

Array 1

1. Kompetensi

1. Mahasiswa mampu memahami dan menjelaskan fungsi array satu dimensi.
2. Mahasiswa mampu membuat program dengan menggunakan konsep array satu dimensi.

2. Alat dan Bahan

1. PC/Laptop
2. Text editor (Sublime Text)
3. JDK

3. Uraian Teori

Array adalah sekumpulan tempat penyimpanan data yang bertipe sama dan memiliki index. Array dapat diibaratkan sebagai sekumpulan variabel yang bertipe sama dan bernama sama. Array biasanya digunakan untuk menyimpan deret angka. Kemudian untuk membedakan nilai/isi dari variabel tersebut, digunakan index.

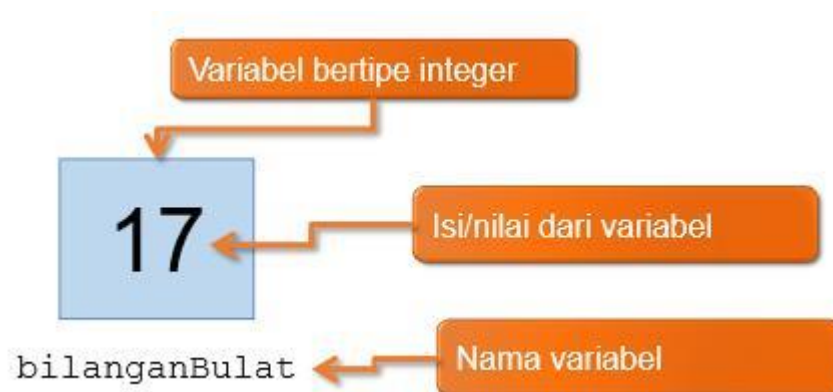
Ada dua tipe array:

- Array satu dimensi
- Array multi dimensi

Pada jobsheet praktikum ini hanya akan dibahas array satu dimensi saja. Array multi dimensi akan dibahas di jobsheet selanjutnya.

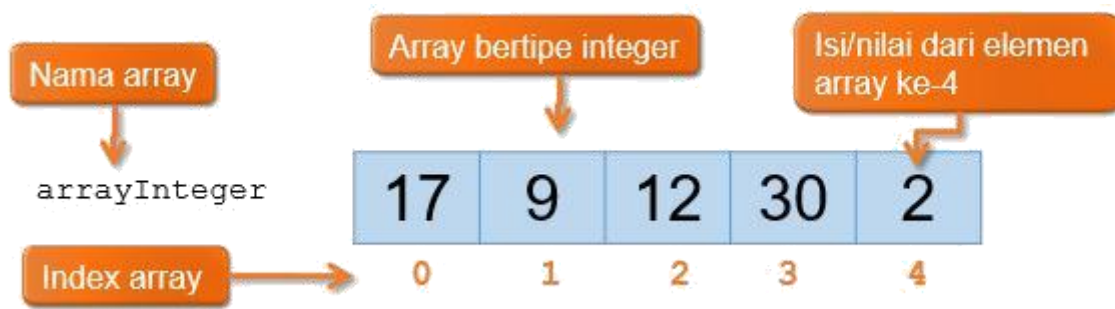
3.1 Array Satu Dimensi

Amati ilustrasi tentang **variabel** berikut ini.



Gambar 1. Variabel

Ilustrasi diatas adalah variabel bertipe integer yang bernama **bilanganBulat** dan berisi nilai 17. Satu buah variabel hanya dapat menyimpan **satu** buah nilai. Sekarang amati ilustrasi tentang **array** berikut ini:



Gambar 2. Array satu dimensi

Ilustrasi diatas adalah **array** bertipe **integer** bernama **arrayInteger** dan memiliki kapasitas 5 buah bilangan integer. Sebuah array dapat menyimpan lebih dari satu nilai (tergantung dari kapasitasnya). Meskipun begitu, nilai-nilai yang disimpan di sebuah array harus bertipe sama. Tiap elemen pada array dinomori dengan **index array**. Index array selalu dimulai dari 0 (nol).

3.2 Cara Mendeklarasikan Array

Secara umum, cara mendeklarasikan array adalah sebagai berikut:

```
tipe[] namaArray = new tipe[kapasitas];
```

- **Tipe** adalah tipe data dari array yang akan dibuat.
- **namaArray** adalah nama dari array yang akan dibuat.
- **Kapasitas** adalah banyaknya nilai yang dapat disimpan didalam array yang akan dibuat.

