



- Dasar Pemrograman – Pertemuan 3

Tim Bahan Ajar Dasar Pemrograman
Teknik Informatika - S1
Fakultas Ilmu Komputer



Review Materi





Capaian Kuliah Pertemuan 3



Capaian Pembelajaran

- Setelah mengikuti matakuliah ini mahasiswa dapat menjelaskan, membuat, dan mempraktikkan input/output, operasi, dan ekspresi pada pemrograman prosedural dengan diterapkan pada aksi sekuensial permasalahan komputasional sederhana.



INPUT/OUTPUT



Input

- Untuk memasukkan suatu nilai kedalam variabel biasanya digunakan assignment. Terkadang perlu juga dilakukan pembacaan dari piranti masukan seperti keyboard, mouse, dll
- Hal ini akan memungkinkan “seakan-akan” user memberikan suatu nilai yang dikirim ke komputer.
- Mekanisme tersebut disebut dengan input.
- Implementasi notasi algoritma:
 - `input(<list-nama>)`
- List-nama bisa di isi variabel dengan suatu tipe/konstanta/fungsi*/prosedur*

Output



- Suatu nilai yang disimpan dalam memory komputer harus dapat dikomunikasikan keluar, dalam hal ini melalui perangkat output seperti monitor atau printer misalnya.
- Jadi dibutuhkan mekanisme untuk mengeluarkan nilai tersebut (bisa juga dipicu oleh variabel/konstanta/fungsi*),
- Notasi Algoritma:
 - output(<list-nama>)
 - output(<konstanta>)
 - output(<ekspresi>)
 - output(<list-nama>,<konstanta>,<ekspresi>)
- Syarat list-nama, konstanta, ekspresi harus sudah terdefinisi nilainya.



OPERASI



Operasi

- Operasi dalam pemrograman tergantung dari tipe yang definisikan
- Operasi menggunakan berbagai operator yang berfungsi pada tipe tertentu
- Contohnya operasi menggunakan operator + akan memiliki hasil berbeda
 - Dalam tipe data numerik memiliki arti penjumlahan
 - Dalam tipe data “string” memiliki arti penggabungan kata

Operator logika pada tipe Boolean

- Boolean merupakan suatu tipe yang memiliki domain nilai benar atau salah (true or false)
- Diketahui suatu variabel a dan b bernilai Boolean.
 - **not** a \rightarrow bernilai True jika a adalah False, bernilai False jika a adalah true
dalam matematika disebut negasi
 - a **and** b \rightarrow bernilai True jika keduanya True, selain itu jika salah satu salah maka salah
 - a **or** b \rightarrow bernilai True jika salah satu atau keduanya adalah True

a	b	a <u>and</u> b	a <u>or</u> b	<u>not</u> a	<u>not</u> b	a <u>eq</u> b	a <u>neg</u> b	Hasil
T	T	T	T	F	F	T	F	Boolean
T	F	F	T	F	T	F	T	Boolean
F	T	F	T	T	F	F	T	Boolean
F	F	F	F	T	T	T	F	Boolean

Operator Aritmatika

- Simbol yang digunakan untuk melakukan operasi aritmatika pada bilangan **Bulat dan real**
- **+** merupakan operator aritmatika untuk penjumlahan.
 - Contoh: $i + j$, pastikan bahwa i dan j bertipe sama (bulat atau real)
 - ***Jika ini diterapkan pada tipe char atau string maka akan bermakna penggabungan/ concat***
- **-** merupakan operator aritmatika untuk pengurangan.
 - Contoh: $i - j$, pastikan bahwa i dan j bertipe sama (bulat atau real)
- **/** merupakan operator aritmatika untuk pembagian khusus bilangan riil.
 - Contoh: i / j
- **div** merupakan operator aritmatika untuk pembagian khusus bilangan bulat.
 - Contoh: $i \text{ div } j$
- **mod** merupakan operator aritmatika untuk sisa bagi atau modulo.
 - Contoh: $i \text{ mod } j$ (bulat atau real)
- ***** merupakan operator aritmatika untuk perkalian.
 - Contoh: $i * j$ (bulat atau real)
- **^** merupakan operator aritmatika untuk pangkat.
 - Contoh: $i ^ j$

Operator Perbandingan

- Simbol yang digunakan untuk melakukan operasi perbandingan pada bilangan **Bulat dan real**.
- Hasil perbandingan akan bernilai **Boolean**
- Contoh operator perbandingan
 - $a == 10$ → operator untuk menguji kesamaan dua nilai
 - $a \neq 10$ → operator untuk menguji apakah kedua ekspresi berbeda nilainya.
 - $a > 10$ → operator lebih dari
 - $a \geq 10$ → operator lebih dari sama dengan
 - $a < 10$ → operator kurang dari
 - $a \leq 10$ → operator kurang dari sama dengan



EKSPRESI



Ekspresi

- **Ekspresi** bisa disebut juga **rumus perhitungan**.
- Terdiri dari operan dan operator
- Operan harus terdefinisi nilainya dengan suatu tipe tertentu.
- Hasilnya adalah suatu nilai yang sesuai dengan tipe operator yang bersangkutan.
- Contoh:
 - ekspresi $3 + 2$ menunjukkan perhitungan untuk objek 5 dengan tipe bilangan bulat.
 - Ekspresi $3.0 + 2.5$ menunjukkan perhitungan untuk objek 5.5 dengan tipe bilangan real.
- Jenis ekspresi: logika/boolean, numerik, karakter, dan string.

