

Mata Kuliah: Struktur Data
Pertemuan ke 10
Topik: Algoritma Sorting

Pendahuluan

Algoritma sorting merupakan teknik pengurutan data yang sangat penting dalam berbagai bidang seperti Artificial Intelligence (AI), Data Mining, Big Data, dan Internet of Things (IoT). Pada materi ini, kita akan membahas beberapa algoritma sorting lanjutan dan aplikasinya dalam berbagai bidang.

Algoritma Sorting

1. Merge Sort

Merge Sort adalah algoritma sorting yang menggunakan teknik divide dan conquer. Algoritma ini membagi data menjadi dua bagian, kemudian menggabungkan dan mengurutkan hasilnya.

Contoh Kode PHP :

```
function mergeSort($arr) {  
    $n = count($arr);  
    if ($n <= 1) return $arr;  
    $mid = floor($n / 2);  
    $left = mergeSort(array_slice($arr, 0, $mid));  
    $right = mergeSort(array_slice($arr, $mid));  
    return merge($left, $right);  
}  
  
function merge($left, $right) {  
    $result = array();  
    while (count($left) > 0 && count($right) > 0) {  
        if ($left[0] <= $right[0]) {  
            $result[] = array_shift($left);  
        } else {  
            $result[] = array_shift($right);  
        }  
    }  
    $result = array_merge($result, $left);  
    $result = array_merge($result, $right);  
    return $result;  
}
```

Aplikasi :

- Pengurutan data pelanggan dalam aplikasi rekomendasi produk (AI)
- Pengurutan data transaksi keuangan untuk analisis pola pembelian (Data Mining)
- Pengurutan data sensor IoT dalam aplikasi monitoring lingkungan (Big Data dan IoT)

2. Quick Sort

Quick Sort adalah algoritma sorting yang menggunakan teknik divide dan conquer. Algoritma ini memilih pivot, kemudian membagi data menjadi dua bagian berdasarkan pivot tersebut.

Contoh Kode PHP :

```
function quickSort($arr) {
    $n = count($arr);
    if ($n <= 1) return $arr;
    $pivot = $arr[0];
    $smaller = array();
    $greater = array();
    for ($i = 1; $i < $n; $i++) {
        if ($arr[$i] <= $pivot) {
            $smaller[] = $arr[$i];
        } else {
            $greater[] = $arr[$i];
        }
    }
    $smaller = quickSort($smaller);
    $greater = quickSort($greater);
    return array_merge($smaller, array($pivot), $greater);
}
```

Aplikasi:

- Pengurutan data pengguna dalam aplikasi rekomendasi konten (AI)
- Pengurutan data transaksi keuangan untuk analisis pola pembelian (Data Mining)
- Pengurutan data sensor kecepatan dalam aplikasi pengendalian lalu lintas (IoT)

3. Heap Sort

Heap Sort adalah algoritma sorting yang menggunakan teknik heap. Algoritma ini membangun heap, kemudian mengurutkan data berdasarkan heap tersebut.

Contoh Kode PHP:

```
function heapSort($arr) {
    $n = count($arr);
    for ($i = floor($n / 2) - 1; $i >= 0; $i--) {
        heapify($arr, $n, $i);
    }
    for ($i = $n - 1; $i >= 0; $i--) {
        $temp = $arr[0];
        $arr[0] = $arr[$i];
        $arr[$i] = $temp;
        heapify($arr, $i, 0);
    }
    return $arr;
}

function heapify($arr, $n, $i) {
    $largest = $i;
    $left = 2 * $i + 1;
    $right = 2 * $i + 2;
    if ($left < $n && $arr[$i] < $arr[$left]) {
        $largest = $left;
    }
    if ($right < $n && $arr[$largest] < $arr[$right]) {
        $largest = $right;
    }
    if ($largest != $i) {
        $temp = $arr[$i];
        $arr[$i] = $arr[$largest];
        $arr[$largest] = $temp;
        heapify($arr, $n, $largest);
    }
}
```

Aplikasi:

- Pengurutan data sensor suhu dalam aplikasi pengendalian suhu ruangan (IoT)
- Pengurutan data log dalam aplikasi monitoring sistem (Big Data)

4. Radix Sort:

Pengurutan berdasarkan digit.

