



Algoritma dan Pemrograman I

Perulangan Menggunakan FOR

Rosa A. S.

Rosa Ariani Sukamto

Blog: <http://hariiniadalahhadiah.wordpress.com>

Facebook: <https://www.facebook.com/rosa.ariani.sukamto>

Email: rosa.ariani@upi.edu

Website: <https://rosa-as.id>

Perulangan FOR

- ❑ Perulangan adalah mekanisme dalam pembuatan program untuk melakukan proses yang sama berulang-ulang
- ❑ Perulangan yang digunakan untuk jumlah perulangan yang sudah pasti diketahui biasanya menggunakan for
- ❑ For digunakan jika banyaknya perulangan yang dilakukan sudah pasti

For count up (1)

```
for(nama_variabel=nilai_awal;  
    nama_variabel operator_relasi  
    nilai_berhenti;nama_variabel++){  
    // proses  
    .....  
}
```

For count up (2-1)

```
× for (i = 0; i < 10; i++) {  
×     printf("alpro untuk masa depanku  
    yang cemerlang %d\n", i);  
× }
```

For count up (2-2)

```
int n, i;
scanf("%d", &n);
for (i = 1; i <= n; i++) {
    printf("alpro untuk masa depanku
    yang cemerlang %d\n", i);
}
```

For count up (3 - 1)

Menjumlahkan angka dari angka satu
sampai angka masukan *user*

For count up (3 - 2)

```
#include <stdio.h>
```

```
int main(){  
    int berapa_kali;   int hasil_jumlah;   int penghitung;  
    hasil_jumlah = 0;  
    printf("masukkan angka berapa kali akan diulang:\n");  
    scanf("%d", &berapa_kali);  
    printf("\n");  
    for(penghitung=1;penghitung<=berapa_kali; penghitung++){  
        hasil_jumlah = hasil_jumlah + penghitung;  
    }  
    printf("%d\n", hasil_jumlah);  
    return 0;  
}
```


Logika

hasil_jumlah = 0

berapa_kali = 3

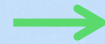
putaran 1

penghitung = 1

hasil_jumlah = 0



hasil_jumlah = 0 + 1



1

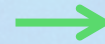
putaran 2

penghitung = 2

hasil_jumlah = 1



hasil_jumlah = 1 + 2



3

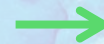
putaran 3

penghitung = 3

hasil_jumlah = 3



hasil_jumlah = 3 + 3



6

For count up (4-1)

Menghitung nilai faktorial dari
masukan *user*

For count up (4-2)

```
#include <stdio.h>
```

```
int main(){
```

```
    int fak_berapa;  int penghitung;  int hasil_faktorial;
```

```
    printf("masukkan angka faktorial yang akan dihitung:\n");
```

```
    scanf("%d", &fak_berapa);
```

```
    printf("\n");
```

```
    hasil_faktorial = 1;
```

```
    for(penghitung=2; penghitung <= fak_berapa; penghitung++){
```

```
        hasil_faktorial =      hasil_faktorial * penghitung;
```

```
    }
```

```
    printf("hasil faktorial: %d\n", hasil_faktorial);
```

```
    return 0;
```

```
}
```

For count up (5 - 1)

Membuat bintang sejajar 1 baris sebanyak masukan *user*

```
#include <stdio.h>

int main(){
    int penghitung;
    int bintang;
    printf("masukkan banyaknya bintang: \n");
    scanf("%d", &bintang);
    for(penghitung=1; penghitung<=bintang; penghitung++){
        printf("*");
    }
    printf("\n");
    return 0;
}
```


For count up (5 - 2)

Membuat bintang sejajar beberapa baris sebanyak masukan *user*

For count up

- × Membuat bintang sejajar beberapa baris sebanyak masukan *user*

$n = 4$

baris ke-1 bintang = 4

baris ke-2 bintang = 4

baris ke-3 bintang = 4

baris ke-4 bintang = 4

For count up (5 - 3)

```
#include <stdio.h>

int main(){
    int baris;
    int kolom;
    int bintang;
    printf("masukkan banyaknya bintang: \n");
    scanf("%d", &bintang);
    for(baris=1; baris<=bintang; baris++){
        for(kolom=1; kolom<=bintang; kolom++){
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```

For count up (5 - 4)

Membuat bintang sejajar beberapa baris sebanyak masukan *user*

*

**

For count up

- × Membuat bintang sejajar beberapa baris sebanyak masukan *user*

*

n = 4

baris ke-1 bintang = 1

**

baris ke-2 bintang = 2

baris ke-3 bintang = 3

baris ke-4 bintang = 4

For count up (5 - 5)

