

The background is white with several decorative elements: a large teal ring in the top-left, a smaller teal circle next to it, a large lime green circle in the top-right, a smaller green circle with a dashed outline next to it, a large orange circle in the bottom-right, a smaller pink circle next to it, a large green circle in the bottom-left, a smaller yellow circle next to it, and a large yellow ring in the bottom-right. A dashed blue line curves around the central text area.

Algoritma dan Pemrograman I

While & Array of Tipe Terstruktur

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>

Pembahasan UTS Soal Nomor 1

***		***
***		***
***		***
<hr/>		
***	***	***
***	***	***
***	***	***
<hr/>		
***	*****	***
***	*****	***
***	*****	***
<hr/>		
***	***	***
***	***	***
***	***	***
<hr/>		
***		***
***		***
***		***

Pembahasan Soal UTS Nomor 2

Memasukkan elemen ke dalam dua buah array

for

2	5	13	12	7	8
---	---	----	----	---	---

for

17	18	19	56	77	18
----	----	----	----	----	----

Pembahasan Soal UTS Nomor 2

Menukar elemen

for

indeks

2	5	13	12	7	8
---	---	----	----	---	---

temp

17	18	19	56	77	18
----	----	----	----	----	----

Pembahasan Soal UTS Nomor 2

Menukar elemen

for

2	5	13	12	7	8
---	---	----	----	---	---

indeks

2

13

temp

temp = arr1[indeks] ;

17	18	19	56	77	18
----	----	----	----	----	----

Pembahasan Soal UTS Nomor 2

Menukar elemen

for

indeks

2	5	19	12	7	8
---	---	----	----	---	---

2

13

temp



```
temp = arr1[indeks] ;  
arr1[indeks] = arr2[indeks];
```

17	18	19	56	77	18
----	----	----	----	----	----

Pembahasan Soal UTS Nomor 2

Menukar elemen

for

indeks

2	5	19	12	7	8
---	---	----	----	---	---

2

13

temp

```
temp = arr1[indeks] ;  
arr1[indeks] = arr2[indeks];  
arr2[indeks] = temp;
```

17	18	13	56	77	18
----	----	----	----	----	----

Pembahasan Soal UTS Nomor 2

Mencetak ke layar

$n/2$

for



2	5	19	12	7	8
---	---	----	----	---	---



17	18	13	56	77	18
----	----	----	----	----	----

while

Perulangan memiliki kemungkinan berhenti di tengah ketika tujuan perulangan telah terpenuhi

Biasanya memiliki minimal 2 syarat kondisi perulangan:

- ⦿ Batas maksimal perulangan dilakukan
- ⦿ Batas jika perulangan berhenti di tengah

```
// inisialisasi
.....
while(kondisi_pengulangan) {

    // proses
    .....
    // iterasi
}
```

Perulangan - while (2)

```
int i;  
  
// inisialisasi + iterasi  
for(i=1; i<=9; i++){  
    // proses  
    .....  
}
```

```
int i;  
  
// inisialisasi  
i = 1;  
while(i <= 9){  
    // proses  
    .....  
    // iterasi  
    i = i + 1;  
}
```

Perulangan – while (3)

“

- © algoritma pencarian pada sebuah *array* bilangan integer apakah di dalam *array* bilangan integer ada angka yang merupakan bilangan ganjil

Perulangan - while (4)

```
#include <stdio.h>

int main(){
    int penghitung;
    int tabInt[5];
    int ketemu;

    for(penghitung=0; penghitung<5;
    penghitung++){
        printf("masukkan angka:\n");
        scanf("%d", &tabInt[penghitung]);
        printf("\n");
    }

    ketemu = 0;
    penghitung = 0;
```

```
while((ketemu == 0) && (penghitung <
5)){

    if((tabInt[penghitung] % 2) ==
1){

        ketemu = 1;

    }else{

        penghitung = penghitung + 1;

    }

}

if(ketemu == 0){

    printf("tidak ada angka
ganjil\n");

}else{

    printf("ada angka ganjil\n");

}

return 0;

}
```

A decorative graphic consisting of various colored circles and rings. In the top left, there are overlapping pink, orange, and blue circles, with a dashed pink circle around the pink one. Below these is a large blue ring. In the bottom left, there is a large green ring and a smaller green circle. In the top right, there are several blue and yellow circles, some solid and some dashed. In the bottom right, there is a small yellow circle and a small green circle.

