

Source Code Algoritma Dijkstra

```
1 <?php
2 class Dijkstra
3 {
4     /**
5     * MENCARI 1 JALUR TERPENDEK DENGAN ALGORITMA DIJKSTRA DARI GRAPH ARRAY
6     * GRAPH ARRAY SEPERTI DIBAWAH INI :
7     [0][0] = '1->10';
8     [0][1] = '2->11';
9     [0][2] = '3->40';
10
11     [1][0] = '0->10';
12     [1][1] = '2->55';
13     [1][2] = '4->20';
14
15     [2][0] = '0->11';
16     [2][1] = '1->55';
17     [2][3] = '1->54';
18
19     [3][0] = '4->89';
20     [3][0] = '4->89';
21
22     [4][0] = '0->90';
23     [4][3] = '3->89';
24
25     * GRAPH ARRAY YANG SUDAH DIKERJAKAN DALAM MPERHITUNGAN ALGORITMA DIJKSTRA :
26     [0][0] = '1->10->y';
27     [0][1] = '2->11->y';
28     [0][2] = '3->40->y';
29     [1][0] = '0->10->y';
30     ...
31     * @param arg_graph array[]
32     * @param simpulAwal int
33     * @param simpulTujuan int
34     * @return string, contoh 0->1->4
35     */
36     function jalurTerpendek($arg_graph, $simpulAwal, $simpulTujuan){
37
38         // SIMPUL_AWAL == SIMPUL_TUJUAN, MAKA DIE()
39         // 0 == 0
40         if($simpulAwal == $simpulTujuan){
41             return json_encode(['status'=>'error','error'=>'lokasi_anda_sudah_dekat','teks'=>'Lokasi Anda Sudah Dekat','content'=>'']);
42         }
43
44         // SIMPUL_AWAL OR SIMPUL_TUJUAN NOT FOUND
45         if(!array_key_exists($simpulAwal, $arg_graph) || !array_key_exists($simpulTujuan, $arg_graph)){
46             return print_r(json_encode(['status'=>'error','error'=>'simpul_input_tidak_ditemukan','teks'=>'could not find the input : $simpulAwal or $simpulTujuan', 'content'=>'']));
47         }
48
49         $graph = $arg_graph;
50         $simpul_awal = $simpulAwal;
51         $simpul_maju = $simpulAwal;
52         $simpul_tujuan = $simpulTujuan;
53         $jml_simpul = count($arg_graph);
54
55         /**
56         * TANDA SIMPUL YANG AKAN DIKERJAKAN DENGAN TANDA BINTANG (*)
57         * MISAL SOAL : CARI JALUR TERPENDEK DARI SIMPUL 0 KE SIMPUL 4 !
58
59         | 0(*) | 1(*) | 2(*) | 3 | 4(*) | <-- KOLOM SIMPUL
60
61         | 0->1=10 | 1->0=10 | 2->0=11 | 3->4=89 | 4->0=90 | <-- BARIS; BOBOT = 90
62         | 0->2=11 | 1->2=55 | 2->1=55 | | 4->3=89 |
63         | 0->3=40 | 1->4=20 | 2->3=54 | | |
64
65         * MAKA HASILNYA $simpulYangDikerjakan = array(0, 1, 2, 4);
66         */
67         $simpulYangDikerjakan = array();
68
69         // UNTUK MENYIMPAN NILAI-NILAI * YANG DITANDA
70         $simpulYanoSudahDikerjakan bawah = array();
```

```

68 // UNTUK MENYIMPAN NILAI-NILAI * YANG DITANDAI
69 $simpulYangSudahDikerjakan_bawah = array();
70
71 $nilaiSimpulYgDitandai = 0;
72 $nilaiSimpulFixYgDitandai = 0;
73
74
75 // #HANDLE PERULANGAN
76 // PERULANGAN INI TIDAK AKAN BERHENTI (--$perulangan;) SAMPAI ALGORITMA DIJKSTRA MENEMUKAN 1 JALUR TERPENDEK
77 //
78 for($perulangan = 0; $perulangan < 1; $perulangan++)
79 {
80 // UNTUK MNDAPATKAN 1 BOBOT PALING MINIMUM DARI SETIAP SIMPUL
81 $perbandinganSemuaBobot = array();
82
83 // DAFTARKAN SIMPUL pertama YANG AKAN DIKERJAKAN KE DALAM ARRAY
84 if(!in_array($simpul_maju, $simpulYangDikerjakan)){
85     array_push($simpulYangDikerjakan, $simpul_maju);
86 }
87
88 /**
89  * PERULANGAN (KOLOM) SIMPUL-SIMPUL YANG DITANDAI
90  *-----
91  * | 0(*) | 1(*) | 2(*) | 3 | 4(*) | <-- KOLOM SIMPUL
92  *-----
93  * | 0->1=10 | 1->0=10 | 2->0=11 | 3->4=89 | 4->0=90 |
94  * | 0->2=11 | 1->2=55 | 2->1=55 | | 4->3=89 |
95  * | 0->3=40 | 1->4=20 | 2->3=54 | | |
96  *-----
97  * PERULANGANNYA : array(0, 1, 2, 4);
98  */
99 for($perulanganSimpul = 0; $perulanganSimpul < count($simpulYangDikerjakan); $perulanganSimpul++)
100 {
101 // HITUNG JUMLAH BARIS PER KOLOM SIMPUL
102 // 0(*) = 3 BARIS; (0->1=10; 0->2=11; 0->3=40)
103 // 1(*) = 3 BARIS:

```

```

104 // 2(*) = 3 BARIS;
105 // 4(*) = 2 BARIS;
106 $jumlah_baris = count($graph[ $simpulYangDikerjakan[$perulanganSimpul] ]);
107
108 // TAMPUNG BOBOT MINIMUM DARI SETIAP KOLOM SIMPUL BERDASARKAN BARIS SCR URUT[0][0],[0][1] DST
109 $bobot_baris = array();
110
111 // JUMLAH BARIS YANG BELUM DIKERJAKAN
112 $baris_belum_dikerjakan = 0;
113
114
115 /**
116  * PERULANGAN BARIS TABEL, CARI BOBOT DARI 1 KOLOM SIMPUL
117  *-----
118  * | 0(*) | <-- KOLOM SIMPUL
119  *-----
120  * | 0->1=10 | <-- baris; bobot = 10;
121  * | 0->2=11 | <-- baris; bobot = 11;
122  * | 0->3=40 | <-- baris; bobot = 40;
123  *-----
124  */
125 for($start_baris = 0; $start_baris < $jumlah_baris; $start_baris++)
126 {
127 // AMBIL VALUE ARRAY graph[][]
128 // MISAL graph[0][0] = '1->10'; VALUENYA BERARTI '1->10'
129 $ruas_dan_bobot = $graph[ $simpulYangDikerjakan[$perulanganSimpul] ][$start_baris]; //pasti berurutan [0][0],[0][1] dst
130
131 // PECAH RUAS & BOBOT BERDASARKAN '->'
132 // RUAS : explode[0] = 1
133 // BOBOT : explode[1] = 10
134 $explode = explode('->', $ruas_dan_bobot);
135
136 /**
137  * CARI BOBOT YG BELUM DIKERJAKAN (YG TIDAK ADA TANDA ->Y)
138  * MISAL :
139  * [0][0] = '1->10->y';

