

ALGORITMA DAN PEMROGRAMAN I

# LARIK (ARRAY) TIPE DASAR DAN OPERASINYA DENGAN 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](mailto:rosa.ariani@upi.edu)
- > Website: <https://rosa-as.id>



## CHAR

- Untuk tipe data char, sama halnya dengan pendeklarasian integer maupun float.
- Cara mendeklarasikan adalah

```
tipe_data nama_variabel;
```

```
tipe_data nama_variabel = nilai;
```

- Tipe char dideklarasikan dengan menggunakan tanda petik satu (')

```
char karakter = 'A';
```

## CHAR (2)

```
#include <stdio.h>

int main(){

    char c;

    scanf("%c", &c);
    printf("c: %c\n", c);

    return 0;
}
```



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

## CHAR (3)

---

```
int main(){
```

```
    char c;
```

```
    scanf("%c", &c);
```

```
    //jika scanf tidak berhenti untuk menunggu masukan
```

```
    while(getchar() != '\n');
```

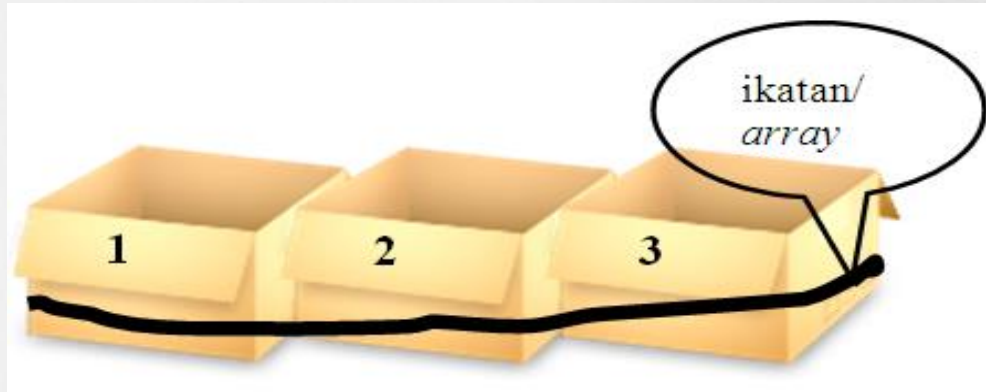
```
    printf("c: %c\n", c);
```

```
    return 0;
```

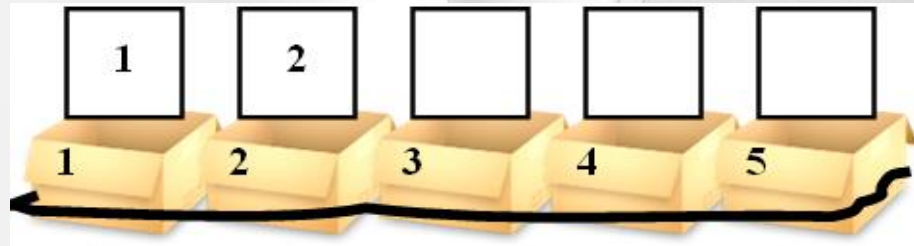
```
}
```



## LARIK (ARRAY)



## LARIK (ARRAY) (2)



# MENGISI ARRAY

```
int main(){
    int tabInt[5];
    int penghitung;
    for(penghitung=0; penghitung < 5; penghitung++){
        tabInt[penghitung] = penghitung;
    }
    for(penghitung=0; penghitung < 5; penghitung++){
        printf("%d ", tabInt[penghitung]);
    }
    return 0;
}
```



## MENGISI ARRAY (2)

```
int main(){
    int n;
    scanf("%d", &n);
    int tabInt[n];
    int i;
    for(i=0; i < n; i++){
        scanf("%d", &tabInt[i]);
    }

    for(i=0; i < n; i++){
        printf("%d ", tabInt[i]);
    }
    return 0;
}
```

```
int main(){
    int n;
    scanf("%d", &n);
    int tabInt[n];
    int i;
    for(i=0; i < n; i++){
        scanf("%d", &tabInt[i]);
    }
    for(i=0; i < n; i++){
        if(tabInt[i] % 2 == 1){
            printf("%d ", tabInt[i]);
        }
    }
    return 0;
}
```

## MENAMPILKAN ISI ARRAY HANYA BILANGAN GANJIL

```
int main(){
    int n;
    scanf("%d", &n);
    int tabInt[n];
    int i;
    for(i=0; i < n; i++){
        scanf("%d", &tabInt[i]);
    }
    int jumlah = 0;
    for(i=0; i < n; i++){
        if(tabInt[i] % 2 == 1){
            jumlah++;
        }
    }
    printf("banyaknya elemen yang termasuk ganjil: %d\n", jumlah);
    return 0;
}
```

## MENGHITUNG BANYAKNYA ELEMEN ISI ARRAY YANG MERUPAKAN BILANGAN GANJIL



# MENAMPILKAN SEPARUH ISI ARRAY

```
int main(){
    int n;
    scanf("%d", &n);
    int tabInt[n];
    int i;
    for(i=0; i < n; i++){
        scanf("%d", &tabInt[i]);
    }

    for(i=0; i < (n/2); i++){
        printf("%d ", tabInt[i]);
    }
    return 0;
}
```

## MENCARI NILAI MAKSIMAL

```
int main(){
    int n;
    scanf("%d", &n);
    int tabInt[n];
    int i;
    for(i=0; i < n; i++){
        scanf("%d", &tabInt[i]);
    }
    int maksimal = tabInt[0];
    for(i=1; i < n; i++){
        if(maksimal < tabInt[i]){
            maksimal = tabInt[i];
        }
    }
    printf("nilai maksimal: %d\n", maksimal);
    return 0;
}
```

## TANTANGAN LATIHAN

- Carilah nilai maksimal dari *array of integer* dari yang bernilai ganjil!
- Diberikan dua buah *array of integer*, jumlahkanlah hasil nilai maksimal *array* pertama dengan hasil nilai minimal dari *array* kedua!
- Hitunglah nilai rata-rata dari semua elemen di dalam *array*!
- Carilah nilai minimal dari elemen *array of integer* yang memiliki indeks/posisi genap!



# 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)
- Perulangan / Pemilihan If
- Perulangan (Looping)
- Prosedur
- Fungsi
- Matriks dalam Larik (Array) Dua Dimensi
- Huruf
- Penggabungan (Sorting)
- Pencarian (Searching)
- Arasp Berurutan (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.



Pemastaran, BL-08BES  
Pusat Buku Pajang 82 Bandung 40264  
Tel (022) 7317812 Fax (022) 7317896  
www.bl08bes.com

Rosa A. S.

## LOGIKA ALGORITMA dan PEMROGRAMAN DASAR



Rosa A. S.

## LOGIKA ALGORITMA dan PEMROGRAMAN DASAR

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