Perulangan – while (5)

- ◎ Buat algoritma yang mengisi sebuah *array integer* dan menampilkan isi *array*, tapi berhenti menampilkan jika ditemukan angka 999!

Perulangan - while (6)

```
#include <stdio.h>

int main(){
    int tabInt[5];
    int penghitung;
    for(penghitung=0; penghitung<5; penghitung++){
        printf("masukkan angka:\n");
        scanf("%d", &tabInt[penghitung]);
        printf("\n");
    }
    penghitung = 0;
    while((tabInt[penghitung] != 999) && (penghitung < 5)){
        printf("%d\n", tabInt[penghitung]);
        penghitung = penghitung + 1;
    }
    return 0;
}
```



Perulangan – while (7)

Tampilkan maksimal 3
angka genap yang ada
pada sebuah array of
integer?

Perulangan – while (8)

```
#include <stdio.h>
int main(){
    int n; scanf("%d", &n);
    int tabInt[n];
    int penghitung;
    for(penghitung=0; penghitung<n; penghitung++){
        printf("masukkan angka:\n");
        scanf("%d", &tabInt[penghitung]);
        printf("\n");
    }
    penghitung = 0;
    int maks = 0;
    while((maks <3) && (penghitung < n)){
        if(tabInt[penghitung] % 2 == 0){
            printf("%d\n", tabInt[penghitung]);
            maks++;
        }
        penghitung = penghitung + 1;
    }
    return 0;
}
```

The background features a light blue dashed circle. Various colored circles and arcs are scattered around it: a large yellow circle at the top left, a large teal arc at the top center, a large orange circle at the bottom left, a large yellow circle at the bottom right, and several smaller circles in green, blue, orange, and pink.

Syarat ABSOLUT BISA ALPRO

Latihan Menyelesaikan Soal

&&

Bertanya jika tidak bisa di dalam latihan

Tipe Terstruktur

bungkusan
titik

```
typedef struct{  
    int x;  
    int y;  
}titik;
```

```
int main(){  
    titik titik1;  
    titik1.x = 9;  
    .....  
    return 0;  
}
```

Array of Tipe Terstruktur (1)



```
typedef struct{  
    int x;  
    int y;  
}titik;
```

```
int main(){  
    titik ikatan_titik[2];  
    ikatan_titik[0].x = 9;  
    .....  
    return 0;  
}
```

A decorative graphic consisting of several overlapping circles and rings in various colors including yellow, pink, orange, green, blue, and red. Some circles have dashed outlines, and some are solid. They are scattered across the slide, with a large yellow circle on the left and a cluster of smaller circles on the right.

