

# Penerapan Graf Berupa Senarai Berkait serta Algoritma Dijkstra dalam Pemrosesan Data Struktur Bangunan

Fatardhi Rizky Andhika -- NIM -- 13508092

Program Studi Teknik Informatika  
Institut Teknologi Bandung  
Jalan Ganesha no 10  
e-mail: if18092@students.if.itb.ac.id

## ABSTRAK

Sebuah bangunan terdiri atas komponen-komponen yang saling sambung dan saling terkait. Tiap-tiap komponen itu bersambung dengan aturan-aturan tertentu, semisal atap bersisian dengan dinding dan sebagainya. Komponen-komponen bangunan ini dapat diukur secara fisik baik bentuk maupun ukurannya. Dalam perancangan sebuah bangunan, seorang ahli bangunan atau arsitek membutuhkan banyak sekali data tentang bangunan yang mereka rancang dan membutuhkan sebuah metode penyimpanan data yang tepat. Data yang ingin disimpan tersebut dapat berupa panjang, lebar, tinggi, maupun ketebalan dari komponen bangunan. Penyimpanan data yang tepat sangat dibutuhkan agar tercipta kemangkusan dan kesanggupan dari tiap proses perancangannya. Masalah tersebut dapat ditanggulangi dengan ditemukannya metode penyimpanan datastruktur bangunan dengan menggunakan metode graf.

Makalah ini bertujuan untuk merancang bagaimana sebuah aplikasi dapat menggunakan graf dalam penyimpanan data struktur bangunan. Graf yang dihasilkan adalah berupa list berkait yang tersusun bermodelkan graf. Tujuan digunakannya graf di sini adalah agar informasi-informasi yang berkaitan dengan bangunan tersebut, akan selalu terjaga keberadaannya dan dapat digunakan untuk operasi-operasi yang lain.

**Kata kunci:** Bangunan, graf, list senarai., Dijkstra

## 1. PENDAHULUAN

Dalam pengertian sehari-hari, bangunan ialah sesuatu yang didirikan dan digunakan untuk keperluan tertentu. Bangunan biasanya tersusun atas sekat-sekat ruangan yang saling sambung dan membentuk suatu kesatuan. Bangunan bersifat teratur dan memiliki isi di dalamnya, terhindar dari pengaruh dan cuaca dari luar.

Bangunan pada umumnya disamakan fungsinya dengan rumah, gedung, maupun segala macam sarana hasil

buatan dan ciptaan manusia. Sejak masa dahulu sampai sekarang, keberadaan bangunan adalah sangat penting. Pada masa dahulu, keberadaan bangunan semisal jembatan merupakan tonggak perekonomian sebuah kerajaan maupun negara. Seperti juga halnya dengan tembok raksasa, bangunan ini dibangun demi kelancaran jalur ekonomi perdagangan sutra. Pada masa sekarang, hampir semua kegiatan manusia dilakukan dengan sarana bangunan. Perkantoran, pertokoan, maupun pemerintahan dilakukan dengan prasarana bangunan.

Perkembangan ilmu pengetahuan juga tidak terlepas dari perkembangan ilmu pengetahuan seperti halnya arsitektur, teknik sipil yang berkaitan dengan bangunan. Bahkan penggunaan trigonometri dalam matematika juga berkaitan dengan bangunan yang diduga digunakan pada masa lalu.

Perkembangan ilmu pengetahuan yang sekarang yaitu graf. Graf merupakan sebuah bentuk kemajuan yang memberi solusi bagi permasalahan tertentu. Graf dapat diimplementasikan di banyak hal. Tidak hanya pada sebuah bangunan, pengimplementasian juga dapat diterapkan pada hal-hal yang lain misalnya dalam menentukan jalur terpendek, rute angkot terpendek, pembuatan *Finite State Machine*, dan sebagainya. Dalam hal ini, akan dibahas tentang pengimplementasian graf pada sebuah bangunan. Penerapan ini akan mempermudah dalam pembuatan bangunan dan dapat juga membantu para arsitek untuk menghitung luas, lebar, dan jarak antara tiap bangunan antar ruang. Graf digunakan untuk merepresentasikan objek-objek dan hubungan antar objek-objek.

## 2. DASAR TEORI

### 2.1 Dasar Struktur Bangunan

Berdasarkan literatur yang berhubungan dengan bangunan, sebuah bangunan, secara filosofis membutuhkan 3 komponen utama. Tanpa adanya salah satu komponen ini, benda tersebut tidak dapat disebut sebagai bangunan. Adapun ketiga komponen utama tersebut adalah :

### 2.1.1 Pondasi

Pondasi sering disebut struktur bangunan paling bawah yang berfungsi untuk mendukung seluruh beban bangunan dan meneruskan ke tanah di bawahnya. Dikarenakan fungsinya yang digunakan untuk menahan seluruh beban bangunan, maka pondasi harus dibuat kuat, aman, stabil, awet dan mampu mendukung beban bangunan. Dalam membangun suatu pondasi maka ada beberapa hal yang harus diperhatikan, antara lain:

