

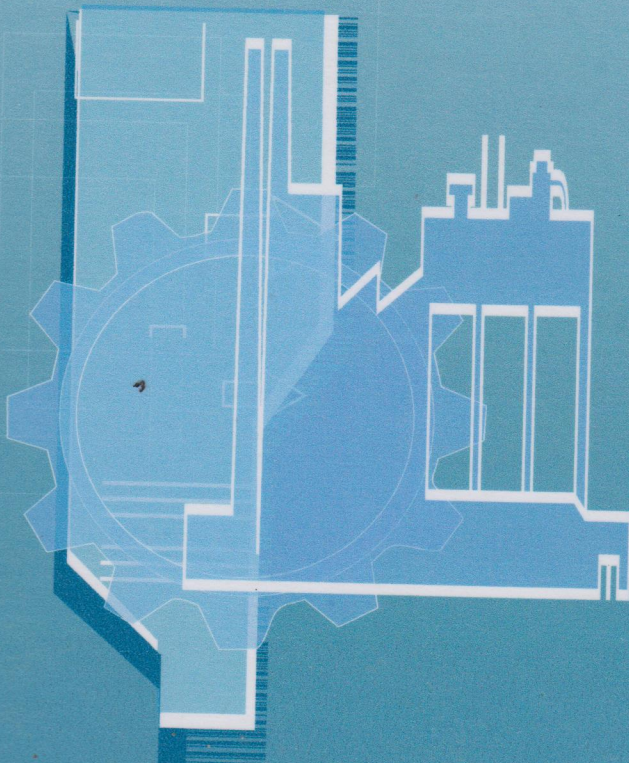
ISSN: 1858-067X



# ***PROSIDING***

## **INDUSTRIAL Gathering 2005**

Tantangan dan Peluang Membangkitkan Daya Saing  
Industri Indonesia Berbasis Teknologi



**Yogyakarta, 11-12 Juni 2005**

Program Studi Teknik Industri  
Universitas Ahmad Dahlan

Jl. Prof.Dr. Soepomo Janturan Yogyakarta  
E-mail: [Industrialgath\\_uad@yahoo.com](mailto:Industrialgath_uad@yahoo.com)



ISSN: 1858-067X



# ***PROSIDING***

## **INDUSTRIAL Gathering 2005**

Tantangan dan Peluang Membangkitkan Daya Saing  
Industri Indonesia Berbasis Teknologi

Reviewer:

Dr. Ir. Dwi Sulisworo, M.T.  
Annie Purwanie, STP., M.T.  
Siti Mahsanah Budijati, STP., MT.  
Endah Utami, S.T., M.T.  
Ir. Tri Budiyanto, M.T.

Editor:

M. Budi Nur Rahman, S.T.  
Harma Didy Wicaksono  
Tri Widyastuti

**Yogyakarta, 11-12 Juni 2005**

Program Studi Teknik Industri  
Universitas Ahmad Dahlan  
Jl. Prof. Dr. Soepomo Janturan Yogyakarta  
E-mail: [Industrialgath\\_uad@yahoo.com](mailto:Industrialgath_uad@yahoo.com)