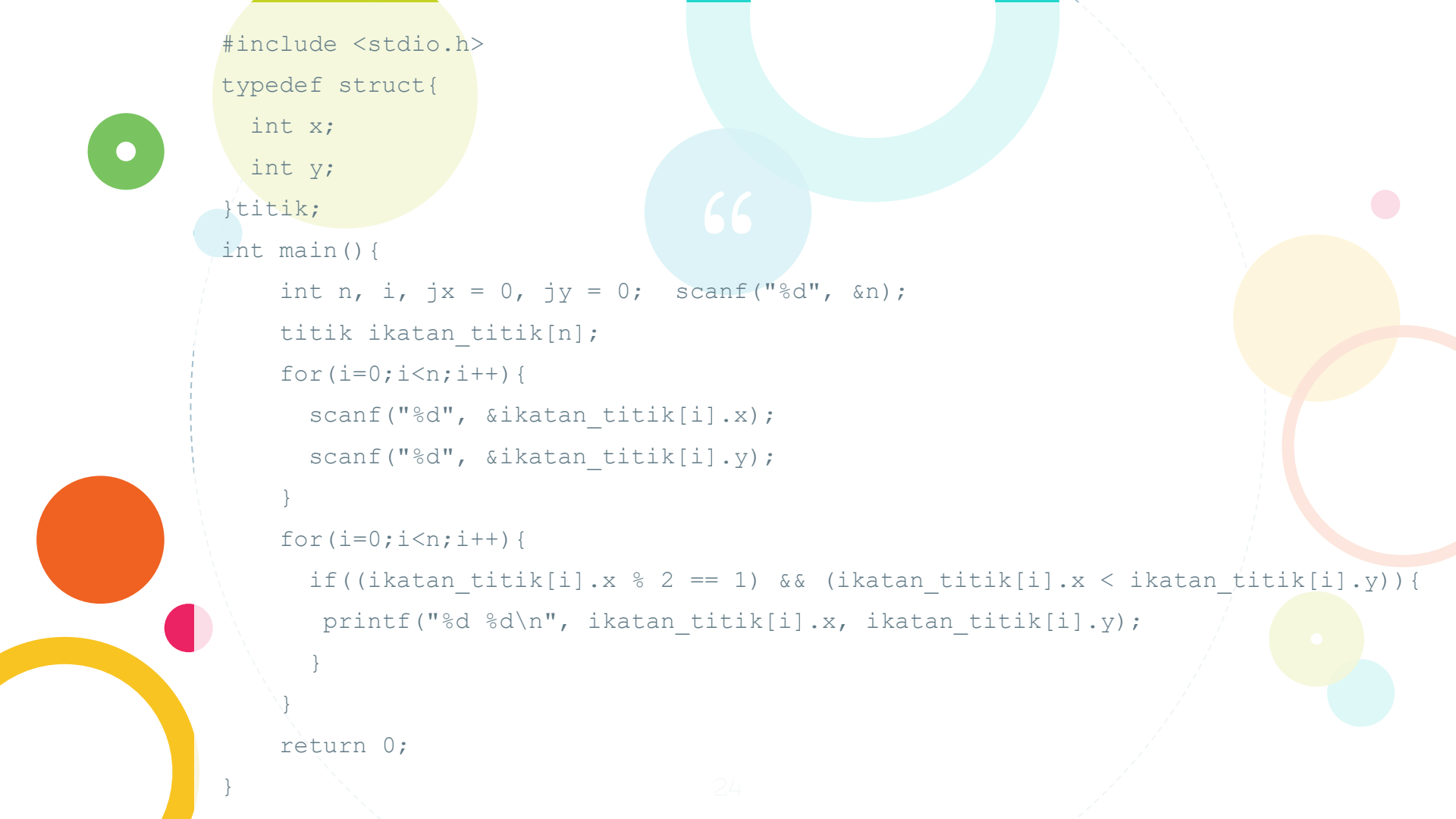
● Array of Tipe Terstruktur (2)

- ◎ Jumlahkan isi X dan isi Y yang ada di dalam Array of Titik

```
#include <stdio.h>
typedef struct{
    int x;
    int y;
}titik;
int main(){
    int n, i, jx = 0, jy = 0;
    scanf("%d", &n);
    titik ikatan_titik[n];
    for(i=0;i<n;i++){
        scanf("%d", &ikatan_titik[i].x);
        scanf("%d", &ikatan_titik[i].y);
    }
    for(i=0;i<n;i++){
        jx = jx + ikatan_titik[i].x;
        jy = jy + ikatan_titik[i].y;
    }
    printf("%d %d\n", jx, jy);
    return 0;
}
```

Array of Tipe Terstruktur (4)

Tampilkan semua titik yang
x-nya ganjil dan y-nya lebih
besar dari x



“

```
#include <stdio.h>
typedef struct{
    int x;
    int y;
}titik;

int main(){
    int n, i, jx = 0, jy = 0;  scanf("%d", &n);
    titik ikatan_titik[n];
    for(i=0;i<n;i++){
        scanf("%d", &ikatan_titik[i].x);
        scanf("%d", &ikatan_titik[i].y);
    }
    for(i=0;i<n;i++){
        if((ikatan_titik[i].x % 2 == 1) && (ikatan_titik[i].x < ikatan_titik[i].y)){
            printf("%d %d\n", ikatan_titik[i].x, ikatan_titik[i].y);
        }
    }
    return 0;
}
```


Mari Mencoba untuk Masa Depan CERIA 😊

- Diberikan sebuah array bungkusan bertipe segitiga (alas, tinggi, sisiMiring), tampilkan maksimal 3 segitiga yang merupakan segitiga siku-siku (gunakan rumus phitagoras untuk memeriksa).
- Diberikan sebuah array bertipe float, tampilkan semua nilai float dimana nilai di depan koma dan nilai di belakang koma adalah genap.
- Diberikan sebuah array bungkusan bertipe pecahan (pembilang, penyebut), tampilkan maksimal 3 pecahan yang merupakan bilangan bulat (pembilang dapat dibagi habis oleh penyebut)
- Diberikan sebuah array bungkusan bertipe pecahan (pembilang dan penyebut), jumlahkan semua pecahan yang merupakan bilangan bulat (hasil bagi pembilang oleh penyebut)

Daftar Pustaka



LOGIKA ALGORITMA dan PEMROGRAMAN DASAR

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 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.