- Kelebihan: Efisien untuk data numerik.
- Kekurangan: Tidak efisien untuk data non-numerik.
- Aplikasi: Pengurutan data IP address, pengurutan data sensor suhu.

Contoh Kode PHP:

```
function radixSort($arr) {  
    $max = max($arr);  
    $exp = 1;  
    while (floor($max / $exp) > 0) {  
        $bucket = array_fill(0, 10, array());  
        foreach ($arr as $value) {  
            $bucket[floor($value / $exp) % 10][] = $value;  
        }  
        $arr = array();  
        foreach ($bucket as $values) {  
            $arr = array_merge($arr, $values);  
        }  
        $exp *= 10;  
    }  
    return $arr;  
}
```

5. Timsort:

Kombinasi Merge Sort dan Insertion Sort.

- Kelebihan: Efisien dan stabil.
- Kekurangan: Kompleksitas tinggi.
- Aplikasi: Pengurutan data pengguna dalam aplikasi rekomendasi konten, pengurutan data log.

Contoh Kode PHP:

```
function timsort($arr) {
    $min_run = 32;
    $n = count($arr);
    for ($i = 0; $i < $n; $i += $min_run) {
        insertionSort(array_slice($arr, $i, $min_run));
    }
    $size = $min_run;
    while ($size < $n) {
        for ($i = 0; $i < $n; $i += $size * 2) {
            merge(array_slice($arr, $i, $size), array_slice($arr, $i + $size, $size));
        }
        $size *= 2;
    }
    return $arr;
}
```

6. Dual-Pivot Quick Sort:

Variasi Quick Sort dengan dua pivot.

- Kelebihan: Efisien dan stabil.
- Kekurangan: Kompleksitas tinggi.
- Aplikasi: Pengurutan data transaksi keuangan, pengurutan data sensor IoT.

Contoh Kode PHP:

```
function dualPivotQuickSort($arr) {
    if (count($arr) <= 1) {
        return $arr;
    }
    $pivot1 = $arr[0];
    $pivot2 = $arr[count($arr) - 1];
    $less = array();
    $equal1 = array();
    $equal2 = array();
    $greater = array();
    for ($i = 1; $i < count($arr) - 1; $i++) {
        if ($arr[$i] < $pivot1) {
            $less[] = $arr[$i];
        } elseif ($arr[$i] == $pivot1) {
            $equal1[] = $arr[$i];
        } elseif ($arr[$i] < $pivot2) {

```

```

$equal2[] = $arr[$i];
} elseif ($arr[$i] == $pivot2) {
$equal2[] = $arr[$i];
} else {
$greater[] = $arr[$i];
}
}
$less = dualPivotQuickSort($less);
$greater = dualPivotQuickSort($greater);
return array_merge($less, $equal1, array($pivot1), $equal2, array($pivot2), $greater);
}
}

```

Aplikasi Algoritma Sorting :

1. Pengurutan data transaksi keuangan (Introsort, Dual-Pivot Quick Sort)
2. Pengurutan data sensor IoT (Radix Sort, Timsort)
3. Pengurutan data pengguna dalam aplikasi rekomendasi konten (Timsort)
4. Pengurutan data log (Timsort)
5. Pengurutan data IP address (Radix Sort)

Contoh Penggunaan :

1. Pengurutan data transaksi keuangan:

```
$arr = array(100, 50, 200, 150, 300);
```

```
$sorted_arr = introsort($arr);
```

```
print_r($sorted_arr);
```

2. Pengurutan data sensor suhu:

```
$arr = array(25, 30, 20, 35, 40);
```

```
$sorted_arr = radixSort($arr);
```

```
print_r($sorted_arr);
```