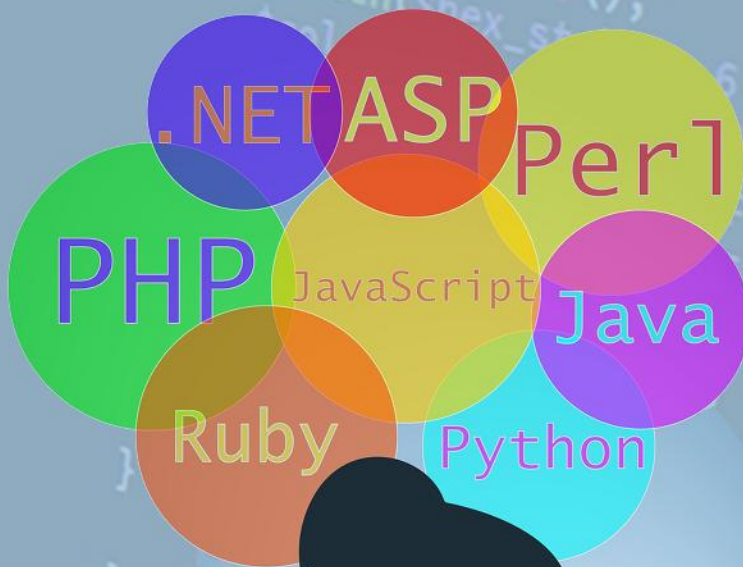




ALGORITMA DAN DASAR PEMROGRAMAN

MENGUNAKAN PHP



Sigit Susanto Putro
Delsa Yurina Cholili
Muhammad Ali Syakur

**ALGORITMA DAN DASAR
PEMROGRAMAN
MENGUNAKAN PHP**

ALGORITMA DAN DASAR PEMROGRAMAN MENGUNAKAN PHP

Sigit Susanto Putro
Delsa Yurina Cholili
Muhammad Ali Syakur



ALGORITMA DAN DASAR PEMROGRAMAN MENGGUNAKAN PHP

Penulis:
Sigit Susanto Putro
Delsa Yurina Cholili
Muhammad Ali Syakur

Editor:
Putri Jasmine

Layouter :
Tim Kreatif PRCI

Cover:
Rusli

Cetakan Pertama : November 2022

Hak Cipta 2022, pada Penulis. Diterbitkan pertama kali oleh:

Perkumpulan Rumah Cemerlang Indonesia
ANGGOTA IKAPI JAWA BARAT
Pondok Karisma Residence Jalan Raflesia VI D.151
Panglayungan, Cipedes Tasikmalaya – 085223186009

Website : www.rcipress.rcipublisher.org
E-mail : rumahcemerlangindonesia@gmail.com

Copyright © 2022 by Perkumpulan Rumah Cemerlang Indonesia
All Right Reserved

- Cet. I – : Perkumpulan Rumah Cemerlang Indonesia, 2022
; 14,8 x 21 cm
ISBN : -

Hak cipta dilindungi undang-undang
Dilarang memperbanyak buku ini dalam bentuk dan dengan
cara apapun tanpa izin tertulis dari penulis dan penerbit

Undang-undang No.19 Tahun 2002 Tentang
Hak Cipta Pasal 72

Undang-undang No.19 Tahun 2002 Tentang Hak Cipta
Pasal 72

Barang siapa dengan sengaja melanggar dan tanpa hak melakukan perbuatan sebagaimana dimaksud dalam pasal ayat (1) atau pasal 49 ayat (1) dan ayat (2) dipidana dengan pidana penjara masing-masing paling sedikit 1 (satu) bulan dan/atau denda paling sedikit Rp.1.000.000,00 (satu juta rupiah), atau pidana penjara paling lama 7 (tujuh) tahun dan/atau denda paling banyak Rp.5.000.000.000,00 (lima miliar rupiah).

Barang siapa dengan sengaja menyiarkan, memamerkan, mengedarkan, atau menjual kepada umum suatu ciptaan atau barang hasil pelanggaran hak cipta terkait sebagai dimaksud pada ayat (1) dipidana dengan pidana penjara paling lama 5 (lima) tahun dan/atau denda paling banyak Rp.500.000.000,00 (lima ratus juta rupiah)

KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas rahmat serta karunia- Nya, sehingga buku ajar ini telah dapat diselesaikan. Buku ini sebagai pedoman bagi Mahasiswa Teknik Informatika. Dalam menunjang pembelajaran mata kuliah. Penulis menyadari apabila dalam penyusunan buku ini masih terdapat kekurangan, tetapi penulis meyakini sepenuhnya bahwa sekecil apapun buku ini tetap memberikan manfaat.