18	Analisis Waktu Dan Gerakan Kerja Bagian Peracikan Obat Guna Meningkatkan Pelayanan Konsumen (Studi Kasus Di Apotik Rst Dr. Soedjono Magelang)	143
	Retno Rusdijjati, Oesman Raliby	
19	Solusi Optimal Masalah-Masalah Linier Programming Dengan Pendekatan <i>Spread Sheet</i>	150
	Oesman Raloeby	
20	Fermentasi Alkohol Dari Bahan Dasar Bonggol Pisang Ambon ( <i>Musa Paradisiaca L.Cv.Ambon</i> )	157
	Dwi Suhartanti	
21	Implementasi Lintasan Terpendek Dengan Delphi	163
	Suprihatin	
22	Analisis Kandungan Etanol, Ph, Jumlah Sel Khamir Pada Proses Fermentasi Buah Mengkudu ( <i>Morinda Citrifolia</i> ) Selama Penyimpanan	173
	Dwi Suhartanti	
23	Kajian Penerapan Matematika Di Bidang Teknik Khusus Pada Volume Benda Putar	178
	Sumargiyani	
24	Aplikasi <i>Business Process Reengineering</i> Menggunakan Model Simulasi Berbasis Komputer	182
	Ali Parkhan, Adhitya Eka Yudha	
25	Evaluasi Program Keselamatan Kerja Dengan Pendekatan <i>Fault Tree Analysis</i> Pada PT. Sundamanik Bogor	190
	Tri Budiyanto, Afan Kurniawan, Deny Wahyudi	
26	Penerapan Jaminan Mutu Pendidikan Dan Kompetensi	201
	Bambang Purwanggono, Arfan Bakhtiar	
27	Beberapa Aspek Pada Teori Keputusan Bagi Penerapan Ergonomi Industri	209
	Widodo Hariyono	
28	Usulan Rancangan Pengaturan Ulang Gudang Bahan Baku Benang Departemen Weaving (Studi Kasus di PT. Dupantex Pekalongan)	216
	Hari Purnomo, Adhi Bowo Yudanto	
29	Pengendalian Kualitas dengan Pendekatan Six Sigma (Studi Kasus : Kerajinan Patung Priminif, Maharani Handycraft)	227
	Reni Dwi Astuti, Endah Utami, Sri Ulina	

# IMPLEMENTASI LINTASAN TERPENDEK DENGAN DELPHI

Suprihatin

Program Studi Ilmu Komputer FMIPA UAD

## Abstrak

*Komputasi mempunyai peranan penting dalam beberapa bidang, termasuk bidang transportasi. Seseorang yang akan bepergian ke suatu kota pasti memikirkan biaya, dan akan memikirkan pula jaraknya, sehingga tiba di tempat tujuan dengan biaya dan waktu yang minimum, dari sekian jalur alternatif pasti akan kesulitan dalam memilih jalur terpendeknya. Penelitian ini bertujuan mengimplementasikan program mencari lintasan terpendek dengan bantuan program komputer yaitu Delphi. Metode yang dipilih adalah metode yang ditemukan oleh E.W.Dijkstra.*

**Kata kunci:** Lintasan Terpendek, E. W. Dijkstra, Delphi

## A. PENDAHULUAN

Komputer sebagai alat bantu menjadi bertambah penting, seiring dengan perkembangan kehidupan manusia. Komputer sebagai alat Bantu dapat dipergunakan untuk mengimplementasikan algoritma-algoritma pemrograman termasuk algoritma mencari lintasan terpendek

Seseorang yang akan bepergian ke suatu kota pasti memikirkan biaya, dan akan memikirkan pula jaraknya, sehingga tiba di tempat tujuan dengan biaya dan waktu yang minimum, dari sekian jalur alternatif pasti akan kesulitan dalam memilih jalur terpendeknya.

Masalah mencari lintasan terpendek banyak dibahas dalam buku-buku matematika diskrit, atau buku-buku tentang graff. Metode dalam penelitian ini adalah metode yang ditemukan oleh E.W.Dijkstra. Lintasan terpendek telah banyak diimplementasikan salah satunya adalah routing dalam internet.

Penelitian ini akan dibuat program yang dapat menggambar peta dengan jarak-jaraknya (digitasi peta) dan mencari lintasan terpendek dari kota satu ke kota lainnya. Bahasa pemrograman yang dipakai adalah Delphi.

## B. PERMASALAHAN

Masalah yang dibahas dalam tulisan ini adalah:

1. Bagaimana algoritma Dijkstra yang berguna dalam mencari lintasan terpendek dalam suatu graff tak berarah.
2. Bagaimana implementasi algoritma Dijkstra dalam bahasa pemrograman Delphi.

