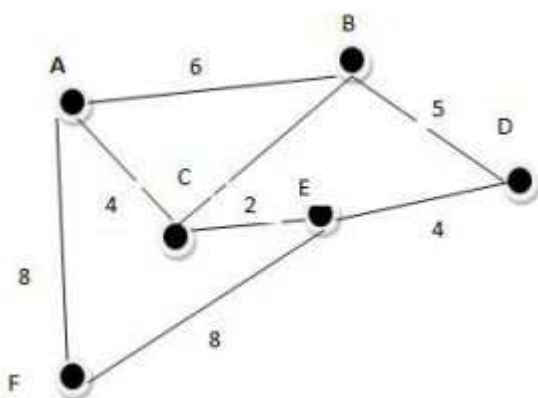


RI32's WEBLOG

Aplikasi mencari jalur terpendek dengan algoritma Dijkstra menggunakan PHP

Posted on 22 May 2011 by Agus Sumarna

About these ads (<http://en.wordpress.com/about-these-ads/>)



(<http://ri32.files.wordpress.com/2011/05/gambar.jpg>)

Jalur Terpendek Dijkstra

Latar Belakang

kakak.. minta bantuannya lg skali ya..

sya mau bwt program yang kalau dimasukan input A-D sesuai dengan foto diatas, output dari programnya adalah **A-C-E-D**

-program akan mencari jalan terpendek untuk sampai ke tujuan. dalam kasus ini A adlah smber, dan D adalah tujuannya..

bisa minta tolong gmn notasi algoritmik nya kakak...

Algoritma Dijkstra

(dinamai menurut penemunya, seorang ilmuwan komputer, Edsger Dijkstra), adalah sebuah algoritma rakus (*greedy algorithm*) yang dipakai dalam memecahkan permasalahan jarak terpendek (*shortest path problem*) untuk sebuah graf berarah (*directed graph*) dengan bobot-bobot sisi (*edge weights*) yang bernilai tak-negatif.

Misalnya, bila *vertices* dari sebuah graf melambangkan kota-kota dan bobot sisi (*edge weights*) melambangkan jarak antara kota-kota tersebut, maka algoritma Dijkstra dapat digunakan untuk menemukan jarak terpendek antara dua kota.

Input algoritma ini adalah sebuah graf berarah yang berbobot (*weighted directed graph*) G dan sebuah sumber *vertex* s dalam G dan V adalah himpunan semua *vertices* dalam graph G .

Setiap sisi dari graf ini adalah pasangan *vertices* (u,v) yang melambangkan hubungan dari *vertex* u ke *vertex* v . Himpunan semua tepi disebut E .

Bobot (*weights*) dari semua sisi dihitung dengan fungsi

$$w: E \rightarrow [0, \infty)$$

jadi $w(u,v)$ adalah jarak tak-negatif dari *vertex* u ke *vertex* v .

Ongkos (*cost*) dari sebuah sisi dapat dianggap sebagai jarak antara dua *vertex*, yaitu jumlah jarak semua sisi dalam jalur tersebut. Untuk sepasang *vertex* s dan t dalam V , algoritma ini menghitung jarak terpendek dari s ke t .

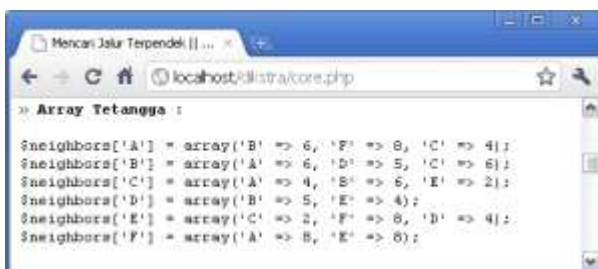
Pseudocode

```

1  function Dijkstra(G, w, s)
2    for each vertex v in V[G] // Initializations
3      d[v] := infinity
4      previous[v] := undefined
5  d[s] := 0 // Distance from s to s
6  S := empty set
7  Q := V[G] // Set of all vertices
8  while Q is not an empty set // The algorithm itself
9    u := Extract_Min(Q)
10   S := S union {u}
11   for each edge (u,v) outgoing from u
12     if d[u] + w(u,v) < d[v] // Relax (u,v)
13       d[v] := d[u] + w(u,v)
14       previous[v] := u

```

Sebuah Kasus



(<http://ri32.files.wordpress.com/2011/05/1.png>)

Data diambil dari gambar diatas

Output

```

» Jalan terpendek dari titik A Menuju ke Titik :

Array
(
    [B] => Array
        (
            [0] => 6
        )
    [F] => Array
        (
            [0] => 8
        )
    [C] => Array
        (
            [0] => 4
        )
    [E] => Array
        (
            [0] => 6
            [1] => C
        )
    [D] => Array
        (
            [0] => 10
            [1] => C
            [2] => E
        )
)

» Keterangan data : Array Tujuan misalnya [D], Total Jarak [0], dan Jalan yang
dilalui [1][2]...[n]

```

(<http://ri32.files.wordpress.com>

/2011/05/21.png)

Tampilan Program setelah diproses

Keterangan

Program akan mencari jalan terpendek dari titik awal ke semua titik "tetangga" nya. dalam hal ini didefinisikan titik awalnya adalah "A" seperti terlihat dalam coding dibawah ini :

```

//untuk titik awal pencarian rute
$point='A';
//untuk memanggil fungsi
$paths = dijkstra($neighbors, $point);
echo '<pre>';
echo "\n\n<b>&raquo; Jalan terpendek dari titik ".$point." Menuju ke Titik :</b>\n\n";
print_r($paths);
echo '</pre>';

