



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

Mata Kuliah  
**Dasar Pemrograman**



# Perulangan Bersarang

TIM DASAR PEMROGRAMAN  
TEKNIK INFORMATIKA S1  
UNIVERSITAS DIAN NUSWANTORO

## Capaian Pembelajaran

- Setelah mengikuti kuliah ini mahasiswa dapat menjelaskan dan mempraktekkan jenis pengulangan berdasarkan jumlah pengulangan dan berdasarkan kondisi kondisi, dua aksi, dan pencacah dalam pemrograman
- Setelah mengikuti kuliah ini mahasiswa dapat menjelaskan dan mempraktekkan jenis pengulangan bersarang

## Perulangan Bersarang (Nested Loop)

- Perulangan bersarang atau nested loop merupakan Teknik memprogram yang memungkinkan programmer untuk membuat perulangan di dalam perulangan
- Dua bagian: outer loop dan inner loop
- Inner loop bisa lebih dari satu
- Inner loop yang paling dalam biasanya akan di eksekusi terlebih dahulu
- Berikutnya melanjutkan iterasi dari inner loop sebelumnya atau outer loop

# Perulangan Bersarang

<var> traversal < sequence>:

    <var> traversal < sequence>:

        <var> traversal < sequence>:

            ekspresi\_inner2

            ...

        ekspresi\_inner1

Ekspresi\_outer

## Hasil perulangan bersarang – Berdasar Pencacah

```
Outer Loop ke - 1
- Inner Loop ke - 1
- Inner Loop ke - 2
Outer Loop ke - 2
- Inner Loop ke - 1
- Inner Loop ke - 2
Outer Loop ke - 3
- Inner Loop ke - 1
- Inner Loop ke - 2
selesai
```

i	j	output
1		Outer Loop ke - 1
	1	- Inner Loop ke - 1
	2	- Inner Loop ke - 2
2		Outer Loop ke - 2
	1	- Inner Loop ke - 1
	2	- Inner Loop ke - 2
3		Outer Loop ke - 3
	1	- Inner Loop ke - 1
	2	- Inner Loop ke - 2
		selesai

## Pertanyaan

- Apakah bisa pengulangan bersarang ini dikombinasikan dengan berbagai jenis perulangan
  - Jawaban: Bisa
- Apakah pengulangan bersarang bisa memiliki lebih dari 1 inner loop?
  - Jawaban: Bisa
- Apakah perulangan bersarang ini akan mempengaruhi performa?
  - Jawaban: tentu, semakin dalam perulangan yang dilakukan semakin banyak operasi yang dilakukan



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DIAN NUSWANTORO**

# Studi Kasus Perulangan Bersarang

## Cetak Matriks Angka

- Buatlah program untuk menampilkan b x k dengan ukuran 3 x 3 yang berisi angka 1 – 9 seperti contoh berikut:

1	2	3
4	5	6
7	8	9



# Program CetakMatriksAngka

## Program CetakMatriksAngka

{Buatlah program untuk menampilkan matriks  $b \times k$  dengan ukuran  $3 \times 3$  yang berisi angka 1-9}

### KAMUS

$n, b, k$  : int

### ALGORITMA

$n \leftarrow 1$

$b$  traversal  $[0 \dots 3]$

$k$  traversal  $[0 \dots 3]$

output( $n$ )

$n \leftarrow n+1$

output("newline")

}

# Program CetakMatriksAngka

**Program CetakMatriksAngka**  
 {Buatlah program untuk menampilkan matriks  $b \times k$  dengan ukuran  $3 \times 3$  yang berisi angka 1-9}