Untuk mengakses (mengisi/membaca) sebuah elemen dari array, kita hanya perlu menuliskan nama array tersebut, kemudian diikuti dengan index yang dituju didalam tanda kurung kotak [].

Contoh program berikut ini membuat array bernama **bil**, bertipe **integer**, jumlah elemen 4, kemudian mengisinya dengan beberapa nilai, kemudian menampilkan isi elemennya ke layar.

```
class MyArray
{
    public static void main(String[] args)
    {
        int[] bil = new int[4];
        bil[0] = 5;
        bil[1] = 12;
        bil[2] = 7;
        bil[3] = 20;
        System.out.println(bil[1]);
        System.out.println(bil[0]);
        System.out.println(bil[3]);
    }
}
```

Deklarasi array

Mengisi array

Menampilkan isi array elemen ke 1, 0 dan 3

Hasil:

```
E:\Java>java MyArray
12
5
20
```

Ada beberapa alternatif cara untuk mendeklarasikan array.

1. Deklarasi array tanpa mengalokasikan jumlah elemennya:

```
int[] myArray;
```

2. Deklarasi array dengan mengalokasikan 10 elemen:

```
int[] myArray = new int[10];
```

3. Mengubah jumlah elemen array dengan 50:

```
myArray = new int[50];
```

4. Deklarasi array dan mengisinya secara langsung:

```
int[] myArray = {10, 20, 30, 40};
String[] myArray2 = {"Malang", "Surabaya"};
```

5. Mengetahui jumlah elemen array:

```
myArray.length;
```

3.3 Menggunakan Perulangan Pada Array Satu Dimensi

Untuk mengakses (mengisi/membaca) nilai dari sebuah array, kita dapat menggunakan perulangan.

```
class MyArray
{
    public static void main(String[] args)
    {
        int[] bil = new int[4];

        bil[0] = 5;
        bil[1] = 12;
        bil[2] = 7;
        bil[3] = 20;

        for(int i = 0; i < 4; i++)
        {
            System.out.println(bil[i]);
        }
    }
}
```

Hasil:

```
E:\Java>java MyArray
5
12
7
20
```

Kita juga bisa menggunakan perulangan untuk menerima input dan menyimpannya kedalam array. Contoh, program yang meminta input sebanyak 5 bilangan kemudian menampilkan kembali 5 bilangan tersebut.

```
import java.util.Scanner;

class MyArray
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int[] bil = new int[5];

        for(int i = 0; i < 5; i++)
        {
            System.out.print("Masukkan bilangan ke-" + i + ": ");
            bil[i] = sc.nextInt();
        }

        for(int i = 0; i < 5; i++)
        {
            System.out.println("Isi bilangan ke-" + i + " adalah: " + bil[i]);
        }
    }
}
```

Hasil:

```
E:\Java>java MyArray
Masukkan bilangan ke-0: 5
Masukkan bilangan ke-1: 10
Masukkan bilangan ke-2: 25
Masukkan bilangan ke-3: 100
Masukkan bilangan ke-4: 15
Isi bilangan ke-0 adalah: 5
Isi bilangan ke-1 adalah: 10
Isi bilangan ke-2 adalah: 25
Isi bilangan ke-3 adalah: 100
Isi bilangan ke-4 adalah: 15
```

3.4 Input Jumlah Elemen Array

Kita juga dapat membuat program yang meminta input berapa banyak jumlah elemen array. Langkahnya secara umum adalah:

1. Deklarasikan array tanpa mengalokasikan jumlah elemennya.
2. Buat input yang menerima jumlah elemen array kemudian simpan di sebuah variabel.
3. Set jumlah elemen array menggunakan variabel yang diinputkan tadi.

Coba amati program berikut ini:

```
import java.util.Scanner;

class MyArray2
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int N;
        int[] arr;

        System.out.print("Masukkan jumlah elemen array: ");
        N = sc.nextInt();

        arr = new int[N];

        for(int i = 0; i < N; i++)
        {
            System.out.print("Masukkan array ke-" + i + ": ");
            arr[i] = sc.nextInt();
        }

        for(int i = 0; i < N; i++)
        {
            System.out.println("Array ke-" + i + " adalah: " + arr[i]);
        }
    }
}
```

Deklarasi array tanpa alokasi jumlah elemen

Input jumlah elemen array, simpan ke variabel N

Set jumlah elemen array menggunakan variabel N

Hasilnya:

```
Masukkan jumlah elemen array: 3
Masukkan array ke-0: 40
Masukkan array ke-1: 20
Masukkan array ke-2: 70
Array ke-0 adalah: 40
Array ke-1 adalah: 20
Array ke-2 adalah: 70
```

4. Langkah Praktikum

Ikuti langkah-langkah praktikum berikut ini.