```

```

138      * MISAL :
139      [0][0] = '1->10->y';
140      [0][1] = '2->11'; <---- ini yang belum dikerjakan / tidak ada tanda y
141      */
142      if(count($explode) == 2)
143      {
144          // TOTAL BARIS YG BELUM ->Y
145          $baris_belum_dikerjakan += 1;
146
147          // CEK KOLOM SIMPUL APAKAH SUDAH DITANDAI (*) APA BLOM, KLO UDH BERARTI NILAI * TIDAK DITAMBAH LAGI / 0
148          // KALO BLM DITANDAI, BERARTI NILAI * BERNILAI $nilaiSimpulYgDitandai
149          if(!empty($simpulYangSudahDikerjakan_bawah))
150          {
151              if(in_array($simpulYangDikerjakan[$perulanganSimpul], $simpulYangSudahDikerjakan_bawah)){
152                  $nilaiSimpulYgDitandai = 0;
153              }else{
154                  $nilaiSimpulYgDitandai = $nilaiSimpulFixYgDitandai;
155              }
156          }
157
158          /**
159          * MASUKKAN BOBOT BARIS YANG SUDAH DIUPDATE KE DALAM ARRAY
160          * ILUSTRASI MENGUPDATE BOBOT DALAM TABEL :
161          -----
162          |      0(*)      | <-- KOLOM SIMPUL
163          -----
164          | 0->1=10 (13) | <-- bobot diupdate = 13;
165          | 0->2=11 (12) | <-- bobot diupdate = 12;
166          | 0->3=40 (45) | <-- bobot diupdate = 45;
167          -----
168          */
169          array_push($bobot_baris, ($explode[1]+$nilaiSimpulYgDitandai)); // (bobot+0) or (bobot+232)
170
171          // UPDATE JUGA BOBOT BARIS PADA graph[][]
172          // MISAL : '1->13'
173          $graph[ $simpulYangDikerjakan[$perulanganSimpul] ][$start_baris] = $explode[0] . "->" . $explode[1] . $nilaiSimpulYgDitandai

```

```

174      }
175      }
176
177
178      // JIKA ADA BARIS DI KOLOM BELUM ->Y SEMUA, MAKA LAKUKAN IF DI BAWAH INI :
179      if($baris_belum_dikerjakan > 0)
180      {
181          // DAPATKAN BOBOT MINIMUM
182          for($index_bobot = 0; $index_bobot < count($bobot_baris); $index_bobot++){
183              if($bobot_baris[$index_bobot] <= $bobot_baris[0]){
184                  $bobot_baris[0] = $bobot_baris[$index_bobot];
185              }
186          }
187
188          // BOBOT TERKECIL DARI SETIAP KOLOM SIMPUL
189          array_push($perbandinganSemuaBobot, $bobot_baris[0]);
190
191      }// end if jika ->y atau ->t belum semua dikerjakan
192      else{
193          // Jika baris di kolom sudah ->y semua, maka lakukan else di bawah ini
194          // System.out.println("=====|Baris sudah ->y semua|=====");
195      }
196
197      // DAFTARKAN SIMPUL SIMPUL YANG baru selesai DIKERJAKAN
198      if(!in_array($simpulYangDikerjakan[$perulanganSimpul], $simpulYangSudahDikerjakan_bawah)){
199          array_push( $simpulYangSudahDikerjakan_bawah, $simpulYangDikerjakan[$perulanganSimpul] );
200      }
201      }// end for perulanganSimpul
202
203
204      /**
205      * DAPATKAN 1 BOBOT PALING MINIMUM DARI SIMPUL YG DITANDAI (*)
206      -----