Jenis Ekspresi

- Ekspresi Uner: ekspresi dengan operator yang hanya butuh satu operan
- Ekspresi Biner: ekspresi dengan operator yang membutuhkan dua operan

- Infix: operator ditengah

$$13 * 5$$

$$((3 * 5) + (4 \text{ div } 7)) - (a * b)$$

- Prefix: operator diawal

$$* 13 5$$

$$- + * 3 5 \text{ div } 4 7 * a b \text{ adalah } - ((+ (* 3 5) (\text{div } 4 7)) (* a b))$$

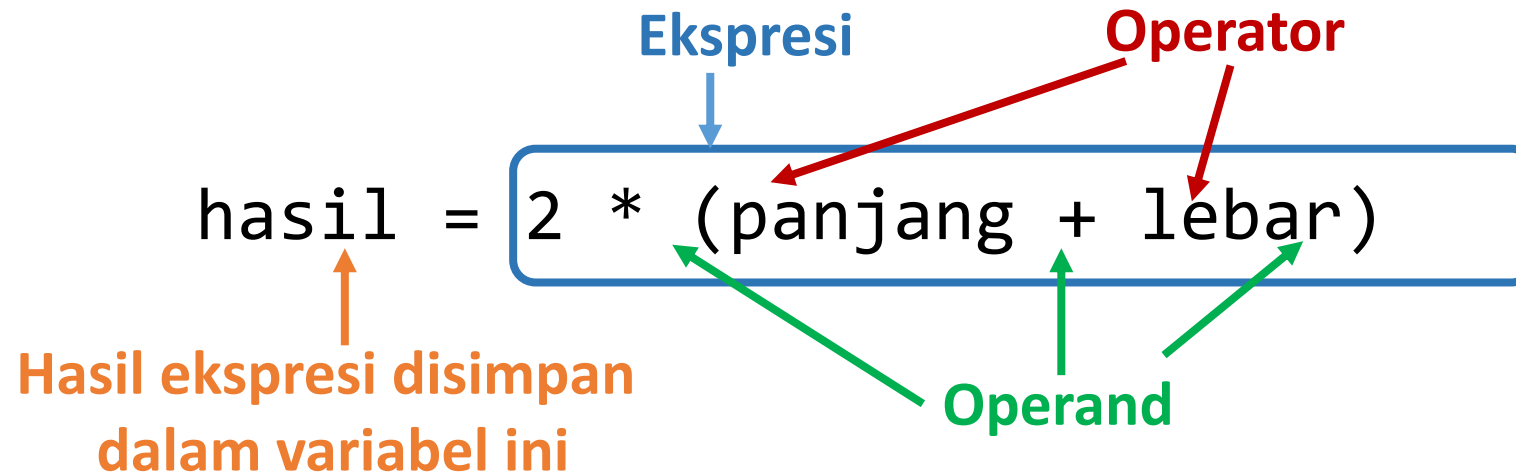
- Sufix: operator di akhir

$$13 5 *$$

$$3 5 * 4 7 \text{ div } a b * + - \text{ adalah } (3 5 *) ((4 7 \text{ div }) (a b *) +) -$$