Praktikum 1

1. Buat Class baru dengan nama MyArray.java
2. Buat array bertipe integer dengan nama bil dengan kapasitas 4 elemen.

```
int[] bil = new int[4];
```

3. Isi masing-masing elemen array bil tadi dengan angka 5, 12, 7, 20.

```
bil[0] = 5;
bil[1] = 12;
bil[2] = 7;
bil[3] = 20;
```

4. Tampilkan ke layar semua isi elemennya:



```
System.out.println(bil[0]);
System.out.println(bil[1]);
System.out.println(bil[2]);
System.out.println(bil[3]);
```

5. Cocokkan dan amati hasilnya dengan gambar berikut ini:

```
run-single:
5
12
7
20
BUILD SUCCESSFUL (total time: 1 second)
```

Pertanyaan

1. Dari percobaan 1 berapakah indeks array terbesar dan terkecil?
2. Jika Isi masing-masing elemen array bil diubah dengan angka 5.0, 12867, 7.5, 2000000. Apa yang terjadi? Mengapa bisa demikian?
3. Ubah statement pada langkah No 4 menjadi seperti berikut

```
for(int i = 0; i < 4; i++)
{
    System.out.println(bil[i]);
}
```

Apa keluaran dari program? Mengapa bisa demikian?

Praktikum 2

1. Buat file baru beri nama "ArrayInputLoop.java".
2. Import dan deklarasikan Scanner untuk keperluan input.

```
import java.util.Scanner;

Scanner sc = new Scanner(System.in);
```

3. Buat array bertipe integer dengan nama nilaiUAS, dengan kapasitas 6 elemen.

```
int[] nilaiUAS = new int[6];
```

4. Menggunakan perulangan, buat input untuk mengisi elemen dari array nilaiUAS.

```
for(int i = 0; i < 6; i++)
{
    System.out.print("Masukkan nilai UAS ke-" + i + ": ");
    nilaiUAS[i] = sc.nextInt();
}
```

5. Menggunakan perulangan, tampilkan semua isi elemen dari array nilaiUAS.

```
for(int i = 0; i < 6; i++)
{
    System.out.println("Nilai UAS ke-" + i + " adalah: " + nilaiUAS[i]);
}
```

6. Cocokkan dan amati hasilnya dengan gambar berikut ini:

```
Masukkan nilai UAS ke-0: 80
Masukkan nilai UAS ke-1: 82
Masukkan nilai UAS ke-2: 75
Masukkan nilai UAS ke-3: 85
Masukkan nilai UAS ke-4: 65
Masukkan nilai UAS ke-5: 70
Nilai UAS ke-0 adalah: 80
Nilai UAS ke-1 adalah: 82
Nilai UAS ke-2 adalah: 75
Nilai UAS ke-3 adalah: 85
Nilai UAS ke-4 adalah: 65
Nilai UAS ke-5 adalah: 70
BUILD SUCCESSFUL (total time: 25 seconds)
```

Pertanyaan

1. Ubah statement pada langkah No 4 menjadi seperti berikut ini :

```
for (int i=0; i<nilaiUAS.length;i++)
{
    System.out.println("masukkan nilai UAS MHS ke-"+i+":");
    nilaiUAS[i] = sc.nextInt();
}
```

Jalankan program, Apakah terjadi perubahan? Mengapa demikian?

2. Apa kegunaan dari nilaiUAS.length ?
3. Ubah statement pada langkah No 5 menjadi seperti berikut ini, sehingga program hanya menampilkan nilai Mahasiswa yang lulus Saja :

```
for(int i = 0; i < nilaiUAS.length; i++)
{
    if(nilaiUAS[i] > 70)
    {
        System.out.println("Mahasiswa ke-" + i + " lulus!");
    }
}
```

Jalankan program dan Jelaskan alur program!