### C KAJIAN PUSTAKA

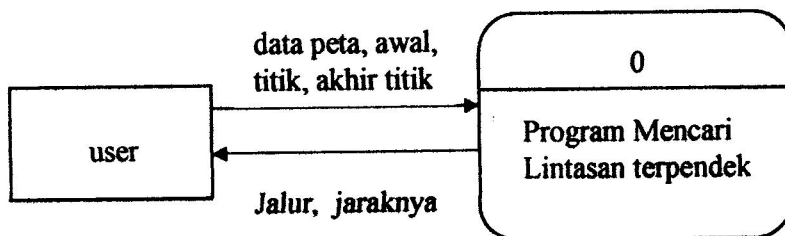
Lintasan terpendek adalah bagian teori graff yang banyak dikembangkan untuk disiplin ilmu antara lain: ekonomi, teknik komputer, otomasi, biologi, dan masih banyak lagi yang lainnya. Graff didefinisikan sebagai  $G=(V,A)$ ,  $V$  adalah himpunan berhingga bukan kosong titik-titik (Vertek) dan  $A$  adalah Himpunan berhingga garis-garis (Ark) (Liu, 1995).

Algoritma Dijkstra adalah algoritma untuk mencari lintasan terpendek. Algoritma mencari lintasan terpendek dari vertek  $a$  ke titik yang lain adalah (Liu, 1995):

1. Mula-mula, misalkan  $P = \{a\}$  dan  $T = V - \{a\}$ . Untuk setiap vertek  $t$  dalam  $T$  misalkan  $l(t) = w(a,t)$
2. Ambil vertek dalam  $T$  yang berindeks terkecil relatif terhadap  $P$ . Lambangkan vertek ini dengan  $x$ .
3. Jika  $x$  ternyata vertek yang ingin dicapai dari  $a$ , maka berhentilah. Jika tidak misalkan  $P' = P \cup \{x\}$  dan  $T' = T - \{x\}$ . Untuk setiap vertek  $t$  dalam  $T'$ , hitunglah indeksnya relatif terhadap  $P'$  dengan rumus  $l'(t) = \min[ l(t), l(x) + w(x,t)]$ .
4. Ulangi langkah 2 dan 3 dengan menggunakan  $P'$  sebagai  $P$  dan  $T'$  sebagai  $T$ .

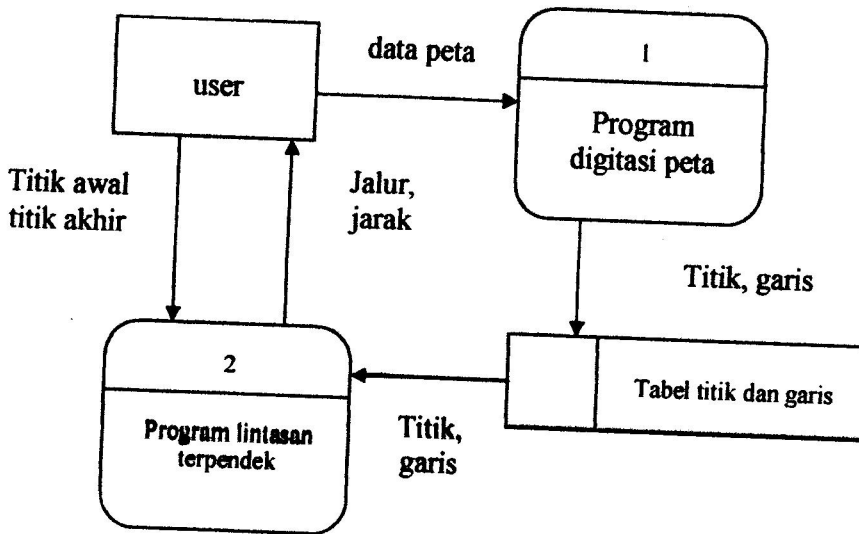
### D. METODE PENELITIAN

Penelitian ini diawali dengan membuat gambar graff dari peta atau digitasi peta. Selanjutnya data tentang peta (posisi titik, nama titik, dan garis-garisnya) digunakan untuk dibaca dan digunakan untuk mencari lintasan terpendek dari titik tertentu ke titik yang lain pada program kedua. Gambar 1 berikut adalah gambar *data flow diagram* program level 0:



Gambar 1. Data flow diagram level 0

Sedangkan data flow diagram pada level 1 terlihat pada gambar2 berikut:

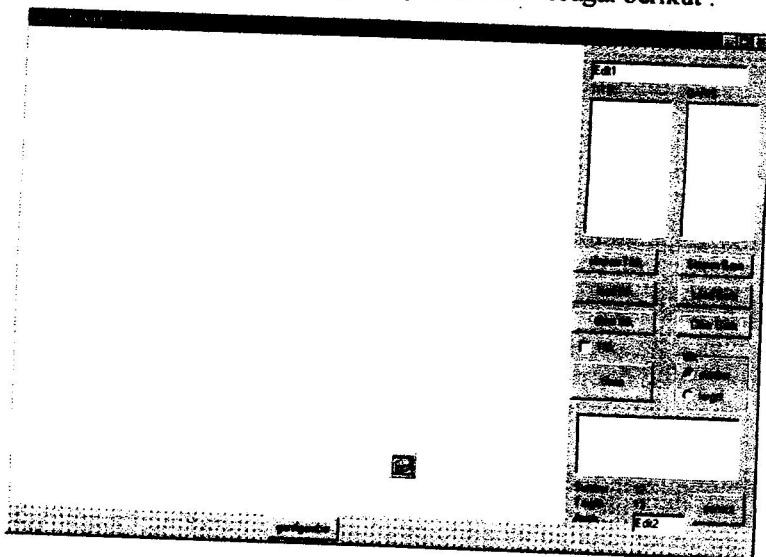


Gambar 2. Data flow diagram level 1

Pertama-tama user membuka peta yang akan digitasi, selanjutnya menandai titik titiknya, dilanjutkan menandai garis-garis dan bobotnya. Setelah selesai kemudian menyimpannya ke *file* penyimpanan. Program lintasan terpendek digunakan untuk mencari jalur terpendek dari satu titik ke titik yang lain dengan terlebih dahulu membuka *file* data titik dan garis yang sudah ada dan petanya.

### E. IMPLEMENTASI PROGRAM

Program terdiri dari 2 yaitu program digitasi dan program mencari lintasan terpendek adapun rancangan *form* untuk program *digitasi* peta adalah sebagai berikut :



Gambar 3. Rancangan form digitasi

---

Terdiri dari daerah kosong untuk menampilkan peta, disertai dengan tombol untuk memanggil peta. Tombol-tombol yang lain adalah: simpan titik (untuk menyimpan titik), *load* titik (untuk memanggil data titik), menghapus titik (menghapus data titik dari memori). Begitu juga untuk tombol garis. *Cek bok* untuk operasi penandaan titik-titik jika tidak dipilih berarti operasi penandaan garis dengan pilihan sumber dan target titiknya. *Edit* pertama untuk nama titiknya sedangkan *edit* kedua untuk bobotnya. *Memo* ada 3 berguna untuk menampilkan titik-titiknya, garis-garisnya, dan nama titik.

Prosedur yang penting dalam program ini yaitu menandai titik dan garisnya dengan mengklik gambar petanya, yaitu prosedur *Image mouse down* sebagai berikut:

```

1. procedure TForm1.Image1MouseDown(Sender: TObject; Button: TMouseButton;
2. Shift: TShiftState; X, Y: Integer);
3. var kal : string;
4. jr2 : real ;
5. i,x1,y1 : integer;
6. bpTtk : Ptitik ;
7. begin
8. if CheckBox1.Checked then
9. begin
10. image1.Canvas.Pen.Color := Clblue;
11. image1.Canvas.Ellipse(x-5,y-5,x+5,y+5);
12. inc(counter);
13. kal := inttostr(x) + ',' + inttostr(y)+',';
14. Memol.Lines.Append(kal);
15. Memol.Lines.Append(edit1.text);
16. end else
17. begin
18. bpTtk := Ltitik ;
19. i := banyaknyatitik+1 ;
20. repeat
21. dec(i);
22. x1 := bpTtk^.isi.X;
23. y1 := bpTtk^.isi.Y;
24. jr2 := sqrt(x1-x) + sqrt(y1-y);
25. bpTtk := bpTtk^.next ;
26. until (jr2 <= er2) or (bpTtk = nil );
27. if RadioGroup1.ItemInaex = 0 then
28. begin
29. xsumber := x1;
30. ysumber := y1;
31. sumber := i;
32. label5.Caption := inttostr(i);
33. end else
34. begin
35. xtarget := x1;
36. ytarget := y1;
37. target := i;
38. label6.Caption := inttostr(i);
39. end;
40. end;
41. end;