```

Maka akan dicari semua jalan dari titik A ke semua titik tetangganya (B,C,D,E,F). jika pertanyaanya adalah mencari jalur terpendek dari titik A ke titik D. maka didapat hasil jalur terpendeknya adalah **A-C-E-D** dengan total jaraknya adalah 10. seperti terlihat pada gambar output yang bertanda garis merah. karena dilihat dari gambar graph diatas jika kita memilih jalur **A-B-D** total jaraknya adalah 11. *problem is resolved*

Potongan Script Fungsi dijkstra

```

function dijkstra($neighbors, $start) {
    $closest = $start;
    while (isset($closest)) {
        $marked[$closest] = 1;
        /* loop through each neighbor for this $closest node and
        * and store the distance and route in an array */
        foreach ($neighbors[$closest] as $vertex => $distance) {
            /* if we already have the route and distance, skip */
            if (isset($marked[$vertex]))
                continue;

            /* distance from $closest to $vertex */
            $dist = (isset($paths[$closest][0]) ? $paths[$closest][0] : 0) + $distance;
            /* if we dont have a path to $vertex yet, create it.
            * If this distance is shorter, override the existing one */
            if (!isset($paths[$vertex]) || ($dist < $paths[$vertex][0])) {

                if (isset($paths[$closest])) {
                    $paths[$vertex] = $paths[$closest];
                }
                $paths[$vertex][0] = $dist;
            }

        }
        unset($closest);
        /* Find the next node we should create a path for */
        foreach ($paths as $vertex => $path) {
            if (isset($marked[$vertex]))
                continue;
            $distance = $path[0];
            if ((isset($min) || $distance < $min) || !isset($closest)) {
                $min = $distance;
                $closest = $vertex;
            }
        }
    }
    return $paths;
}

```

Download

http://www.4shared.com/file/a_ZVqAkB/Dijkstra.html (http://www.4shared.com/file/a_ZVqAkB/Dijkstra.html)

Referensi

- <http://www.phpbuilder.com/snippet/download.php?type=snippet&id=2076> (<http://www.phpbuilder.com>)

[/snippet/download.php?type=snippet&id=2076\)](#)

- http://en.giswiki.net/wiki/Dijkstra%27s_algorithm (http://en.giswiki.net/wiki/Dijkstra%27s_algorithm)
- <http://www.artfulsoftware.com/infotree/queries.php#766> (<http://www.artfulsoftware.com/infotree/queries.php#766>)
- http://id.wikipedia.org/wiki/Algoritma_Dijkstra (http://id.wikipedia.org/wiki/Algoritma_Dijkstra)
- <http://www.informatika.org/~rinaldi/Stmik/Makalah/MakalahStmik16.pdf> (<http://www.informatika.org/~rinaldi/Stmik/Makalah/MakalahStmik16.pdf>)

Sekian dari saya semoga bermanfaat untuk teman-teman dan anda semua...aamiin

Depok, 22 Mei 2011

noname

Filed under: [Belajar PHP](#) Tagged: | [algoritma](#), [aplikasi](#), [dengan](#), [Dijkstra](#), [jalur](#), [mencari](#), [menggunakan](#), [php](#), [terpendek](#)

« [Template Admin Blue Sendiri Menyepi-Edcoustic](#) »

8 Responses



andi, on [22 May 2011 at 5:53 PM](#) said:
sampe sekarang saya gak ngerti algoritma dijkstra 😊
mudah2an abis baca artikel ini bs ngrti

[Reply](#)



survey dan pemetaan, on [7 December 2011 at 12:18 PM](#) said:
saya tertarik dengan program dijkstra ini, bagaimana cara mengaplikasikan dijkstra dalam program2 pemetaan seperti arcview. mohon penjelasannya.
trimakasih

[Reply](#)



Rindi, on [25 January 2012 at 8:24 PM](#) said:
nuhun kang.. sangat amat membantu!

[Reply](#)



Kang Agus, on [26 January 2012 at 6:42 PM](#) said:
sawangsulnya. alhamdulillah 😊



arman nggl, on [30 April 2012 at 1:06 PM](#) said:

mohon pencerahan nya masbro,
saya lgi nyusun skripsi,
tentang pencarian rute terpendek dengan menggunakan algoritma dijkstra,
penerapan dijkstra ke pemrograman nya saya kurang paham, dan graph yg digunakan dlm dijkstra ini
jga saya belum mengerti?

mohon pencerahan nya yah
tq b4

[Reply](#)



ub41d, on [20 October 2012 at 2:51 PM](#) said:
mas kalo ini kan udah di tentuin ya titik akhir nya
nah kalo kita ingin nentuin sendiri titik akhirnya ,gmn tuh mas kode mana yg harus di ubah

misal kita ingin dari F ke B
gmn tuh ...???

[Reply](#)



roku666, on [20 December 2012 at 7:47 PM](#) said:
kalo dihubungin ma database gmn?
misalnya ada tabel arah dengan field awal, tujuan dan bobot

[Reply](#)

Novran Aditya Rachman, on [19 December 2013 at 3:14 AM](#) said:
kalau pakai algoritma A* star gimana coding php nya mas
mohon pencerahan mas untuk menyusun tugas akhir perkuliahan mas

[Reply](#)

[Blog at WordPress.com.](#) [The Digg 3 Column Theme.](#)

Follow

Follow “RI32's WEBLOG”

Powered by [WordPress.com](#)