tangan untuk bisa mengambil benda yang ada di dalamnya. Di dalam kotak terdapat tiga buah bola berukuran sama, dan masing-masing bola terdapat label dengan angka tertentu.

Tuliskan Langkah-langkah untuk Menyusun tiga bola tersebut terurut berdasarkan label angkanya di luar kotak.

Asumsi: setiap pengambilan hanya boleh satu bola. Diperbolehkan memberi asumsi lain yang dirasa perlu.

Jawaban

Langkah-langkah

1. Ambil Bola Pertama

* Masukkan tangan ke dalam lubang dan ambil satu bola secara acak dari dalam kotak.
* Catat angka yang tertera pada bola tersebut.

1. Ambil Bola Kedua

* Keluarkan bola pertama dari kotak dan letakkan di samping kotak.
* Masukkan tangan ke dalam kotak lagi dan ambil bola kedua secara acak.
* Catat angka pada bola kedua.

1. Bandingkan Angka

* Sekarang, kita memiliki dua bola dengan angka yang berbeda. Bandingkan kedua angka tersebut.
* Jika angka pada bola kedua lebih kecil dari angka pada bola pertama, simpan bola kedua di depan bola pertama. Jika tidak, simpan bola pertama di depan bola kedua.

1. Ambil Bola Ketiga

* Keluarkan bola yang sudah disusun dari kotak (yang memiliki dua bola) dan letakkan di samping kotak.
* Masukkan tangan ke dalam kotak dan ambil bola ketiga.
* Catat angka pada bola ketiga.

1. Sisipkan Bola Ketiga

* Bandingkan angka pada bola ketiga dengan dua bola yang sudah disusun.
* Jika angka pada bola ketiga lebih kecil dari angka pada bola terkecil, letakkan bola ketiga di depan kedua bola.
* Jika angka pada bola ketiga lebih besar dari angka pada bola terbesar, letakkan bola ketiga di belakang kedua bola.
* Jika angka pada bola ketiga berada di antara dua angka yang sudah ada, sisipkan bola ketiga di antara kedua bola tersebut sesuai urutan.

1. Selesai

* Setelah semua bola dikeluarkan dan disusun, kita akan memiliki tiga bola yang terurut berdasarkan label angkanya.
* **Notasi Algoritma**

**Pseudocode** adalah notasi yang disepakati untuk menuliskan algoritma sehingga mudah dibaca, dipahami dan diimplementasikan menjadi kode program.

* Penulisan algoritma terdiri dari 3 bagian, yaitu :

1. Judul
2. Deklarasi kamus
3. Instruksi algoritma

A screenshot of a computer program

Description automatically generatedExample:

* Aturan pemberian NAMA

1. Dalam bahasa pemrograman, setiap nama mempunyai aturan penulisan. Pada teks algoritma tidak ada aturan ketat.
2. Pemilihan nama harus interpretative, tidak menimbulkan kerancuan, jika singkat ahrus disertai penjelasan.

Example: x-y akan membingungkan sebab mungkin “x minus y”

1. Nama merupakan satu kesatuan leksikal, maka sebuah nama harus dituliskan secara utuh (tidak dipisah blank) supaya satu nama dapat dibedakan dengan nama lainnya (besaran leksikal lain).

**Variable, Input Output, Assignment, Tipe Data dan Operasi**

* **Instruksi Primitif – Assignment**
* Variable adalah suatu Lokasi memori yang telah diberi nama
* Assignment adalah instruksi menyalin data ke dalam variable
* Notasi assignment adalah < variable > ⬅ < value >

Dimana < variable > adalah nama dari suatu variable dan **value** adalah nilai yang akan disimpan ke lokasi tersebut.

Contoh:

Data ⬅ 43

Total ⬅ total + 37

A diagram of a diagram

Description automatically generated

* **Instruksi Primitif Input/Output**

Instruksi input : mengambil data dari piranti masukan dan menyimpannya ke suatu lokasi memori.

input(<data variable>)

Instruksi output : mengirim data ke piranti keluaran.

output(daftar data)

Variasi kata kunci : print

Example

input(abc)

output(“Selamat Datang”, abc\*3)

* **Tipe Data dan Variable**

Tipe data merupakan pengelompokan data dalam pemrograman computer sehingga memungkinkan dilakukan proses komputasi.