```

Baris 8-16 merupakan perintah jika yang digunakan penandaan titik, baris 17-40 jika yang digunakan penandaan garis. Struktur data garis berupa *link list* dan begitu juga garisnya. Adapun struktur datanya adalah sebagai berikut:

*type*

*relasi = record*

*s,t,*

*bobot : integer ;*

*end;*

*prelasi = ^pointerrelasi;*

*pointerrelasi = record*

*isi : relasi ;*

*next : prelasi ;*

*end;*



```
ptitik = ^pointertitik ;  
pointertitik = record  
isi : Tpoint ;  
next : ptitik ;  
end;
```

```
SiTitik = record  
a : integer ;  
jalur : string ;  
jarak : integer ;  
end;
```

```
PSiTitik = ^pointerSiTitik ;  
PointerSiTitik = record  
isi : SiTitik ;  
next : PSiTitik ;  
end;
```

```
Phimpunan = ^pointerHimpunan ;  
pointerhimpunan = record  
isi : integer ;  
next : pHimpunan ;  
end;
```

Penggunaan struktur *link list* karena sifatnya yang dinamis, yaitu jumlah memori tergantung dari jumlah datanya, dapat dihapus ataupun ditambahkan.

Program kedua yaitu program mencari lintasan terpendek adapun rancangan *form*-nya adalah sebagai berikut:

Gambar 4 Rancangan *Form* Program Mencari Lintasan Terpendek

Untuk memudahkan *file* peta, *file* data namanya tertentu sehingga tidak perlu memanggil *file* dengan memasukan namanya. *Form* terdiri dari 3 tombol cari lintasan, *clear* dan *exit* serta pilihan titik awal dan titik akhir. Hasilnya berupa jalur dan jaraknya. Adapun prosedur yang penting adalah:

1. *Carigriff* adalah prosedur utama yaitu mencari lintasan terpendek
2. *initPQ* adalah inisialisasi himpunan P dan Q (T)
3. *cariQ* adalah prosedur mencari himpunan Q
4. *cariMinQ* adalah prosedur mencari harga minimum dari Q
5. *hapusQ* adalah prosedur menghapus x dalam Q dan menambahkan ke P

Adapun prosedur-prosedurnya adalah sebagai berikut:

```

1. procedure Tform1.carigravf;
2. var k : PStTitik ;
3. bpSt : PStTitik ;
4. takhir : integer ;
5. begin
6.   initPQ;
7.   repeat
8.     cariQ;
9.     cariMinQ(k);
10.    new(bpSt);
11.    bpSt^.isi := k^.isi ;
12.    bpSt^.next := P ;
13.    P := bpSt ;
14.    takhir := k^.isi.a ;
15.    hapusQ(k^.isi.a);
16.  until ( Q = nil ) or ( takhir = Finish);
17. end;

```

### Prosedur 2 Mencari lintasan terpendek

```

1. procedure Tform1.InitPQ;
2. var j : integer ;
3. kar : string ;
4. bpTtk : Ptitik ;
5. bpQ : PStTitik;
6. begin
7.   Q := nil; j := 1 ;
8.   bpTtk := Ltitik ;
9.   repeat
10.    new(bpQ);
11.    bpQ^.isi.a := j ; bpQ^.isi.jarak := maxbobot ;
12.    bpQ^.Next := Q ; Q := bpQ ;
13.    inc(j); bpTtk := bpTtk^.next ;
14.  until bpTtk = nil ;
15.  new(P);
16.  with P^.isi do
17.  begin
18.    a := q0; str(q0, kar); jalur := kar + ',';
19.    jarak := 0 ;
20.  end;
21.  p^.next := nil ;
22.  hapusQ(q0); ;
23. end;

```

### Prosedur 3 Inisialisasi himpunan P dan Q

```

1. procedure TForm1.cariQ;
2. var bpQ : PStTitik ;

3. begin
4.   bpQ := Q ;
5.   repeat
6.     carititik(bpQ);
7.     bpQ := bpQ^.next ;
8.   until bpQ = nil ;
9. end;