- Berat bangunan yang harus didukung
- Jenis tanah dan daya dukungnya
- Bahan bangunan untuk pondasi yang tersedia, mudah didapat
- Kedalaman tanah yang digunakan untuk pondasi

### 2.1.2 Badan (Rangka)

Rangka Bangunan ialah bagian dari bangunan yang merupakan struktur utama pendukung berat bangunan dan beban luar yang bekerja padanya. Rangka bangunan terdiri atas balok portal yang merangkai kolom menjadi sebuah kesatuan, balok menerima seluruh beban dari plat-lantai dan meneruskan ke kolom-kolom pendukung. Hubungan balok dan kolom ialah jepit-jepit, yaitu system dukungan yang dapat Manahan Momen, Gaya vertical maupun horizontal.

### 2.1.3. Kepala (Atap)

Fungsi atap ialah melindungi bangunan beserta isinya dari pengaruh panas dan hujan. Bentuk dan bahan atap harus serasi dengan rangka bangunannya, agar dapat menambah indah dan anggun serta menambah nilai dari harga bangunannya. Bentuk atap ditinjau dari besarnya kemiringannya, atap dibagi menjadi 2 macam yaitu:

- Atap Landai.  
Atap ini menggunakan penutup atap dengan lembaran besar, seperti seng gelombang atau asbes. Atap ini sangat stabil dikarenakan tekanan angin yang diterima rendah.
- Atap Runcing.  
Atap runcing memberi kesan megah dan anggun bagi pemiliknya. Namun dikarenakan bentuknya yang runcing maka ia akan mendapatkan tekanan angin yang lebih besar.

## 2.2 Dasar Teori Graf

Graf dapat didefinisikan sebagai himpunan tidak kosong antara pasangan simpul-simpul dan sisi-sisi yang menghubungkan sepasang simpul. Himpunan simpul tidak boleh kosong, sedangkan himpunan sisi boleh kosong. Jadi suatu titik juga bisa disebut suatu graf. Graf yang hanya terdiri dari satu buah simpul tanpa sebuah sisi pun disebut graf trivial,

Berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf, graf dapat digolongkan menjadi dua jenis:

### a. Graf Sederhana

Graf sederhana adalah graf yang tidak mengandung gelang maupun sisi ganda. Contoh graf sederhana direpresentasikan dengan jaringan komputer. Pada graf sederhana sisi merupakan pasangan tak terurut. Jadi sisi  $(u, v)$  sama saja dengan  $(v, u)$ .

### b. Graf Tak Sederhana

Graf tak sederhana adalah graf yang mengandung sisi ganda atau gelang. Graf sederhana dibagi menjadi dua macam, yaitu **graf ganda** dan **graf semu**. Graf ganda adalah graf yang mengandung sisi ganda. Sedangkan graf semu adalah graf yang mengandung gelang. Sisi pada graf semu dapat terhubung ke dirinya sendiri.

Berdasarkan orientasi arah pada sisi, maka graf dibedakan menjadi dua jenis:

### a. Graf Tak Berarah

Graf tak berarah adalah graf yang sisinya tidak mempunyai orientasi arah. Urutan pasangan simpul pada graf berarah tidak diperhatikan, jadi sisi  $(u, v)$  sama dengan  $(v, u)$ . Contoh graf tak berarah dalam kehidupan sehari-hari adalah jaringan pada saluran telepon yang dapat beroperasi secara dua arah.

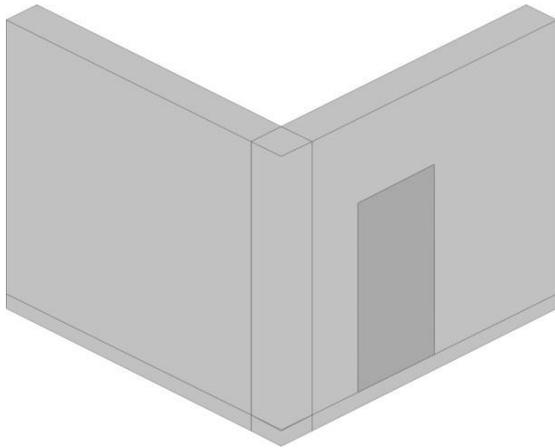
### b. Graf Berarah

Graf berarah adalah graf yang setiap sisinya diberikan orientasi arah. Sisi-sisi yang berarah ini biasa disebut busur. Pada graf berarah, sisi  $(u, v)$  tidak sama dengan  $(v, u)$ . Untuk busur  $(u, v)$ , simpul  $u$  merupakan simpul asal dan simpul  $v$  merupakan simpul terminal. Dalam kehidupan sehari-hari, graf berarah biasa sering dipakai untuk menggambarkan aliran suatu proses.