4. Modifikasi program agar menampilkan semua mahasiswa, dan ditandai mana yang lulus dan tidak lulus.

```

-
masukkan nilai UAS MHS ke-0:
80
masukkan nilai UAS MHS ke-1:
60
masukkan nilai UAS MHS ke-2:
70
masukkan nilai UAS MHS ke-3:
85
masukkan nilai UAS MHS ke-4:
75
masukkan nilai UAS MHS ke-5:
73
Mahasiswa ke-0 lulus!
Mahasiswa ke-1 tidak lulus!
Mahasiswa ke-2 tidak lulus!
Mahasiswa ke-3 lulus!
Mahasiswa ke-4 lulus!
Mahasiswa ke-5 lulus!
BUILD SUCCESSFUL (total time: 15 seconds)

```

Praktikum 4

Pada praktikum ini, akan dilakukan percobaan untuk menjumlahkan Array. Program akan menerima input sebanyak 10 nilai mahasiswa. Kemudian program akan menampilkan nilai rata-rata dari 10 Mahasiswa.

1. Buat class baru beri nama "rataNilai".
2. Import dan deklarasikan Scanner untuk keperluan input.

```

import java.util.Scanner;

Scanner sc = new Scanner(System.in);

```

3. Buat array nilaiMHS bertipe integer dengan kapasitas 10. Kemudian deklarasikan variable total dan rata seperti gambar berikut ini

```

int[] nilaiMHS = new int[10];
int total = 0;
double rata;

```

4. Menggunakan perulangan, buat input untuk mengisi array nilaiMHS

```

for (int i=0; i<nilaiMHS.length;i++)
{
    System.out.print("masukkan nilai Mahasiswa ke-"+(i+1)+":");
    nilaiMHS[i] = sc.nextInt();
}

```

5. Menggunakan perulangan untuk menghitung jumlah keseluruhan nilai.


```
for(int i = 0; i < nilaiMHS.length; i++)
{
    total+=nilaiMHS[i];
}
```

6. Kemudian hitung nilai rata-rata dengan cara nilai total dibagi jumlah elemen dari array nilaiMHS

```
rata= total/nilaiMHS.length;
System.out.println(rata);
```

7. Amati hasilnya

```
masukkan nilai Mahasiswa ke-1:85
masukkan nilai Mahasiswa ke-2:90
masukkan nilai Mahasiswa ke-3:75
masukkan nilai Mahasiswa ke-4:70
masukkan nilai Mahasiswa ke-5:80
masukkan nilai Mahasiswa ke-6:60
masukkan nilai Mahasiswa ke-7:55
masukkan nilai Mahasiswa ke-8:75
masukkan nilai Mahasiswa ke-9:85
masukkan nilai Mahasiswa ke-10:50
nilai rata-rata kelas = 72.0
BUILD SUCCESSFUL (total time: 34 seconds)
```

Pertanyaan

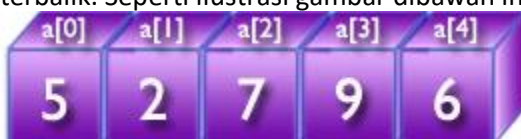
1. Pada praktikum 4 no 6. Mengapa perhitungan rata berada diluar perulangan?
2. Modifikasi program pada praktikum 4 sehingga bisa mengeluarkan output seperti gambar berikut ini

```
masukkan jumlah Mahasiswa:5
masukkan nilai Mahasiswa ke-1:90
masukkan nilai Mahasiswa ke-2:80
masukkan nilai Mahasiswa ke-3:70
masukkan nilai Mahasiswa ke-4:60
masukkan nilai Mahasiswa ke-5:50
nilai rata-rata yang lulus = 85.0
nilai rata-rata yang lulus = 60.0
BUILD SUCCESSFUL (total time: 17 seconds)
```

6. Tugas

Kerjakan tugas sesuai dengan instruksi berikut ini.

1. Buatlah program yang terdapat array dengan jumlah elemen 5, buatlah input untuk mengisi elemen array tersebut, kemudian tampilkan isi array tersebut dengan urutan terbalik. Seperti ilustrasi gambar dibawah ini.





