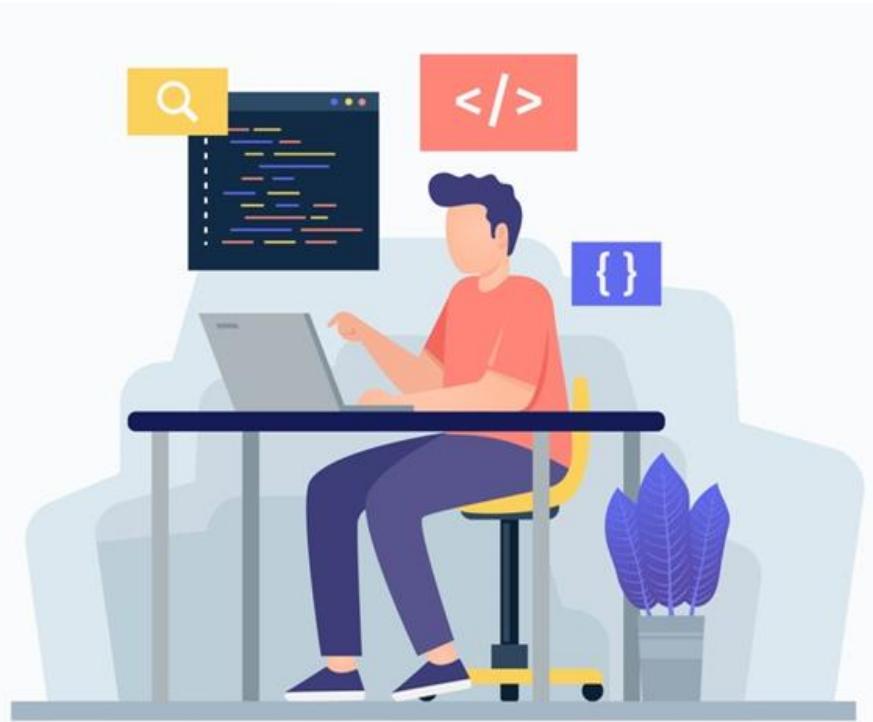




**PROGRAM STUDI
TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO**

Mata Kuliah
Dasar Pemrograman



Input/Output, Operasi, Ekspresi

TIM DASAR PEMROGRAMAN
TEKNIK INFORMATIKA S1
UNIVERSITAS DIAN NUSWANTORO

Capaian Pembelajaran

- Setelah mengikuti kuliah ini mahasiswa dapat menjelaskan dan mempraktekkan input/output, operasi, dan ekspresi pada pemrograman procedural dengan diterapkannya pada aksi sekuensial pada permasalahan komputasional sederhana



FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

Input/Output

Input

- Untuk memasukkan suatu nilai kedalam variable 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 computer
- Mekanisme tersebut disebut dengan input
- Implementasi notasi algoritma:
 - Input(<list nama>)
- List nama bisa diisi variable dengan suatu tipe/konstanta/fungsi*/prosedur*

Output

- Suatu nilai yang disimpan dalam memory computer harus dapat dikomunikasikan keluar, dalam hal ini melalui perangkat output seperti monitor atau printer.
- Jadi dibutuhkan mekanisme untuk mengeluarkan nilai tersebut (bisa juga dipicu oleh variable/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



FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

Operasi

Operasi

- Operasi dalam pemrograman tergantung dari tipe yang didefinisikan
- Operasi menggunakan berbagai operator yang berfungsi pada tipe tertentu
- Contohnya operasi menggunakan + akan memiliki hasil yang 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 variable a dan b bernilai Boolean
 - **not a** → bernilai true jika a adalah false, bernilai false jika a adalah true. Dalam matematika disebut negasi
 - **a and b** → bernilai true jika keduanya true, selain itu jika salah satu salah maka salah
 - **a or b** → bernilai true jika salah satu atau keduanya 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>neq</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 operator 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 bulat
 - Contoh: i / j

Operator Aritmatika

- **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 atau 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 nilai
 - $a > 10$ → operator lebih dari
 - $a \geq 10$ → operator lebih dari sama dengan
 - $a < 10$ → operator kurang dari
 - $a \leq 10$ → operator kurang dari sama dengan



FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

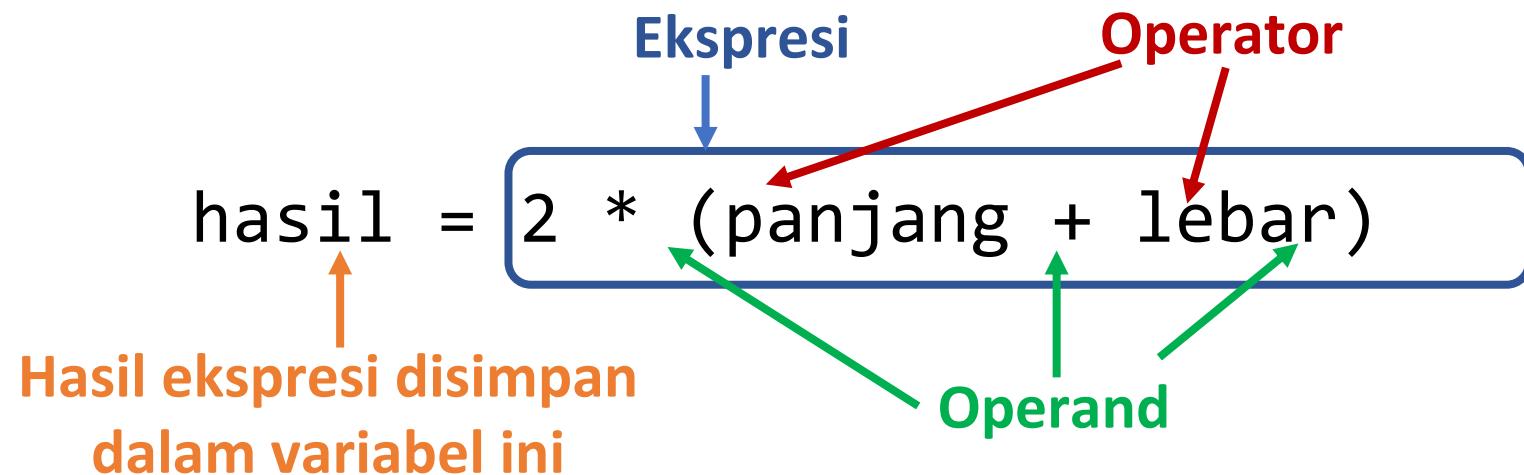
Ekspressi

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

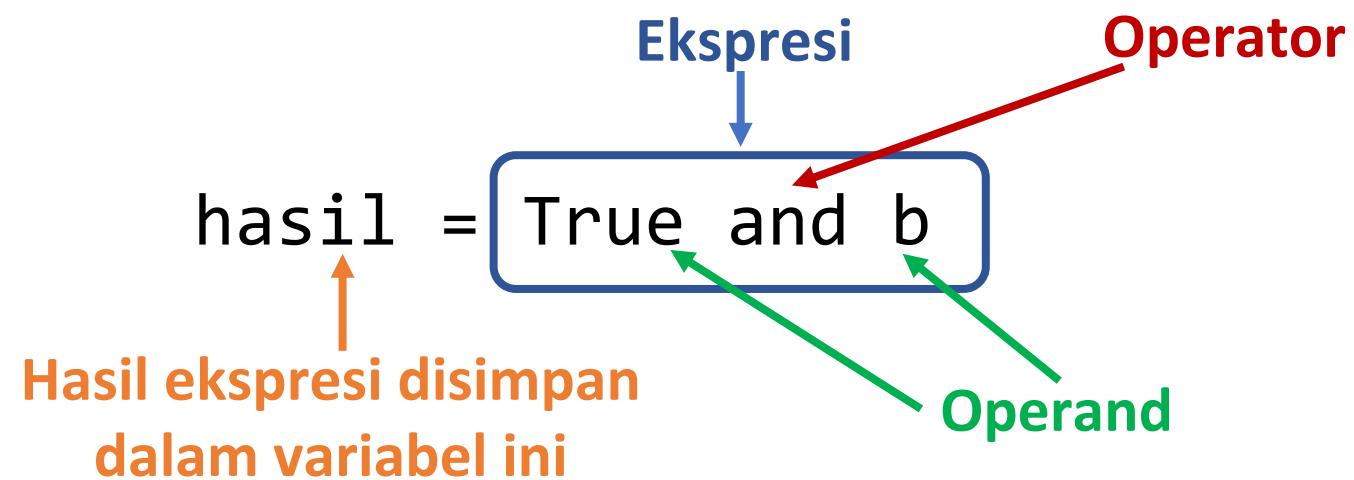
Contoh Ekspresi Numerik

- Algoritma keliling persegi panjang



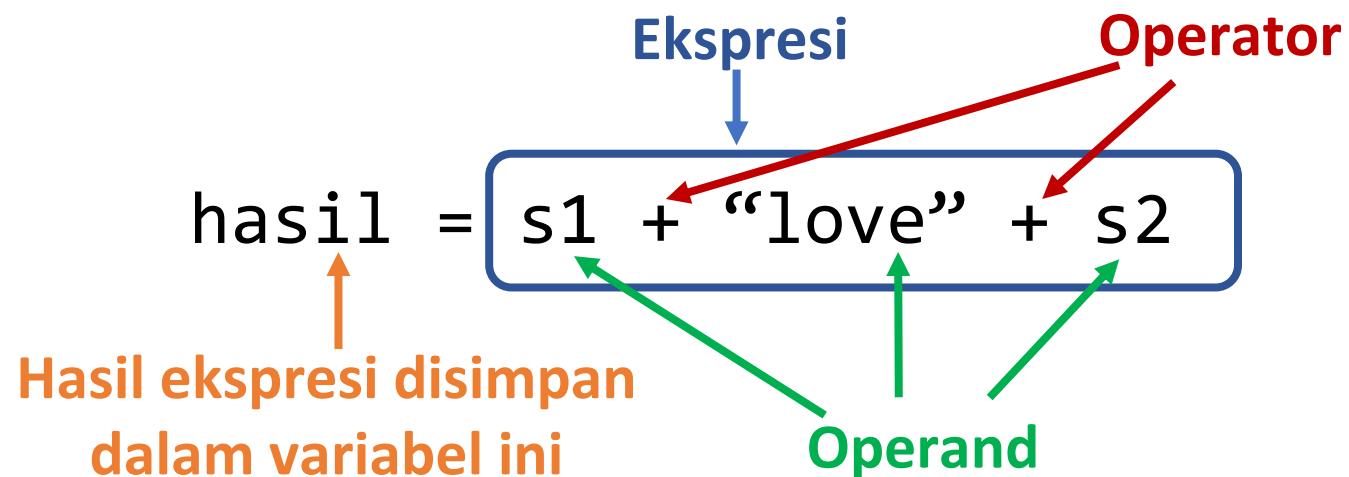
Contoh Ekspresi Boolean

- Pastikan bahwa variable hasil dan b bertipe Boolean



Contoh Ekspresi Karakter

- Pastikan bahwa variable 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



FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

Notasi Algoritmik untuk aksi Sekuensial dengan operator dan ekspresi

Program Sequential 1

Program Sequential1

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

KAMUS

i : integer

x : real

hasil : integer

ALGORITMA

input(i)

x \leftarrow 12.5

hasil \leftarrow i * 10

output(i)

output(x + 5.4)

output(i,x,hasil)

Ribuan Ratusan Puluhan Satuan

- Terdapat sebuah bilangan bulat positif ribuan. Hitung angka ribuan, ratusan, puluhan, dan satuan!
- Contoh n=5713 akan menghasilkan 5, 7, 1, 3 yang artinya 5 ribuan, 7 ratusan, 1 puluhan, 3 satuan
 - Initial State? Bilangan bulat positif ribuan (n)
 - Final State? Ribuan, ratusan, puluhan, dan satuan

```
Masukkan angka ribuan antara 0 sampai 9999 = 5713  
5 ribuan, 7 ratusan, 1 puluhan, dan 3 satuan.
```