```
#include <stdio.h>

int main(){
    int baris;
    int kolom;
    int bintang;
    printf("masukkan banyaknya bintang: \n");
    scanf("%d", &bintang);
    for(baris=1; baris<=bintang; baris++){
        for(kolom=1; kolom<=baris; kolom++){
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```

For count down (1)

```
for(nama_variabel=nilai_awal; nama_variabel  
    operator relasi nilai_berhenti;  
    nama_variabel--){  
    // proses  
    .....  
}
```

For count down (2)

```
for (i = 10; i > 0; i--) {  
    printf("hitung %d\n", i);  
}  
printf("Teeeeeeetttttt.....\n");
```


For count down (3)

```
int n;  
scanf("%d", &n);  
for (i = n; i > 0; i--){  
    printf("hitung %d\n", i);  
}  
printf("Teeeeeeetttttt.....\n");
```

```

*      n = 4      for (baris=1;baris<=n;baris++) {

**                for (kolom=1;kolom<=baris;kolom++) {
                  printf("*");
                  }
***             printf("\n");
**             }

****           for (baris=n-1;baris>=1;baris--) {

***           for (kolom=1;kolom<=baris;kolom++) {
**             printf("*");
*             }
              printf("\n");
            }

```

n = 4

```

- - - *
- - **
- ***
****
- ***
- - **
- - - *
```

```

for (baris=1;baris<=n;baris++) {
    for (kolom=n-baris;kolom>=1;kolom--) {
        printf(" ");
    }

    for (kolom=1;kolom<=baris;kolom++) {
        printf("*");
    }
    printf("\n");
}

for (baris=1;baris<=n-1;baris++) {
    for (kolom=1;kolom<=baris;kolom++) {
        printf(" ");
    }
    for (kolom=n-baris;kolom>=1;kolom--) {
        printf("*");
    }
    printf("\n");
}
```

n = 4

```

-- -- * -- -- *
-- -- ** -- -- **
-- *** -- ***
****
```

```
for (baris=1; baris<=n; baris++) {
    for (kolom=n-baris; kolom>=1; kolom--) {
        printf(" ");
    }

    for (kolom=1; kolom<=baris; kolom++) {
        printf("*");
    }

    for (kolom=n-baris; kolom>=1; kolom--) {
        printf(" ");
    }

    for (kolom=1; kolom<=baris; kolom++) {
        printf("*");
    }
    printf("\n");
}
```


n=5

--- * --- *****
--- *** --- *****
-- ***** -- *****
- ***** - *****

- ***** - *****
-- ***** -- *****
--- *** --- *****
--- * --- *****

n=5

```

-- -- -- * -- -- -- * * * * * * * * * *
-- -- * * * -- -- -- * * * * * * * * *
-- -- * * * * * -- -- * * * * * * * *
- * * * * * * * * - * * * * * * *
* * * * * * * * * * * * * * * *
-----
- * * * * * * * * - * * * * * * *
-- -- * * * * * -- -- * * * * * * *
-- -- -- * * * -- -- -- * * * * * * *
-- -- -- -- * -- -- -- * * * * * * * * *

```

```

for() {
    for() {
        printf(" ");
    }
    for() {
        printf("*");
    }
    for() {
        printf("*");
    }
    for() {
        printf(" ");
    }
    for() {
        printf("*");
    }
    for() {
        printf("*");
    }
    printf("\n");
}

```

Tantangan latihan

Bisa mencoba soal-soal di *Online Judge* CSPC

Daftar Pustaka



Rosa A. S.

