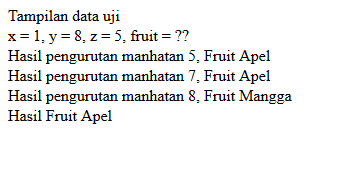
Hasil uji aplikasi



|  |  |
| --- | --- |
| <?php  $data = array();  // data latih  $data = [  // x, y, z, fruit  [4, 6, 7, 0],  [2, 4, 9, 0],  [4, 2, 4, 1],  [3, 5, 8, 1],  [5, 5, 7, 0],  [3, 9, 3, 0],  [4, 3, 9, 1] ];  // data test fruit ??  // x. y, z, ??  $test = array(1, 8, 5);  $man = new manhattan(); // memanggil kelas manhattan  $hasilRumusDiff = array(); // deskripsi hasil manhattan berupa array  $hasilPengurutan = array(); // desk hasil pengurutan  $hasilRumusDiff = $man->valueDiff($data, $test); // menyimpan hasil perhitungan manhattan  // mengurutkan angka dari kecil ke besar  $hasilPengurutan = $man->urutkan($hasilRumusDiff);  //$man->jumlahK($hasilPengurutan, 3);  /// tampilan  echo "Tampilan data uji <br>";  echo "x = " . $test[0] . ", y = " . $test[1] . ", z = " . $test[2] . ", fruit = ??";  echo "<br>";  $sum=0;  for($a=0; $a < 3; $a++) {  echo "Hasil pengurutan manhatan " . $hasilPengurutan[$a][0];  if($hasilPengurutan[$a][1] == 0) {  echo ", Fruit Apel <br>";  } else {  echo ", Fruit Mangga <br>";  }  $sum += $hasilPengurutan[$a][1];  }  if($sum < 2 ){  echo "Hasil Fruit Apel";  } else {  echo "Hasil Fruit Mangga";  }  //var\_dump($hasilPengurutan);  class manhattan {    public function valueDiff($latih = array(), $uji = array()) {  // menghitung perulangan data latih  // $array = array();  // count(latih) -- menghitung jumlah data latih $m = 5  for($m = 0; $m < count($latih); $m++){  // menghitung jumlah kolom pada data uji ---  // deskripsi mula-mula julah manhatan bernilai 0  $sumManhattan[$m] = 0;  // count(uji) -- menghitung jumlah data uji $n = 3  for($n = 0; $n < count($uji); $n++){  // menghitung nilai xDiff, yDiff, zDiff  $nDiff[$m][$n] = $latih[$m][$n] - $uji[$n];  // memanggil fungsi absolute  $absDiff[$m][$n] = $this->absolute($nDiff[$m][$n]);  // menjumlahkan nilai xDiff + yDiff + zDiff = Manhattan  $sumManhattan[$m] += $absDiff[$m][$n];    } // end for n....  // echo "<br>";    $array[$m] = array($sumManhattan[$m], $latih[$m][3]);  //var\_dump($array);  } // end for m...    return $array;    // var\_dump($uji);    } // end func...    public function absolute($v) {  // jika nilai nDiff kurang dari 0  if($v < 0) {  $val = ($v \* $v);  $abs = sqrt($val); //sqrt(x) -- pencabutan akar    } else {  $abs = $v;  }  return $abs; // kembalikan nilai abs  }    // fungsi pengurutan  public function urutkan($nilai = array()) {  // pengurutan menggunakan bable sort  for($i=0; $i < count($nilai); $i++) { // n  // echo $nilai[$i][0];  for($j=$i; $j < count($nilai); $j++) { // n    if($nilai[$i][0] > $nilai[$j][0]) {  $save = $nilai[$i][0]; // n - 1  $nilai[$i][0] = $nilai[$j][0]; // n - 1  $nilai[$j][0] = $save; // n - 1  } // end if....  } // end for $j ....    } // end for $i ....  return $nilai;  }    public function jumlahK($nilai = array(), $k ){  for($a = 0; $a < $k; $a++ ){  // echo $nilai[$k][1];  // echo "<br>";  var\_dump($nilai[$k]);  }  }  }  ?> |  |