Akhir kata untuk menyempurnakan buku ini, kritik dan saran dari pembaca sangatlah berguna untuk penulis kedepannya. penulis berharap buku ini dapat memberikan manfaat bagi para pembacanya.

Penulis

DAFTAR ISI

KATA PENGANTAR	I
DAFTAR ISI	I
BAB I PENGENALAN PHP DAN ALGORITMA PEMROGRAMAN	1
A. Pengenalan PHP Sebagai Bahasa Pemrograman Web	1
B. Konsep Dasar Algoritma Pemrograman	2
C. Menyiapkan Lingkungan Pengembangan PHP	3
D. Menulis dan Mengembangkan PHP Sederhana	4
E. Contoh-Contoh Algoritma Dasar Menggunakan PHP	5
BAB II DATA, TIPE DATA, VARIABEL, DAN OPERATOR	9
A. Data dalam PHP	10
B. Tipe data dalam PHP	12
C. Variabel dalam PHP	14
D. Operator dalam PHP	16
1. Operator Aritmatika	16
2. Operator Pembanding	18
3. Operator Logika	19
4. Operator String	19
5. Operator Penugasan	20
6. Operator Increment dan Decrement	21
7. Operator Array	21
8. Operator <i>Ternary</i>	22
BAB III STRUKTUR KONTROL DAN PEMILIHAN	23
A. Pernyataan Kondisional Dalam PHP	23
B. Penggunaan Operator Logika	27
C. Studi Kasus	28
BAB IV FUNGSI DAN PEMROSESAN STRING	34
A. Pengertian dan Manfaat Fungsi dalam Pemrograman PHP	34
B. Membuat dan Memanggil Fungsi dalam PHP	35
C. Penggunaan Fungsi Bawaan PHP	37
a. Fungsi untuk Pemrosesan String	37
b. Fungsi untuk Pemrosesan Array:	37
c. Fungsi untuk Pemrosesan Tanggal dan Waktu:	38
d. Fungsi untuk Pemrosesan File:	38
e. Fungsi untuk Pemrosesan Database:	38
D. Pemrosesan String	38

E. Studi Kasus	40
BAB V ARRAY DAN PEMROGRAMAN BERORIENTASI OBJEK	43
A. Pengenalan Array dan Jenis-jenisnya dalam PHP	44
B. Fungsi Array	50
C. Pemrograman Berorientasi Objek (OOP) dalam PHP	54
D. Studi Kasus	60
DAFTAR PUSTAKA	63

BAB I

PENGENALAN PHP DAN ALGORITMA PEMROGRAMAN

A. Pengenalan PHP Sebagai Bahasa Pemrograman Web

PHP (PHP: *Hypertext Preprocessor*) adalah bahasa pemrograman yang banyak digunakan dalam pengembangan aplikasi web. Dalam pengenalan ini, akan dibahas secara ringkas mengenai PHP sebagai bahasa pemrograman web.

PHP adalah bahasa pemrograman sisi server yang dirancang khusus untuk pengembangan aplikasi web. Dalam hal ini, PHP berperan sebagai bahasa skrip yang diproses di server sebelum dikirimkan ke browser pengguna. PHP berfungsi untuk menghasilkan konten dinamis di web, seperti halaman web yang dapat berinteraksi dengan pengguna, mengambil dan menyimpan data ke dalam database, dan lain sebagainya.

Keunggulan PHP sebagai bahasa pemrograman web terletak pada kemampuannya untuk berintegrasi dengan mudah dengan sejumlah basis data, seperti MySQL, PostgreSQL, dan Oracle. PHP juga mendukung berbagai protokol jaringan dan format data, seperti HTTP, FTP, XML, dan banyak lagi.

Selain itu, PHP memiliki sintaks yang sederhana dan mudah dipahami oleh pemula. Ini menjadikannya pilihan yang populer bagi banyak pengembang web yang ingin mempelajari bahasa pemrograman web pertama mereka. Selain itu, PHP memiliki komunitas pengguna yang besar dan dukungan dokumentasi yang kaya, yang memudahkan pengembang untuk mencari bantuan dan sumber daya yang dibutuhkan.

Dalam pengembangan aplikasi web dengan PHP, terdapat sejumlah kerangka kerja (*framework*) populer yang dapat digunakan, seperti Laravel, CodeIgniter, dan Symfony. Kerangka kerja ini menyediakan struktur dan komponen yang siap pakai, yang dapat mempercepat proses pengembangan aplikasi web.