LOGIKA ALGORITMA dan PEMROGRAMAN DASAR

- Pendahuluan Algoritma dan Pemrograman
- Komentar
- Tipe Data
- Masukan (Input) dan Keluaran (Output)
- Operator
- Array (Larik)
- Percabangan / Pemilihan If
- Perulangan (Looping)
- Prosedur
- Fungsi
- Matriks dalam Larik (Array) Dua Dimensi
- Rekursif
- Pengurutan (Sorting)
- Penggabungan Larik
- Pencarian (Searching)
- Arap Beruntun (Sequential File)
- Mesin Abstrak
- Flowchart

Algoritma berarti solusi. Ketika orang berbicara mengenai algoritma di bidang pemrograman, maka yang dimaksud adalah solusi dari suatu masalah yang harus dipecahkan dengan menggunakan komputer. Algoritma harus dibuat secara runtut agar komputer mengerti dan mampu mengeksekusinya. Analisis kasus sangat dibutuhkan dalam membuat sebuah algoritma, misalnya proses apa saja yang sekiranya dibutuhkan untuk menyelesaikan masalah yang harus diselesaikan. Ketajaman dalam menganalisis sebuah kasus dapat dilatih dengan berlatih menyelesaikan kasus-kasus algoritma, mulai dari yang paling sederhana sampai rumit. Kesabaran sangat dibutuhkan dalam mempelajari algoritma.

Esensi dari belajar algoritma adalah membuat solusi untuk menyelesaikan permasalahan, jadi hasilnya adalah dapat menyelesaikan permasalahan. Maka sangat tidak sesuai dengan esensi jika algoritma dipelajari dengan hanya tahu pola-pola atau teknik-teknik algoritma tapi tidak mampu secara logika menyelesaikan permasalahan yang berbeda-beda. Maka dari itu sangat dibutuhkan latihan soal dalam mempelajari algoritma dan pemrograman agar logika terlatih untuk membuat solusi dari permasalahan.

Penulis berharap buku ini dapat membantu para pembaca pada umumnya dan mahasiswa, siswa SMP, SMK, SMA khususnya dalam memahami pemrograman secara lebih baik. Dalam buku ini banyak diberikan penggambaran/ilustrasi secara visual agar pembaca lebih mudah dalam memahami isi buku. Selain itu, dalam buku ini juga diberikan bagaimana mengimplementasikan suatu algoritma dalam bahasa algoritmik, bahasa Pascal, bahasa C, C++, dan Java agar pembaca dapat lebih mudah mengimplementasikannya secara langsung.

Rosa A. S.

LOGIKA ALGORITMA dan PEMROGRAMAN DASAR

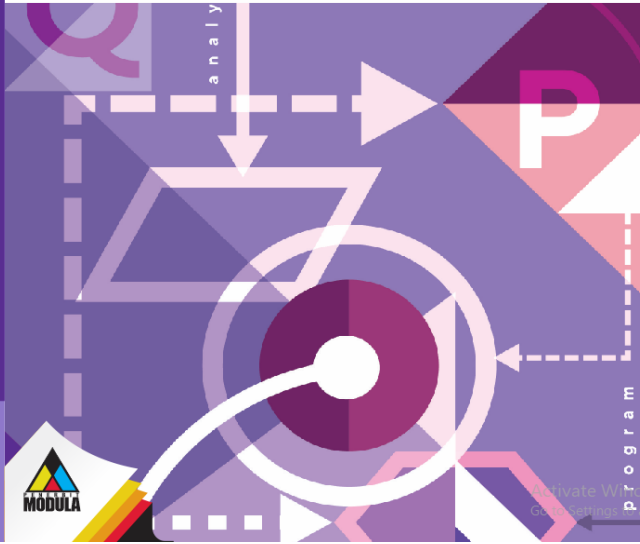


Pemasaran: BI-OBSES
Pasar buku Palasari 82 Bandung 40264
Tel: (022) 7317812 Fax: (022) 7317896
www.biobses.com

LOGIKA ALGORITMA dan PEMROGRAMAN DASAR

- Pendahuluan Algoritma dan Pemrograman
- Komentar
- Tipe Data
- Masukan (Input) dan Keluaran (Output)
- Operator
- Array (Larik)
- Percabangan / Pemilihan If
- Perulangan (Looping)
- Prosedur
- Fungsi
- Matriks dalam Larik (Array) Dua Dimensi
- Rekursif
- Pengurutan (Sorting)
- Penggabungan Larik
- Pencarian (Searching)
- Arap Beruntun (Sequential File)
- Mesin Abstrak
- Flowchart

Rosa A. S.



Activate Windows
Go to Settings to activate Windows.