```

```

204      /**
205      * DAPATKAN 1 BOBOT PALING MINIMUM DARI SIMPUL YG DITANDAI (*)
206      *
207      * -----
208      * | 0(*) | 1(*) | 2(*) |
209      * -----
210      * | 0->1=10(*) | 1->0=10 | 2->0=11 |
211      * | 0->2=11(*) | 1->2=55 | 2->1=55 |
212      * | 0->3=40 | 1->4=20(*) | 2->3=54 |
213      * -----
214      */
215      for($min_indexAntarRobotYgDitandai = 0; $min_indexAntarRobotYgDitandai < count($perbandinganSemuaRobot); $min_indexAntarRobotYgDitandai
216      ++
217      {
218          if($perbandinganSemuaRobot[$min_indexAntarRobotYgDitandai] <= $perbandinganSemuaRobot[0]){
219              $perbandinganSemuaRobot[0] = $perbandinganSemuaRobot[$min_indexAntarRobotYgDitandai];
220          }
221      }
222      // DAPATKAN INDEX SIMPUL+BOBOTNYA YG ASLI DARI SIMPUL YG DITANDAI
223      $indexAwalAsli = 0; // index simpulnya
224      $baris_belum_dikerjakan1 = 0;
225      $dapat_indexAsliRobot = 0;
226      $simpul_lama = 0;
227      // $length_baris = count($graph[$simpulYangDikerjakan[$indexAwalAsli]);
228
229      foreach($simpulYangDikerjakan as $idx => $v)
230      {
231          /**
232          * JUMLAH BARIS per KOLOM SIMPUL
233          * -----
234          * | 0(*) | <-- KOLOM SIMPUL
235          * -----
236          * | 0->1=10 (13) | <-- baris 1
237          * | 0->2=11 (12) | <-- baris 2
238          * | 0->3=40 (45) | <-- baris 3
239          */

```

```

239      */
240      $length_baris = $graph[$simpulYangDikerjakan[$idx]];
241
242      for($baris1 = 0; $baris1 < $length_baris; $baris1++)
243      {
244          if( isset($graph[ $simpulYangDikerjakan[$indexAwalAsli] ][$baris1]) )
245          {
246              $bobot_baris_dan_ruas1 = $graph[ $simpulYangDikerjakan[$indexAwalAsli] ][$baris1];
247              $explodel = array();
248              $explodel = explode('-', $bobot_baris_dan_ruas1);
249              if(count($explodel) == 2)
250              {
251                  if($perbandinganSemuaRobot[0] == $explodel[1])
252                  {
253                      $dapat_indexAsliRobot = $baris1;
254                      $simpul_lama = $simpulYangDikerjakan[$indexAwalAsli];
255                      $simpul_maju = $explodel[0];
256                      $baris_belum_dikerjakan1 += 1;
257                  }
258              } // end if cek ->y atau ->t
259          } // end if cek baris != null
260          else{
261              break;
262          }
263      } // end for limit baris = 100
264
265      $indexAwalAsli++; // index simpul di tambah 1
266      } // end for simpul yang dikerjakan
267
268      // BULETIN BOBOT MINIMUM YANG UDH DIDAPAT dan HAPUS RUAS YANG BERRHUBUNGAN
269      if($baris_belum_dikerjakan1 > 0){
270          $graph[$simpul_lama][$dapat_indexAsliRobot] = $graph[$simpul_lama][$dapat_indexAsliRobot] . "->y";
271      }
272
273      // HAPUS RUAS LAIN

```

```

272
273 // HAPUS RUAS LAIN
274 for($min_kolom = 0; $min_kolom < $jml_simpul; $min_kolom++)
275 {
276 // JUMLAH BARIS PER KOLOM SIMPUL
277 $length_baris1 = count($graph[$min_kolom]);
278
279 for($min_baris = 0; $min_baris < $length_baris1; $min_baris++)
280 {
281 if(isset($graph[$min_kolom][$min_baris]))
282 {
283 $ruasYgAkanDihapus = $graph[$min_kolom][$min_baris];
284 $explode3 = explode('->', $ruasYgAkanDihapus);
285 if(count($explode3) == 2){
286 if($explode3[0] == $simpul_maju){
287 $graph[$min_kolom][$min_baris] = $graph[$min_kolom][$min_baris]:"->";
288 }
289 }
290 } // end if cek baris != null
291 } // end for baris
292 } // end for kolom
293 } // end if cek baris_belum_dikerjakan sudah ->y atau ->t semua apa belum
294
295 // =====
296 // # JIKA ALUR GRAPH ANDA SALAH, MAKA DIE() SAMPAI IF INI
297 // =====
298 if(!isset($perbandinganSemuaBotol[0]))
299 return json_encode(['status'=>'error', 'error'=>'alur_graph_anda_salah', 'teks'=>'Alur graph Anda Salah', 'content'=>'']);
300
301 // NILAI * YG DITANDAI
302 $nilaiSimpulFixYgDitandai = $perbandinganSemuaBotol[0];
303
304 // LOOPING $perulangan lagi jika SIMPUL_MAJU != SIMPUL_TUJUAN
305 if($simpul_maju != $simpul_tujuan){
306 --$perulangan;
307