Dalam kesimpulan, PHP adalah bahasa pemrograman sisi server yang digunakan secara luas dalam pengembangan aplikasi web. PHP memungkinkan pengembang untuk membuat aplikasi web yang dinamis, berinteraksi dengan basis data, dan mendukung berbagai protokol dan format data. PHP juga mudah dipelajari dan memiliki dukungan yang luas dari komunitas pengguna.

B. Konsep Dasar Algoritma Pemrograman

Konsep dasar algoritma pemrograman merupakan fondasi yang penting untuk mengembangkan solusi perangkat lunak. Algoritma adalah serangkaian langkah-langkah logis yang dirancang untuk menyelesaikan masalah atau menjalankan tugas tertentu. Berikut adalah penjelasan mengenai konsep dasar algoritma pemrograman:

1. **Pemahaman Masalah:** Langkah pertama adalah memahami masalah yang akan diselesaikan. Untuk memahami input yang dibutuhkan, keluaran yang diharapkan, dan batasan yang ada. Ini membantu dalam merumuskan strategi solusi yang tepat.
2. **Pemecahan Masalah:** Setelah memahami masalah, langkah selanjutnya adalah merancang strategi untuk memecahkannya. Pemecahan masalah melibatkan pemikiran logis dan pemahaman terhadap struktur data yang sesuai dan algoritma yang tepat. Untuk mendapatkan hasil yang diinginkan perlu merancang langkah-langkah yang diperlukan.
3. **Representasi Algoritma:** Algoritma dapat direpresentasikan dalam berbagai bentuk, seperti *pseudocode*, diagram alir, atau bahasa pemrograman. Pseudocode adalah deskripsi algoritma menggunakan bahasa yang mirip dengan bahasa pemrograman, tetapi lebih berfokus pada pemikiran logis dari pada sintaksis. Diagram alir adalah representasi visual langkah-langkah algoritma menggunakan bentuk simbol-simbol yang dapat dipahami dengan mudah.
4. **Implementasi:** Implementasikan algoritma ke dalam bahasa pemrograman yang dipilih. Ini melibatkan menuliskan kode

sesuai dengan sintaksis dan aturan bahasa pemrograman yang digunakan. Pastikan untuk mengikuti langkah-langkah dan logika yang telah dirancang dalam algoritma.

5. Pengujian dan *Debugging*: Uji algoritma dengan memberikan berbagai jenis input dan periksa apakah keluaran sesuai dengan harapan. Jika ada kesalahan atau bug dalam algoritma, lakukan debugging untuk menemukan dan memperbaikinya. Uji dan debugging membantu memastikan bahwa algoritma bekerja dengan benar dan memberikan hasil yang diinginkan.
6. Analisis dan Optimisasi: Evaluasi kinerja algoritma untuk memastikan efisiensi dan efektivitasnya. Identifikasi bagian-bagian yang dapat dioptimisasi untuk meningkatkan kecepatan atau penggunaan sumber daya. Pengoptimalan algoritma dapat mencakup perubahan struktur data, penggunaan teknik pengurangan kompleksitas waktu, atau strategi lainnya.
7. Dokumentasi: Penting untuk mendokumentasikan algoritma dengan baik. Tuliskan komentar di dalam kode yang menjelaskan langkah-langkah penting, penjelasan logika, dan tujuan dari setiap bagian. Dokumentasi yang baik memudahkan orang lain untuk memahami dan bekerja dengan algoritma.

Konsep dasar algoritma pemrograman ini membantu dalam merancang dan mengimplementasikan solusi yang efisien

C. Menyiapkan Lingkungan Pengembangan PHP

Untuk menyiapkan lingkungan pengembangan PHP, membutuhkan beberapa komponen utama, yaitu:

1. Web Server: PHP adalah bahasa pemrograman sisi server, sehingga membutuhkan web server untuk menjalankan skrip PHP. Beberapa pilihan populer untuk web server adalah Apache, Nginx, dan Microsoft IIS.
2. PHP: Unduh dan instal PHP versi terbaru dari situs resmi PHP (<https://www.php.net/downloads.php>). Ikuti petunjuk instalasi yang sesuai dengan sistem operasi yang digunakan. Pastikan untuk mengatur PATH PHP sehingga

dapat diakses dari baris perintah.