Ribuan Ratusan Puluhan Satuan

JUDUL Program Ribuan Ratusan Puluhan Satuan

{Input bilangan bulat positif ribuan}

{Mencari masing-masing 1 angka ribuan, ratusan, puluhan, satuan dan tuliskan hasilnya}

KAMUS

n : integer {angka yang dibaca, 0 sampai 9999}

rib : integer {ribuan, bilangan bulat positif}

rat : integer {ratusan, bilangan bulat positif}

pul : integer {puluhan, bilangan bulat positif}

sat : integer {satuan, bilangan bulat positif}

ALGORITMA

output("Masukkan angka ribuan antara 0 sampai 9999 = ")

input(n)

rib \leftarrow n/1000

rat \leftarrow (n mod 1000) / 100

pul \leftarrow (n mod 100) / 10

sat \leftarrow n mod 10

output(rib + " ribuan, " + rat + " ratusan, " + pul + " puluhan, dan " + sat + " satuan.")

Masukkan angka ribuan antara 0 sampai 9999 = 5713
5 ribuan, 7 ratusan, 1 puluhan, dan 3 satuan.

Flowchart

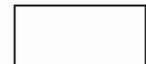
- Gambaran dari langkah-langkah suatu proses



An oval is used to indicate the beginning or end of an algorithm.



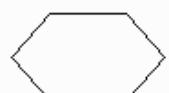
A parallelogram indicates the input or output of information.



A rectangle indicates a computation, with the result of the computation assigned to a variable.



A diamond indicates a point where a decision is made.



A hexagon indicates the beginning of the repetition structure.