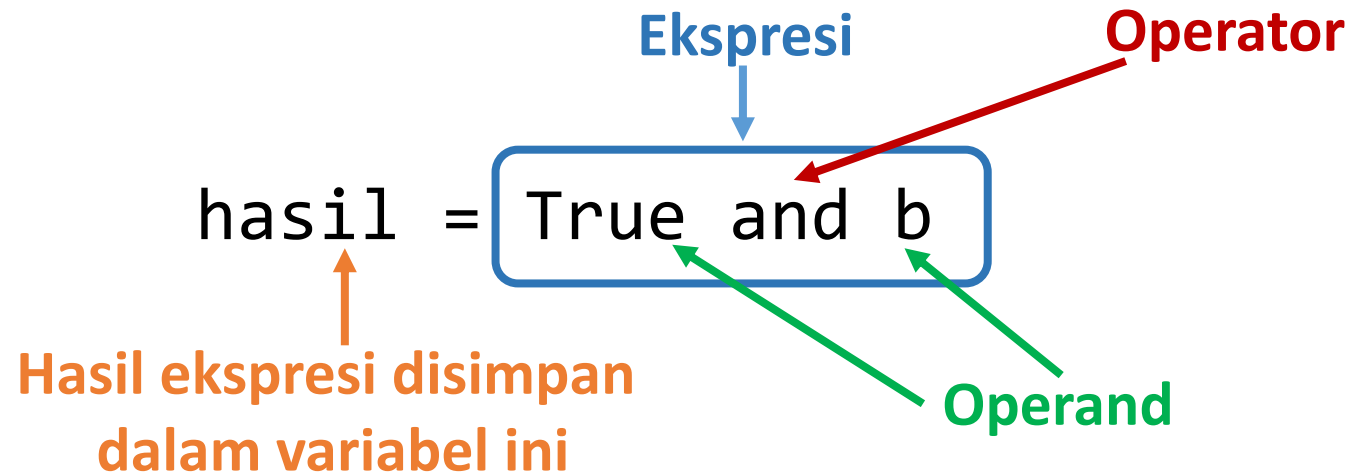
Contoh Ekspresi Karakter

- Algoritma keliling persegi panjang



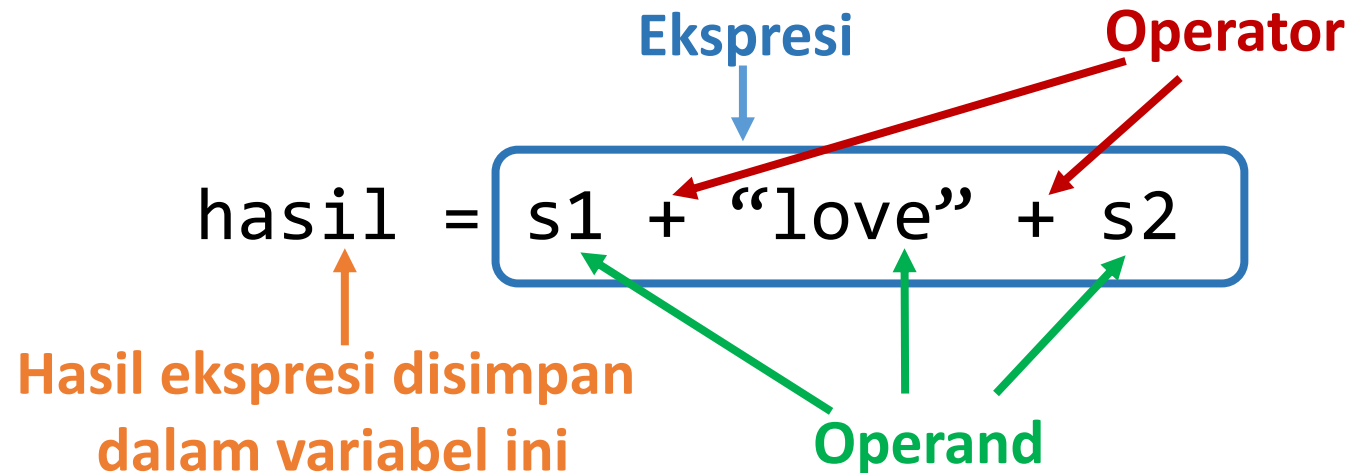
Contoh Ekspresi Boolean

- Pastikan bahwa variabel hasil dan b bertipe Boolean



Contoh Ekspresi Boolean

- Pastikan bahwa variabel hasil, s1, s2 bertipe String



- String merupakan tipe untuk menyimpan suatu nilai berupa kata atau kalimat dengan nilai di apit oleh tanda "<petik>"
- Bentuk satuan dari string adalah tipe char



Notasi Algoritmik untuk Aksi Sekuensial yang memanfaatkan Operator dan Ekspresi



Program Sequential1

Program Sequential1

{Contoh penulisan aksi sekuensial dengan memanfaatkan apa yang sudah dipelajari sebelumnya}

KAMUS

i : integer
x : real
hasil : integer

Initial State

ALGORITMA

input(i)
x \leftarrow 12.5
hasil \leftarrow i * 10
output(i)
output(x+5.4)
output(i,x,hasil)

Final State

Program PersegiPanjang

Program PersegiPanjang

{program untuk menghitung keliling dan luas persegi panjang}

KAMUS

p : real
l : real
hasilKeliling : real
hasilLuas : real

ALGORITMA

input(p)
input(l)
hasilKeliling $\leftarrow 2 * (p+l)$
hasilLuas $\leftarrow p*l$
output(hasilKeliling)
output(hasilLuas)

Referensi



Utama :

1. Liem, Inggriani. Diktat Pemrograman Prosedural Informatika ITB. IF-ITB. 2007
2. Bjarne Stroustrup, 2014, Programming: Principles and Practice Using C++ (Second Edition), Addison-Wesley Professional

Pendukung :

1. Introduction to Computer Science and Programming in Python, MIT
<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016>
2. Introduction to Computer Science and Programming, MIT <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-00sc-introduction-to-computer-science-and-programming-spring-2011/index.htm>