2. Buatlah program yang menerima input jumlah elemen array, inputkan isi arraynya, kemudian tampilkan mana yang genap dan mana yang ganjil. Contoh hasil program:

```

masukkan jumlah Array:6
masukkan Array ke-0:3
masukkan Array ke-1:4
masukkan Array ke-2:5
masukkan Array ke-3:6
masukkan Array ke-4:7
masukkan Array ke-5:8
angka genap :4
angka genap :6
angka genap :8
angka ganjil :3
angka ganjil :5
angka ganjil :7
BUILD SUCCESSFUL (total time:
|

```

3. Dengan menggunakan konsep Array, Buatlah program untuk menghitung Nilai IP Semester berdasarkan Matakuliah yang Anda Ambil semester ini. Program meminta pengguna untuk memasukkan jumlah matakuliah yang diambil pada semester ini, selanjutnya program tersebut meminta pengguna untuk memasukkan nama matakuliah, bobot SKS masing-masing matakuliah, dan Nilai dari masing-masing matakuliah. Contoh hasil program:

```

masukkan jumlah MK yang anda ambil:10
masukkan nama MK 1:Kewarganegaraan
masukkan nama MK 2:Pancasila
masukkan nama MK 3:Ilmu Komunikasi Dan Organisasi
masukkan nama MK 4:Aplikasi Komputer Perkantoran
masukkan nama MK 5:Bahasa Inggris 1
masukkan nama MK 6:Konsep Teknologi Informasi
masukkan nama MK 7:Matematika Diskrit
masukkan nama MK 8:Keselamatan Dan Kesehatan Kerja
masukkan nama MK 9:Dasar Pemrograman
masukkan nama MK 10:Praktikum Dasar Pemrograman

```



```

masukan bobot SKS Kewarganegaraan:2
masukkan nilai Kewarganegaraan(A,B+,B,C+,C,D):A

masukan bobot SKS Pancasila:2
masukkan nilai Pancasila(A,B+,B,C+,C,D):B+

masukan bobot SKS Ilmu Komunikasi Dan Organisasi:2
masukkan nilai Ilmu Komunikasi Dan Organisasi(A,B+,B,C+,C,D):A

masukan bobot SKS Aplikasi Komputer Perkantoran:2
masukkan nilai Aplikasi Komputer Perkantoran(A,B+,B,C+,C,D):B

masukan bobot SKS Bahasa Inggris 1:2
masukkan nilai Bahasa Inggris 1(A,B+,B,C+,C,D):A

masukan bobot SKS Konsep Teknologi Informasi:2
masukkan nilai Konsep Teknologi Informasi(A,B+,B,C+,C,D):A

masukan bobot SKS Matematika Diskrit:2
masukkan nilai Matematika Diskrit(A,B+,B,C+,C,D):C

masukan bobot SKS Keselamatan Dan Kesehatan Kerja:2
masukkan nilai Keselamatan Dan Kesehatan Kerja(A,B+,B,C+,C,D):C+

masukan bobot SKS Dasar Pemrograman:2
masukkan nilai Dasar Pemrograman(A,B+,B,C+,C,D):A

masukan bobot SKS Praktikum Dasar Pemrograman:2
masukkan nilai Praktikum Dasar Pemrograman(A,B+,B,C+,C,D):A

IP semester Anda adalah=3.5
LD SUCCESSFUL (total time: 2 minutes 59 seconds)
    
```

Catatan : konversi nilai yang digunakan sesuai dengan peraturan polinema

Nilai Huruf	Nilai Setara
A	4
B+	3,5
B	3
C+	2,5
C	2
D	1
E	0

4. Buatlah program untuk menghapus elemen tertentu pada sebuah Array!
5. Buatlah program yang menerima input jumlah elemen array, inputkan isi arraynya, kemudian tampilkan bilangan terbesar dari isi elemen arraynya. Contoh hasil program:

```

Masukkan isi array: 4
Masukkan array ke-0: 25
Masukkan array ke-1: 10
Masukkan array ke-2: 55
Masukkan array ke-3: 15
Bilangan terbesar: 55
    
```

s

7. Materi Pengayaan

7.1 Pencarian Pada Array

Salah satu hal yang sering dilakukan pada operasi array adalah pencarian atau *searching*. Pencarian dilakukan untuk menemukan nilai tertentu pada elemen didalam array. Ada banyak algoritma searching, namun yang paling mudah adalah Linear Search.

Misalkan pada sebuah array, kita ingin mencari dimana posisi index dari sebuah array. Pada Linear Search, kita membandingkan “key” atau angka yang ingin kita cari, dengan tiap elemen yang ada didalam array. Amati gambar berikut:



Gambar 2. Linear Search.