3. Database: PHP sering digunakan untuk berinteraksi dengan database. Beberapa database yang harus dipilih dan diinstall seperti MySQL, PostgreSQL, atau SQLite sesuai dengan kebutuhan. Pastikan database telah dikonfigurasi dan diatur pada pengaturan koneksi yang diperlukan.
4. Text Editor atau IDE: Pilihlah text editor atau *Integrated Development Environment* (IDE) yang sesuai dengan preferensi. Beberapa pilihan populer termasuk Visual Studio Code, Sublime Text, PhpStorm, atau NetBeans. Pastikan editor memiliki fitur penyorotan sintaks PHP.
5. Browser: Untuk menguji dan melihat hasil dari aplikasi web PHP, dalam hal ini membutuhkan browser web seperti Google Chrome, Mozilla Firefox, atau Safari. Pastikan untuk menggunakan versi terbaru dari browser untuk memastikan kompatibilitas yang baik.

Setelah menginstal dan mengkonfigurasi komponen-komponen di atas, aplikasi web menggunakan PHP siap untuk dikembangkan. Pastikan untuk memahami sintaks dan konsep dasar PHP, serta menggunakan direktori yang sesuai dalam web server untuk menyimpan file PHP. Dapat dimulai dengan membuat file PHP sederhana dan mengaksesnya melalui web browser untuk memastikan semuanya berfungsi dengan baik.

Selain itu, jika ingin menggunakan kerangka kerja PHP seperti Laravel, CodeIgniter, atau Symfony, perlu mengikuti petunjuk instalasi yang terkait dengan kerangka kerja yang dipilih. Setiap kerangka kerja biasanya memiliki persyaratan dan langkah instalasi tersendiri.

D. Menulis dan Mengembangkan PHP Sederhana

Berikut adalah contoh sederhana untuk menulis dan mengembangkan aplikasi PHP:

Buatlah file PHP baru dengan ekstensi “.php” (misalnya, “index.php”) menggunakan text editor atau IDE sesuai pilihan.

- Mulailah dengan tag PHP , dibuka dengan <?php dan

tutup dengan ?>

- Tulis kode PHP di dalam tag tersebut

```
- <?php
- echo "Selamat belajar";
- ?>
```

- Setelah menulis kode, simpan file tersebut dan pastikan simpan dengan file ekstensi .php
- Jalankan aplikasi PHP di web server lokal. Pastikan web server sudah berjalan dan file PHP disimpan di direktori yang tepat dalam web server.
- Buka web browser dan masukkan URL untuk mengakses file PHP. Misalnya, jika file PHP disimpan di direktori root server lokal, maka URL-nya mungkin menjadi <http://localhost/index.php>. Halaman web akan ditampilkan dengan pesan yang ditulis dalam tag php.

E. Contoh-Contoh Algoritma Dasar Menggunakan PHP

Berikut adalah contoh-contoh algoritma dasar yang dapat diimplementasikan menggunakan PHP:

- Algoritma Menjumlahkan Dua Bilangan

Input :

```
<?php
$bilangan1 = 10;
$bilangan2 = 5;
$jumlah = $bilangan1 + $bilangan2;
echo "Hasil penjumlahan: " . $jumlah;
?>
```

Output:

Hasil penjumlahan: 15

Penjelasan:

Algoritma Menjumlahkan Dua Bilangan:

1. Pada contoh ini, memiliki dua variabel, '\$bilangan1' dan '\$bilangan2', yang masing-masing memiliki nilai 10 dan 5.

2. Dengan menggunakan operator penjumlahan (+) untuk menjumlahkan kedua bilangan tersebut dan menyimpan hasilnya pada variabel `$jumlah`.
 3. Hasil penjumlahan ditampilkan menggunakan perintah `'echo'`.
- Algoritma Pemilihan Bilangan Terbesar

Input :

```
<?php
$a = 5;
$b = 7;
if ($a > $b) {
    echo "Bilangan terbesar: " . $a;
} else {
    echo "Bilangan terbesar: " . $b;
}
?>
```

Output :

← → ↻ ⓘ localhost/percobaan/percobaan2.php

Bilangan terbesar: 7

- Algoritma Menentukan Bilangan Prima

Input :

```
<?php
$n = 17;
$prima = true;
for ($i = 2; $i <= $n/2; $i++) {
    if ($n % $i == 0) {
        $prima = false;
        break;
    }
}
if ($prima) {
    echo $n . " adalah bilangan prima";
} else {
    echo $n . " bukan bilangan prima";
} ?>
```

Output :



localhost/percobaan/percobaan2.php

17 adalah bilangan prima

- Algoritma Menghitung Faktorial

Input :

```
<?php
$n = 5;
$faktorial = 1;

for ($i = 1; $i <= $n; $i++) {
    $faktorial *= $i;
}

echo "Faktorial dari " . $n . " adalah " . $faktorial;
?>
```