```

```

306 if($simpul_maju != $simpul_tujuan){
307 --$perulangan;
308 }
309 else{
310 break; // akhiri perulangan
311 }
312 } // end for handle perulangan
313
314
315
316 // TARUH SIMPUL GABUNGAN KE ARRAY; MISAL : SIMPUL 6-10
317 $gabungSimpulPilihan = array();
318 for($h = 0; $h < $jml_simpul; $h++)
319 {
320 // JUMLAH BARIS PER KOLOM SIMPUL
321 $length_baris2 = count($graph[$h]);
322
323 for($n = 0; $n < $length_baris2; $n++)
324 {
325 if(isset($graph[$h][$n]))
326 {
327 $str_graph = $graph[$h][$n];
328 if( substr($str_graph, (strlen($str_graph)-1), strlen($str_graph)) == "y" ){
329 $explode4 = explode('->', $graph[$h][$n]);
330 $simpulGabung = $h . "-" . $explode4[0];
331
332 array_push($gabungSimpulPilihan, $simpulGabung);
333 }
334 } // end if cek isi graph != null
335 } // end for looping baris
336 } // end looping kolom (simpul)
337
338
339 // UNTUK MEMASUKKAN SIMPUL YG SUDAH DIURUTKAN (DARI SIMPUL TUJUAN KE SIMPUL AWAL). (NANTI DIREVERSE ARRAYNYA)
340 $simpulFix_finish = array();
341

```

```

341
342 // MASUKKAN PERTAMA KALI SIMPUL TUJUAN (SIMPUL AKHIR) KE ARRAY DGN INDEX 0. (NANTI DIBALIK(REVERSE) ARRAYNYA)
343 array_push($simpulFix_finish, $simpul_tujuan);
344
345 $simpul_explode = $simpul_tujuan;
346 for($v = 0; $v < 1; $v++)
347 {
348     for($w = 0; $w < count($gabungSimpulPilihan); $w++)
349     {
350         $explode_simpul = $gabungSimpulPilihan[$w];
351         $explode5 = explode('-', $explode_simpul);
352         if($simpul_explode == $explode5[1])
353         {
354             array_push($simpulFix_finish, $explode5[0]);
355             $simpul_explode = $explode5[0];
356         }
357         if($simpul_explode == $simpul_awal){
358             break;
359         }
360     }
361     if($simpul_awal != $simpul_explode){
362         --$v;
363     }else{
364         break;
365     }
366 }
367 // end for cari simpul yang dibuletin lalu dibandingkan dgn simpul_tujuan
368
369
370 // ARRAY DI BALIK INDEXNYA; JADI SIMPUL TUJUAN DI PINDAH POSISI KE AKHIR INDEX ARRAY
371 $simpulFix_finish_reverse = array_reverse($simpulFix_finish);
372
373 $jalur_terpendek = "";
374 for($x = 0; $x < count($simpulFix_finish_reverse); $x++)
375 {
376     if($x == (count($simpulFix_finish_reverse)-1))
377     {
378         $jalur_terpendek .= $simpulFix_finish_reverse[$x];
379     }else{
380         $jalur_terpendek .= $simpulFix_finish_reverse[$x] . "->";
381     }
382 }
383
384 $json['status'] = 'success';
385 $json['success'] = 'generate_jalur_terpendek';
386 $json['teks'] = 'Jalur berhasil dibuat';
387 $json['content'] = $jalur_terpendek;
388
389 return json_encode($json);
390
391 } // end function jalurTerpendek
392 }
393 ?>

```