* Tipe data dasar:

1. Integer : bilangan bulat. Contoh : 1, 1234, -45.
2. Boolean : representasi logika. Contoh : true, false.
3. Real : bilangan berkoma. Contoh : 3,14.
4. String : teks. Contoh : CII-1A3 Pengenalan Pemrograman.
5. Character : himpunan symbol yang terdefinisi oleh suatu enumerasi. Contoh : “a” direpresentasikan oleh bilangan 97 pada ASCII.

Untuk menggunakan suatu tipe data, harus disediakan yang namanya variable dengan tipe data yang dimaksud.

* **Variable**

Variable merupakan suatu lokasi di memori computer yang telah diberi nama.

1. Nama Variable

Dimulai dengan huruf dan dapat diikuti dengan sejumlah huruf, angka, atau garis bawah. Contoh: ketemu, found, rerata, mhs1, data2.

1. Tipe Data Variable

Data yang tersimpan dalam suatu variable dapat mempunyai tipe apa saja selama diperkenankan oleh bahasa pemrograman yang digunakan.

Tipe data yang umum tersedia adalah integer, real atau float, boolean, karakter, string, dan alamat memori.

1. Akses Nilai Variable

Dengan menyebutkan nama variable, nilai data yang tersimpan dalam variable tersebut dapat diambil.

1. Akses Lokasi atau Alamat Variable

Pada Bahasa Pemrograman\*

Lokasi atau informasi alamat dari variable dapat diperoleh dengan memberikan prefix & pada nama variable tersebut.

Example: & ketemu akan menghasilkan alamat memori dari variable ketemu.

* **Deklarasi Variable**
* Variable harus dideklarasikan dulu sebelum digunakan.
* Variable juga harus diinisialisasi dulu (diisi data) agar nilai yang tersimpan diketahui dengan jelas dan eksekusi algoritma menjadi terprediksi.

kamus

suhu : real

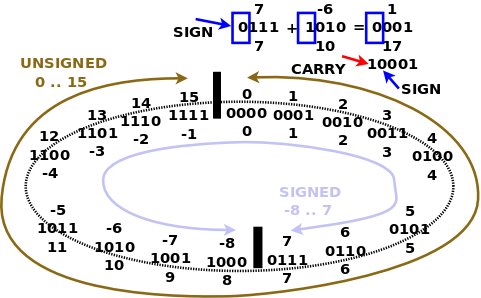
umur : integer

a1, a2, a3, a4 : char

nama\_lengkap, alamat, agama : string

hidup : Boolean

* **Tipe Data Integer**
* Representasi internal : bilangan bulat dan bilangan bulat bukan negative.
* Umumnya operasi dalam operasi aritmatika dan operasi logika pada tingkat bit.



* **Operasi Aritmatika**
* Operator + - \* / %
* Pembagian

Pembagian integer 71 div 21 = 3

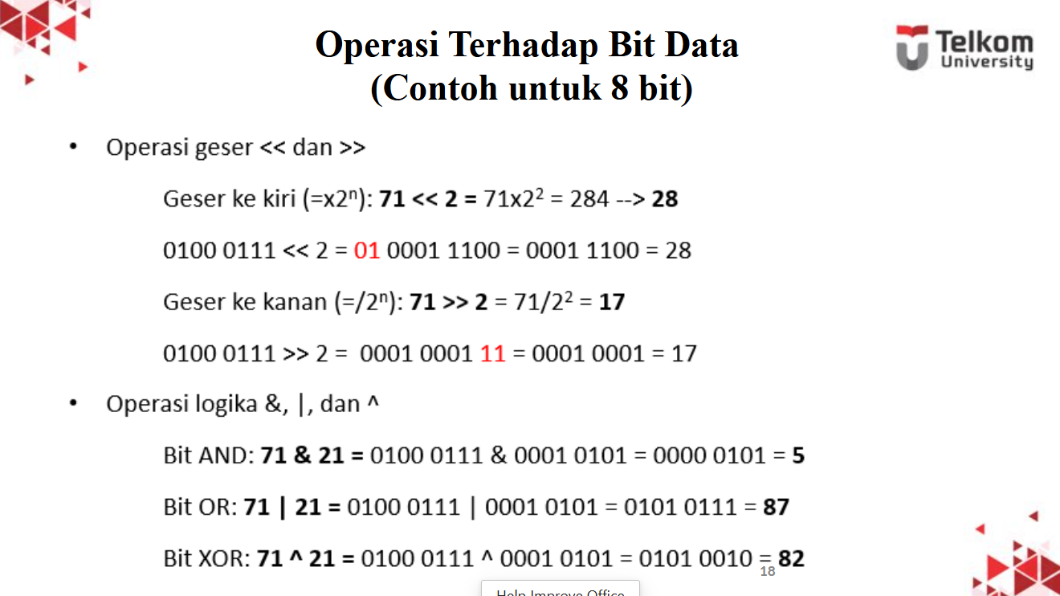
Modulo (sisa pembagian) 71 mod 21 = 71 – (3\*21) = 71 – 63 = 8