Output :



localhost/percobaan/percobaan2.php

Faktorial dari 5 adalah 120

- Algoritma Pengurutan Array Secara *Ascending*

Input :

```
<?php
$array = [5, 2, 8, 1, 4];
$n = count($array);

for ($i = 0; $i < $n - 1; $i++) {
    for ($j = $i + 1; $j < $n; $j++) {
        if ($array[$i] > $array[$j]) {
            $temp = $array[$i];
```

```

        $array[$i] = $array[$j];
        $array[$j] = $temp;
    }
}
}

echo "Array setelah diurutkan: ";
foreach ($array as $value) {
    echo $value . " ";
}
?>

```

Output :

← → ↻ ⓘ localhost/percobaan/percobaan2.php

Array setelah diurutkan: 1 2 4 5 8

- Algoritma Mencari Elemen Tertentu dalam Array
Input :

```

<?php
$array = [5, 2, 8, 1, 4];
$target = 8;
$found = false;

foreach ($array as $value) {
    if ($value == $target) {
        $found = true;
        break;
    }
}

if ($found) {
    echo $target . " ditemukan dalam array";
} else {
    echo $target . " tidak ditemukan dalam array";
}
?>

```

Output :

← → ↻ ⓘ localhost/percobaan/percobaan2.php

8 ditemukan dalam array

BAB II

DATA, TIPE DATA, VARIABEL, DAN OPERATOR

A. Data dalam PHP

Data adalah kumpulan fakta, informasi, atau statistik yang dikumpulkan, diatur, dan disimpan untuk analisis atau penggunaan berikutnya. Data dapat berupa teks, gambar, suara, angka, atau representasi informasi lainnya. Biasanya digunakan sebagai dasar untuk pemodelan, pengambilan keputusan, atau penciptaan pengetahuan baru.

Data dapat terstruktur atau tidak terstruktur. Data terstruktur mengacu pada data dengan format yang ditentukan, seperti basis data dengan tabel dan kolom yang terdefinisi. Sebaliknya, data tidak terstruktur mengacu pada data dengan format yang tidak ditentukan, seperti teks bebas, pesan, atau file gambar.

Data dapat berupa string, angka, boolean, array, objek, atau tipe data lainnya yang disimpan atau digunakan dalam program PHP. Contoh data PHP adalah *"Hello, world!"*, 42, true, dan [1, 2, 3], sedangkan "tipe data" merujuk pada klasifikasi atau kategori yang menggambarkan jenis nilai yang dapat disimpan dalam variabel atau digunakan dalam program PHP. PHP adalah bahasa pemrograman berbasis tipe data dinamis, yang berarti tipe data ditentukan secara otomatis berdasarkan nilai yang diberikan ke variabel. Beberapa tipe data yang umum digunakan dalam PHP antara lain: String:

Dalam PHP, beberapa tipe data yang umum digunakan antara lain: String, Tipe data integer, Tipe data boolean, tipe data float, Tipe data array, Tipe data objek, Null. Jadi, "data" biasanya mengacu pada nilai atau informasi yang digunakan dalam program, sedangkan "tipe data" mengacu pada klasifikasi nilai dalam PHP berdasarkan jenisnya.

Berikut adalah beberapa contoh cara untuk bekerja dengan data dalam PHP:

1. Variabel: Dapat menggunakan variabel untuk menyimpan data dalam PHP. Contoh penggunaan variabel:

```
$nama = "Bunga Mawar";  
$umur = 25;
```

2. Array: Array digunakan untuk menyimpan sekumpulan nilai dalam satu variabel. Ada beberapa jenis array dalam PHP, seperti array numerik, asosiatif, dan multidimensi. Contoh penggunaan array:

```
// Array numerik  
$angka = array(1, 2, 3, 4, 5);  
// Array asosiatif  
$mahasiswa = array(  
    "nama" => "Bunga Mawar",  
    "umur" => 25,  
    "jurusan" => "Informatika"  
);  
// Array multidimensi  
$matrix = array(  
    array(1, 2, 3),  
    array(4, 5, 6),  
    array(7, 8, 9)  
);
```

3. String: String adalah tipe data yang digunakan untuk menyimpan teks. Melakukan operasi dan manipulasi string menggunakan berbagai fungsi yang disediakan oleh PHP.

Contoh penggunaan string:

```
$nama = "Bunga Mawar";  
$panjang = strlen($nama); // Menghitung  
panjang string  
$huruf_pertama = $nama[0]; // Mendapatkan  
huruf pertama  
$huruf_terakhir = substr($nama, -1); //  
Mendapatkan huruf terakhir
```

4. Objek: PHP juga mendukung pemrograman berorientasi objek, yang memungkinkan membuat kelas dan objek. Objek