A double lined rectangle is used at a point where a subprogram is used.



An arrow indicates the direction of flow of the algorithm. Circles with arrows connect the flowchart between pages.

Flowchart



An oval is used to indicate the beginning or end of an algorithm.



A parallelogram indicates the input or output of information.



A rectangle indicates a computation, with the result of the computation assigned to a variable.



A diamond indicates a point where a decision is made.



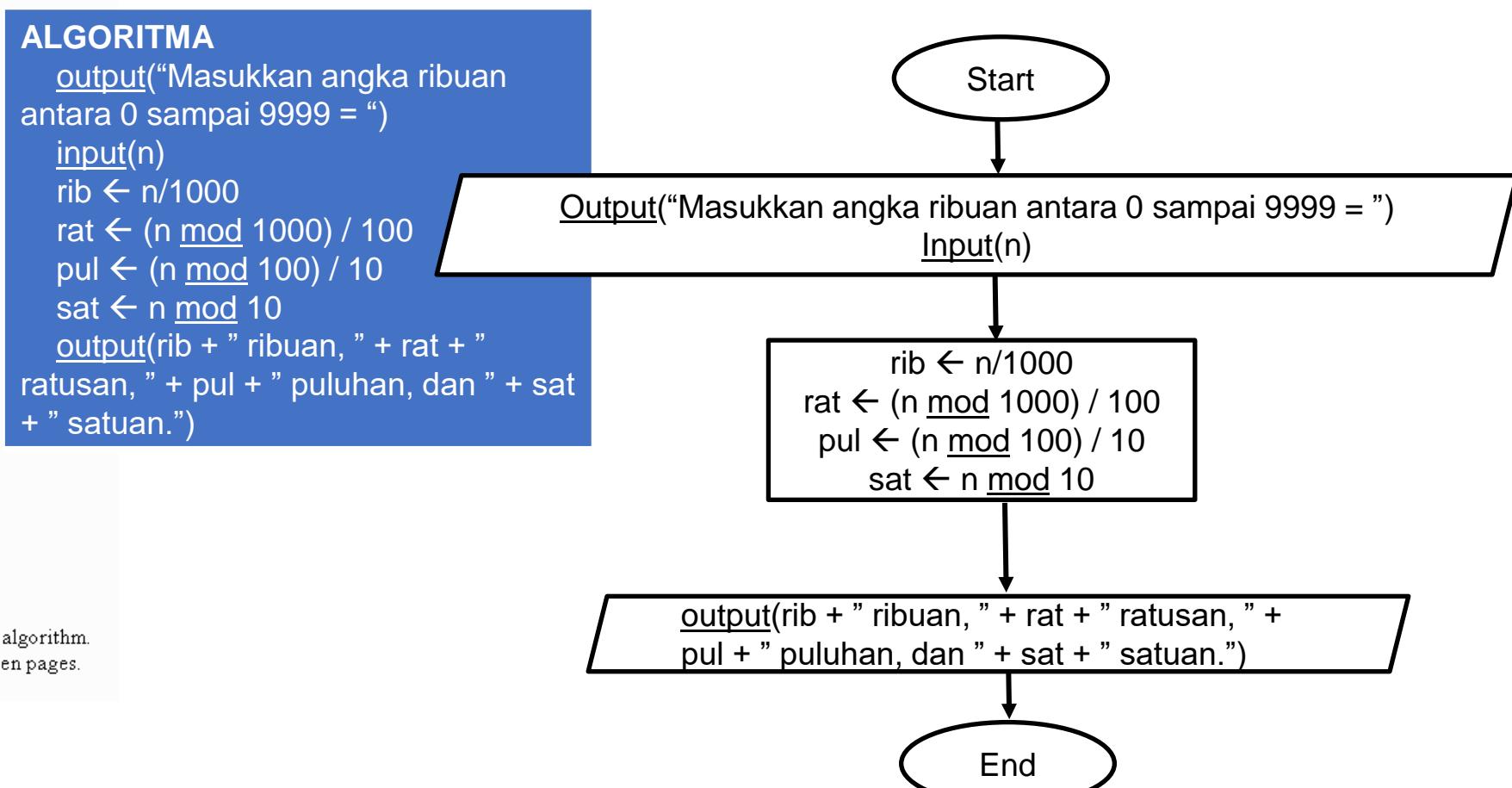
A hexagon indicates the beginning of the repetition structure.



A double lined rectangle is used at a point where a subprogram is used.



An arrow indicates the direction of flow of the algorithm. Circles with arrows connect the flowchart between pages.



Referensi

Utama:

1. 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>



TERIMA KASIH

ANY QUESTIONS?