* Jika terjadi overflow, maka diambil sebanyak bit yang ada

Perkalian 71\*21 = 1491 ⮕ 211 (sebanyak bit yang ada)

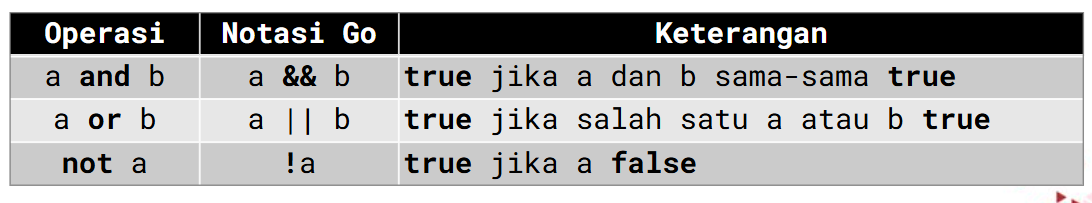
* Konversi basis bilangan digital ke decimal : 0100 0111 = ?

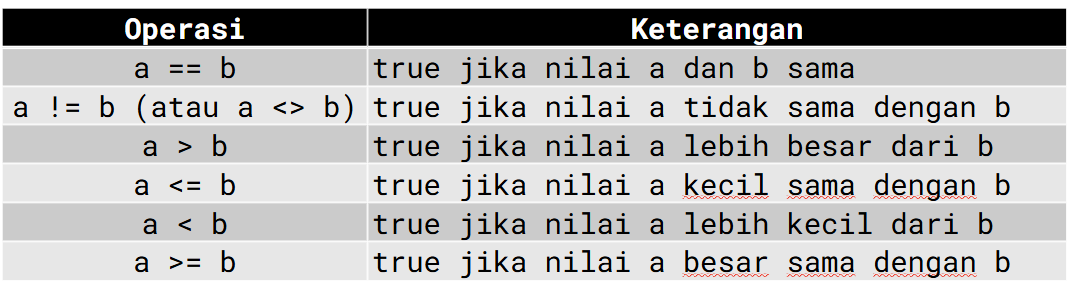


* **Operasi Terhadap Bit Data (contoh untuk 8 bit)**

**Tipe Data dan Analisis Masalah (lanjutan)**

**Tipe Data Boolean, Character, String**

* **Tipe Data Boolean**
* Hanya mempunyai nilai true atau false
* Operasi Boolean : operasi logika dan operasi perbandingan
* Dapat dibaca, ditulis, dan diproses
* Biasanya disimpan dalam 1 word memori (bit 0 berarti false, jika tidak berarti true)
* **Operasi Perbandingan**
* Kedua data dengan tipe yang sama, tetapi bukan tipe Boolean
* Hasil dari operasi perbandingan adalah nilai Boolean, true/false



A screenshot of a computer

Description automatically generated

* **Tipe Data Character**
* Representasi internal : bilangan bulat yang merupakan nomor urut karakter tersebut dalam suatu table/daftar karakter yang diakui.
* Umumnya ada dua table yaitu : table ASCII dan table UTF-16 (penyempurnaan ASCII dan symbol non huruf latin).
* Operasi dasar karakter tidak ada karena secara internal dianggap sebagai bilangan bulat atau integer.
* Pada saat input, pembacaan karakter diubah menjadi nilai integernya.
* Pada saat print, data integer diubah menjadi tampilan karakter.
* **Tipe Data String**

A white text with black text

Description automatically generated

* **Persoalan Komputasi**
* Problem komputasi seringkali berkaitan dengan dunia nyata
* Sekarang banyak sekali aplikasi untuk mendukung kegiatan (online shopping, banking, transportasi, dll. eGoverment, autonomous vehicles, dll)
* Dari sudut pandang informatika, semuanya terkait pemprosesan data (apa inputnya, proses apa yang harus dilakukan, apa yang harus dihasilkan)