```

### Prosedur 4 Mencari himpunan Q

```

1. procedure Tform1.cariMinQ(var jadi: PSttitik);
2. var minim, j, si : integer ;
3. jd, bpQ : PstTitik ;
4. begin
5. bpQ := Q ;
6. minim := maxbobot ;
7. repeat
8. if bpQ^.isi.jarak < minim then
9. begin
10. minim := bpQ^.isi.jarak ;
11. jd := bpQ ;
12. end;
13. bpQ := bpQ^.next ;
14. until bpQ = nil ;
15. jadi := jd;
16. end;

```

### Prosedur 5 Mencari Minimum himpunan Q

```

procedure Tform1.hapusQ(q0 : integer);
var pbQ1, pbQ2 : PSttitik ;
begin
pbQ1 := Q ;
if pbQ1^.isi.a = q0 then begin Q := Q^.next ; dispose(pbQ1); end
Else
begin
pbQ1 := Q ;
pbQ2 := pbQ1^.next ;
While pbQ2^.isi.a <> q0 do
begin
pbQ1 := pbQ1^.Next ;
pbQ2 := pbQ2^.Next ;
end;
pbQ1^.next := pbQ2^.next ;
dispose(pbQ2);
end;
end;

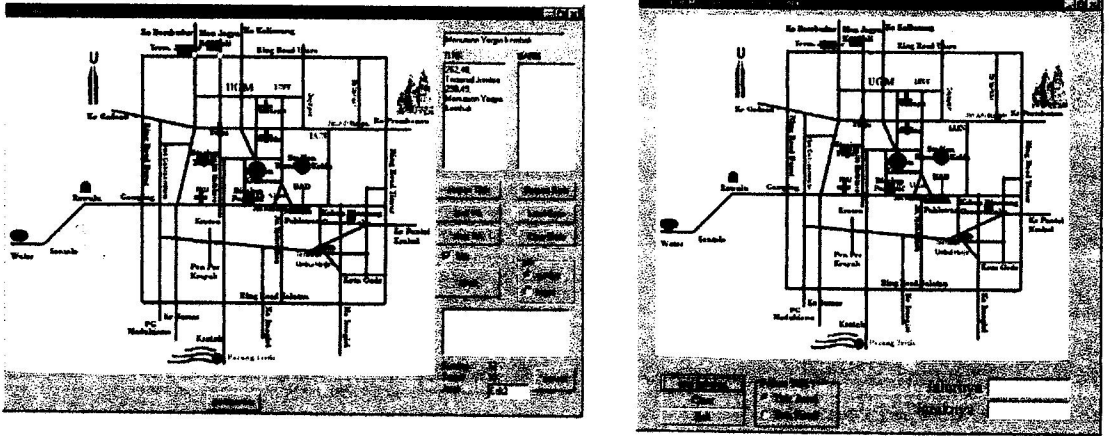
```

### Prosedur 6 menghapus x dalam Q dan menambahkan ke P



## F. HASIL PROGRAM

Berikut ini adalah dua contoh tampilan program, yang pertama digitasi peta dan yang kedua adalah program mencari lintasan terpendek:



Gambar 5. Hasil running program

## G. KESIMPULAN DAN SARAN

Telah dibuat program mencari lintasan terpendek, yang dapat mencari jalur terpendek dari titik satu ke titik yang lain. Gambar peta sebagai masukan data titik dan garis, dengan begitu program dapat disesuaikan dengan peta yang dimasukkan. Program ini masih dapat dikembangkan agar peta yang tampil dapat besar, tidak terbatas seperti program di atas. Nama data harus tertentu ini hanya untuk memudahkan proses, untuk selanjutnya dapat dikembangkan nama data sesuai yang dikehendaki pemakai

## DAFTAR PUSTAKA

- Bucknall Julian, 2001, *The Tomes of Delphi Algorithms and Data Structures*, Wordware, Texas
- Dwi Sanjaya, 2001, *Berpeualangan Struktur Data di Planet Pascal*, J&J Learning, Yogyakarta
- Liu C.L, 1995, *Dasar-dasar Matematika Diskret*, Gramedia, Jakarta
- Susilo Djoko, 2005, *Grafika Komputer dengan Delphi*, Graha Ilmu, Yogyakarta