Pada gambar diatas, key atau angka yang ingin kita cari adalah 3. Menggunakan looping kita bandingkan masing-masing elemen dari array. Dan ternyata angka 3 berada di index ke 5. Maka setelah ketemu, looping akan berhenti, dan hasil akhir dari program adalah 5 (index dimana angka 3 berada). Amati contoh program berikut:

```
import java.util.Scanner;

class LinearSearch
{
    public static void main(String[] args)
    {
        int[] arr = {6, 4, 1, 9, 7, 3, 2, 8};
        int key = 3;
        int hasil = 0;

        for(int i = 0; i < arr.length; i++)
        {
            if(key == arr[i])
            {
                hasil = i;
                break;
            }
        }

        System.out.println("Key ada di array ke: " + hasil);
    }
}
```

arr.length digunakan untuk mengetahui jumlah elemen array

Jika key ketemu, akhiri looping menggunakan perintah break

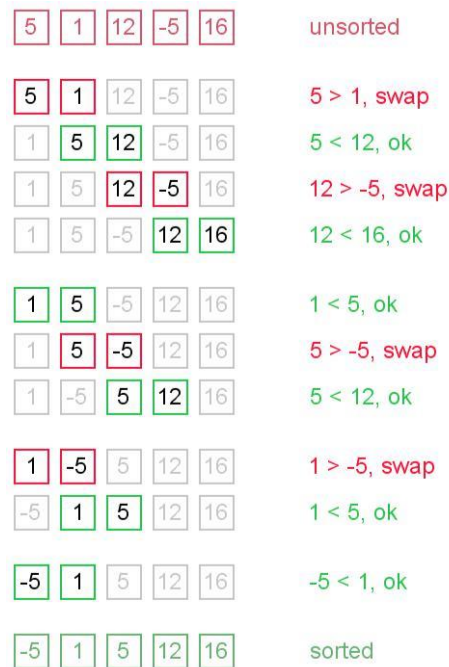
Hasil:

```
E:\Java>java LinearSearch
Key ada di array ke: 5
```

7.2 Sorting

Sorting adalah proses mengurutkan elemen array dari yang terkecil ke besar (ascending) atau sebaliknya (descending). Ada banyak algoritma untuk pengurutan, namun yang paling mudah adalah Bubble Sort.

Didalam Bubble Sort, dilakukan looping dari elemen pertama sampai elemen terakhir dari array. Kemudian tiap elemen dibandingkan dengan elemen berikutnya. Jika elemen tersebut lebih besar dari elemen berikutnya, maka akan ditukar. Amati ilustrasi berikut ini:



Gambar 3. Bubble Sort.

Berikut adalah contoh program Bubble Sort:

```
class BubbleSort
{
    public static void main(String[] args)
    {
        int[] arr = {16, 4, 10, 90, 27, 3, 12, 28};
        int n = arr.length;
        int temp = 0;

        for(int i = 0; i < n; i++)
        {
            for(int j = 1; j < (n-i); j++)
            {
                if(arr[j-1] > arr[j])
                {
                    //swap elements
                    temp = arr[j-1];
                    arr[j-1] = arr[j];
                    arr[j] = temp;
                }
            }
        }

        System.out.println("Hasil pengurutan:");

        for(int i = 0; i < n; i++)
        {
            System.out.println(arr[i]);
        }
    }
}
```

Hasilnya:

```
E:\Java>java BubbleSort
Hasil pengurutan:
3
4
10
12
16
27
28
90
```



8. Tugas Pengayaan

Kerjakan tugas sesuai dengan instruksi berikut ini.

1. Buatlah program yang menerima input jumlah array, isi array, key yang ingin dicari. Cetak ke layar index posisi elemen dari key yang ingin dicari. Contoh hasil program:

```
Masukkan jumlah elemen array: 4
Array ke 0: 5
Array ke 1: 12
Array ke 2: 25
Array ke 3: 10
Masukkan key yang ingin dicari: 12
Key ada di posisi index ke: 1
```

2. Buatlah program yang menerima input jumlah array, isi array, kemudian urutkan array tersebut, kemudian tampilkan ke layar hasil pengurutannya. Contoh hasil program:

```
Masukkan jumlah elemen array: 4
Array ke 0: 5
Array ke 1: 12
Array ke 2: 25
Array ke 3: 10 Hasil
pengurutan: 5 10 12 25
```