**Struktur Kontrol : Perulangan A (For-Loop)**

* **Perulangan**
* Tugas utama computer adalah melakukan suatu instruksi/proses secara berulang dan terus-menerus tanpa adanya perbedaan.
* Hal ini berbeda dengan manusia yang bila melakukan hal sama secara berulang bisa melakukan kesalahan.
* Setiap baris instruksi dieksekusi satu persatu.
* Satu atau lebih instruksi bisa dieksekusi berulang kali.
* **Syarat Perulangan**
* Perulangan harus berhenti
* Apabila perulangan tidak pernah berhenti, maka algoritmanya salah (proses lama ≠ tidak pernah berhenti)
* Pemrograman harus mengetahui perulangan akan berhenti/tidak sebelum program dijalankan
* **Jenis Instruksi Perulangan**
* Berdasarkan jumlah iterasi
* Berdasarkan kondisi (kapan harus diulangi/berhenti)
* **(A) Berdasarkan Jumlah Iterasi**
* Untuk menyelesaikan kasus dengan jumlah iterasi diketahui di awal
* Iterasi adalah variable bertipe integer untuk menampung iterasi perulangan
* a dan b adalah variable atau integer
* Aksi dieksekusi berulang sebanyak b-a+1 kali (iterasi)
* Setiap aksi dieksekusi, nilai dari variable iterasi selalu bertambah 1, dari a hingga b

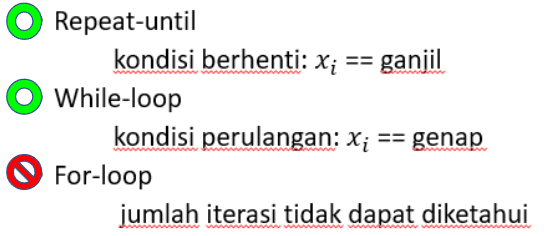
A close-up of a sign

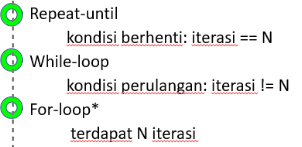
Description automatically generated

A diagram of a process

Description automatically generated

**Struktur Kontrol : Perulangan (dengan While-Loop & Repeat-Until)**

* **Analisis Masalah**
* Input : integer x1, x2, x3, … dst (syarat xi == genap)
* Proses : total = x1 + x2 + x3, … dst (syarat xi == genap)
* Output : total
* **Jenis Perulangan**
* **Analisis Masalah**
* Input : integer N, N > 0
* Proses : perulangan dari 1 sampai N
* Output : 1 2 3 4 … N
* **Jenis Perulangan**

****

* **Analisis Masalah**

Masukan : integer x dan y, x dan y > = 0

Proses : nilai x dikurangi y secara terus-menerus (nilai x selalu diupdate dengan hasil pengurangan) hingga x tidak bisa dikurangi. Banyaknya pengurangan adalah hasil div, sedangkan nilai x yang tidak bisa dikurangi adalah hasil mod. Contoh x = 20 dan y = 6

A number and symbols on a white background

Description automatically generated

Operasi pengurangan dilakukan 3x, sedangkan nilai x terakhir adalah 2

Hasil 20 div 6 adalah 3, sedangkan hasil 20 mod 6 adalah 2

Keluaran : hasil x mod y dan hasil x div y

**Boolean dan Operasinya**

* **Tipe Data Boolean**
* Hanya mempunyai nilai true atau false
* Operasi Boolean : operasi logika dan operasi perbandingan
* Dapat dibaca, ditulis, dan di proses
* Biasanya disimpan dalam 1-word memory (bit 0 berarti false, jika tidak berarti true)
* Umumnya berkaitan dengan statement pengkondisian pada struktur control
* **Operasi Bilangan**
* Tipe data antara operan adalah sama
* Hasil dari operasi pembandingan adalah nilai Boolean, true/false

A table with text on it

Description automatically generated

* **Struktur Kontrol : Percabangan**
* Setiap baris instruksi dieksekusi satu persatu
* Adanya instruksi bersyarat (kondisi)
* Instruksi dieksekusi apabila memenuhi kondisi atau syarat tertentu
* Kontruksi dalam algoritma yang memungkinkan kita untuk melakukan pilihan instruksi yang berbeda-beda sesuai dengan kondisi yang dihadapi