ALGORITMA DAN PENIROGRAMANI PERULANGAN MENGGUNAKAN FOR

ROSA ARIANI SUKAMTO

Blog: http://hariiniadalahhadiah.wordpress.com

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

Email: rosa_if_itb_01@yahoo.com

Website: https://rosa-as.id

PERULANGAN FOR

- Perulangan adalah mekanisme dalam pembuatan program untuk melakukan proses yang sama berulangulang
- 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 COUNT UP (2-1)

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

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);
}</pre>
```

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;</pre>
  penghitung++) {
       hasil jumlah = hasil jumlah + penghitung;
  printf("%d\n", hasil jumlah);
  return 0;
```

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++) {</pre>
      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;</pre>
  penghitung++) {
    printf("*");
  return 0;
```

FOR COUNT UP (5 - 2)

Membuat bintang sejajar beberapa baris sebanyak masukan *user*

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++) {</pre>
    for(kolom=1; kolom<=bintang; kolom++) {</pre>
      printf("*");
    printf("\n");
  return 0;
```

FOR COUNT UP (5 - 4)

Membuat bintang sejajar beberapa baris sebanyak masukan *user*

*

**

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++) {</pre>
    for(kolom=1; kolom<=baris; kolom++) {</pre>
      printf("*");
    printf("\n");
  return 0;
```

FOR COUNT DOWN (1)

FOR COUNT DOWN (2)

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

FOR COUNT DOWN (3)

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

TANTANGAN LATIHAN

Bisa mencoba soal-soal di *Online Judge* CSPC

DAFTAR PUSTAKA



PEMROGRAMAN

□ Komentar
□ Tipe Data

Masukan (Input) dan Keluaran (Output)

□ Operator □ Array (Larik)

p Fungsi

Matriks dalam Larik (Array) Dua Dimensi

Rekursif
 Pengurutan (Sorting)

Penggabungan Larik
 Pencarian (Searching)
 Arsip Beruntun (Sequential File)

Mesin Abstrak

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 runut 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 polapola 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.





Pemasaran: BI-OBSES Pasar buku Palasari 82 Bandung 40264 Tel.(022) 7317812 Fax.(022) 7317896 Rosa A.

LOGIKA ALGORITMA **PEMROGRAMAN**

Rosa A. S.

□ 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)
- Arsip Beruntun (Sequential File)
- Mesin Abstrak