## KAMUS

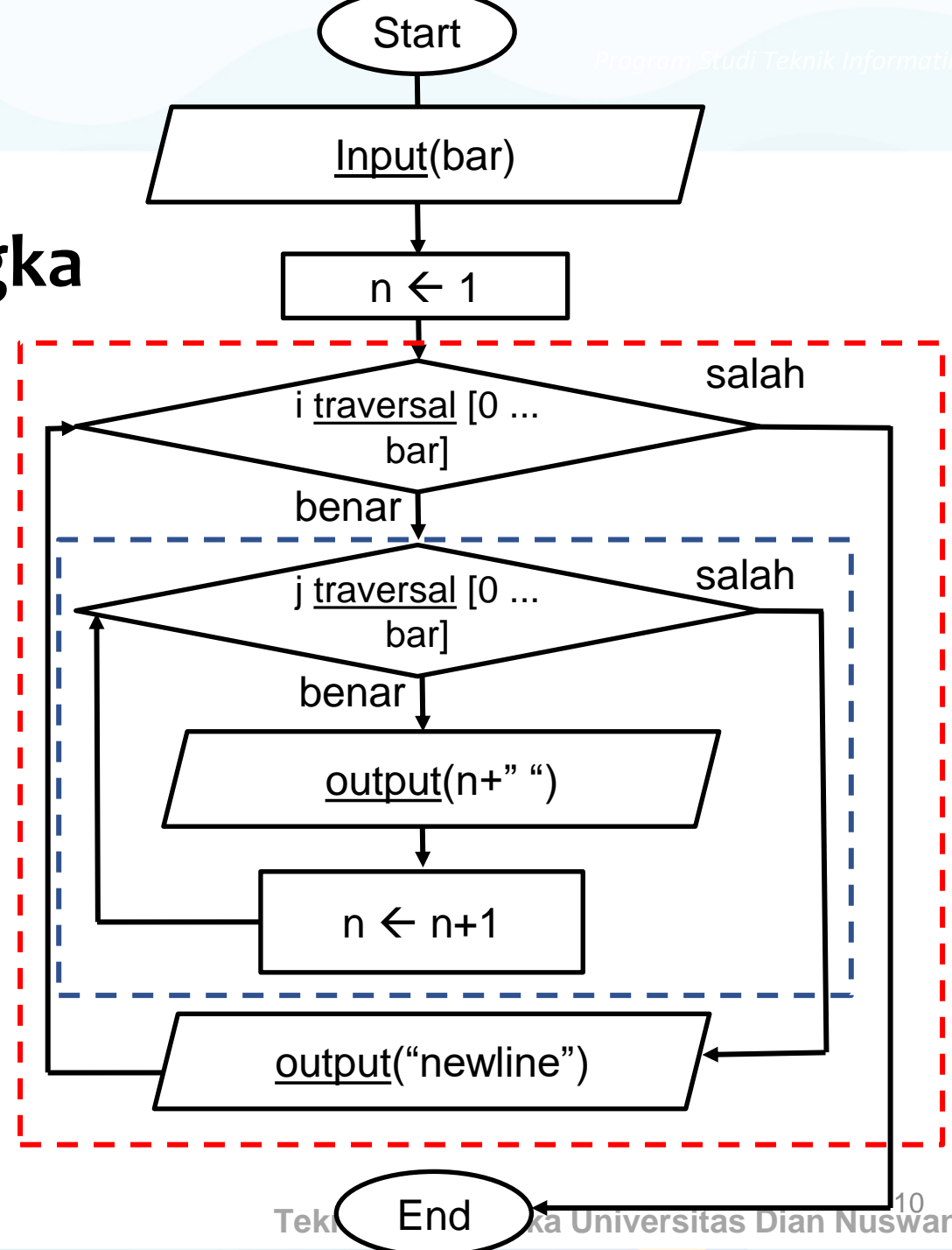
$n, b, k : \text{int}$

## ALGORITMA

$n \leftarrow 1$

```

b traversal [0 ... 3]
  k traversal [0 ... 3]
    output(n+" ")
    n ← n+1
    output("newline")
  }
  
```



# Program CetakMatriksAngkaV2

## Program CetakMatriksAngka

{Buatlah program untuk menampilkan matriks b x k dengan ukuran col x bar, dimana colom adalah batas kolom dan bar adalah batas baris yang di input oleh user}

### KAMUS

n,b,k : int  
col,bar : int

### ALGORITMA

```
n ← 1
input(col)
input(bar)
b traversal [0 ... bar]
  k traversal [0 ... col]
    output(n)
    n ← n+1
  output("newline")
```

# Program CetakMatriksAngkaV3

## Program CetakMatriksAngka

{Buatlah program untuk menampilkan matriks  $b \times k$  dengan ukuran  $col \times bar$ , dimana kolom adalah batas kolom dan bar adalah batas baris yang di input oleh user}

### KAMUS

$n, b, k$  : int  
 $col, bar$  : int

### ALGORITMA

```
n ← 1
b ← 0
input(col)
input(bar)
while b < col do
    b ← b + 1
    k traversal [0 ... bar]
        output(n)
        n ← n+1
    output("newline")
```

# Program CetakMatriksAngkaV4

## Program CetakMatriksAngka

{Buatlah program untuk menampilkan matriks  $b \times k$  dengan ukuran  $col \times bar$ , dimana kolom adalah batas kolom dan bar adalah batas baris yang di input oleh user}

### KAMUS

$n, b, k$  : int  
 $col, bar$  : int

### ALGORITMA

```
n ← 1
b ← 0
input(col)
input(bar)
repeat
    b ← b + 1
    k traversal [0 ... bar]
        output(n)
        n ← n+1
    output("newline")
until b < col
```

## Kasus Segitiga Bintang

- Gambar segitiga dengan n bertipe bilangan bulat sebagai batas segitiga yang di input user sehingga akan menghasilkan output seperti ini:

\*

\* \*

\* \* \*

\* \* \* \*

- Jika input user adalah 4

# Program SegitigaBintang

```
Program ProgramSegitigaBintang  
{buat segitiga sama kaki dengan batas n yang di input user}
```

## KAMUS

```
n,i,j : int
```

## ALGORITMA

```
input(n)  
i traversal [0 ... n]  
  j traversal [0 ... i]  
    output("*")  
  output("newline")
```

# Program ProgramSegitigaBintang

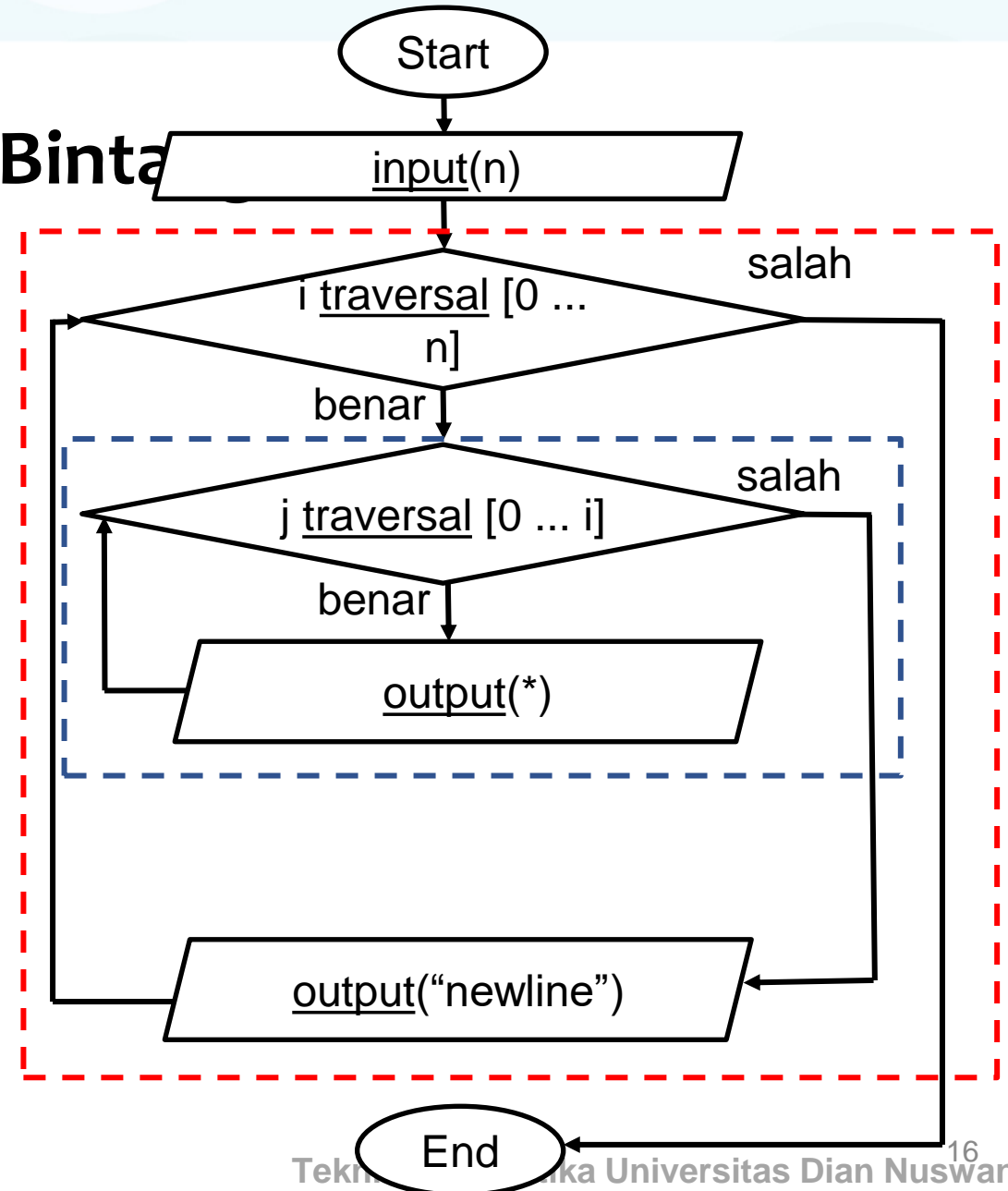
**Program ProgramSegitigaBintang**  
 {buat segitiga sama kaki dengan batas n yang di input user}

## KAMUS

n,i,j : int

## ALGORITMA

input(n)  
 i traversal [0 ... n]  
 j traversal [0 ... i]  
 output("\*")  
 output("newline")





# 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?