## 3. PEMBAHASAN

### 3.1 Pemodelan Komponen Struktural Bangunan dalam Graf

Setiap bangunan, memiliki informasi yang berkaitan dengan geometri dan topologi bangunan tersebut yang direpresentasikan dalam model *B-rep* (*boundary representation*). Perbedaan informasi-informasi tersebut akan membedakan jenis relasi yang disimpan dalam model. Dalam translasinya ke dalam model *graf*, sebuah bangunan akan dibagi ke dalam komponen-komponen individu. Setiap komponen tersebut telah dilengkapi dengan informasi geometri dan topologi. Sebagai contoh, diketahui sebuah bangun  $X$ , seperti pada gambar 1.

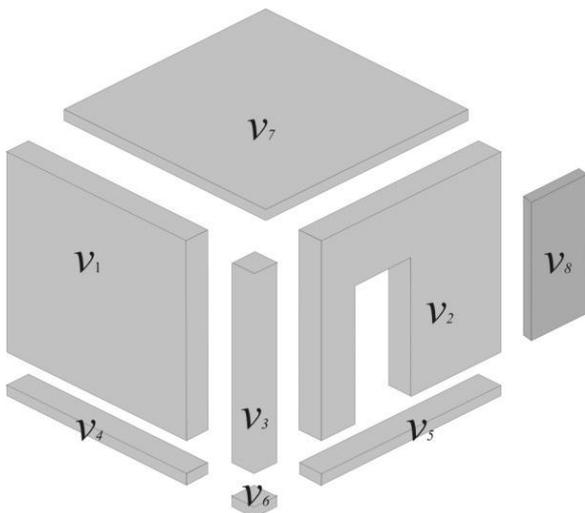


Gambar 1. Rancang Bangun X

Bangun X dapat dibagi menjadi beberapa komponen individu,  $v_1, v_2, v_3, v_4, v_5, v_6, v_7$ , seperti diilustrasikan pada gambar 2. Bangun geometri  $v_1$  dan  $v_2$  adalah representasi dari tembok bangunan.  $v_3$  representasi dari tiang,  $v_4, v_5, v_6$  merepresentasikan pondasi bangunan,  $v_7$  merepresentasikan lantai bangunan dan  $v_8$  merepresentasikan pintu.

Dengan merepresentasikan komponen-komponen bangunan tersebut dengan simpul  $V$ , *vertex*, maka dapat disimpulkan bahwa

$$V = \{ v_1, v_2, v_3, v_4, v_5, v_6 \}$$



Gambar 2. Komponen-komponen Individu Penyusun Bangun X

Berdasarkan relasi antar komponen individu pada bangun X, maka dapat dituliskan bahwa

- $e_1=(v_1, v_3)$ ;
- $e_2=(v_2, v_3)$ ;
- $e_3=(v_2, v_5)$ ;
- $e_4=(v_1, v_4)$ ;
- $e_5=(v_7, v_4)$ ;
- $e_6=(v_7, v_5)$ ;
- $e_7=(v_8, v_5)$ ;
- $e_8=(v_4, v_6)$ ;

$e_9=(v_5, v_6)$ ; dan

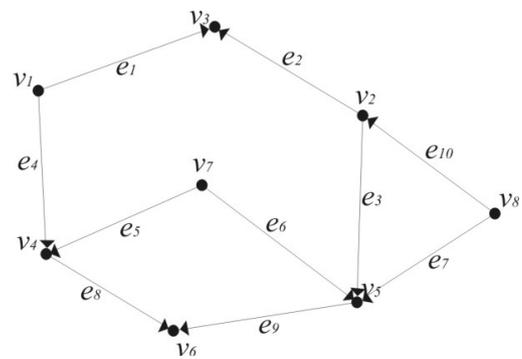
$e_{10}=(v_8, v_2)$ ,

atau dapat ditulis  $e_x = (v_n, v_m)$ , dengan  $e_x$  adalah *edge* yang ke  $x$  yang merelasikan  $v_n$  dengan  $v_m$ . Karena  $e$  adalah *edge* pada graf berarah, maka  $e_x = (v_n, v_m) \neq (v_m, v_n)$ . Dari persamaan diatas, dapat disimpulkan bahwa :

$$E = \{ e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9, e_{10} \}$$

dengan  $E$  adalah *edge*. Dari persamaan-persamaan di atas tersebut, dapat dibuat menjadi sebuah graf  $G$ , dengan :  $G (V, E)$

Permodelan graf  $G$  dapat digambarkan sebagai berikut:



Gambar 3. Representasi Graf G dari Bangun X

### 3.2 Representasi Graf dalam Senarai (List) Berkait

Graf adalah sebuah teknik untuk menjaga keterkaitan sebuah data dengan data yang lain, dengan merepresentasikan data tersebut menjadi sebuah node. Dalam pengkaitan dengan node yang lain, sebuah graf dapat direpresentasikan dalam senarai berkait / *linked list*, atau yang apabila diubah dalam bentuk matriks disebut dengan *adjacency* matriks (matriks yang bernilai 1 atau 0 untuk merepresentasi ada hubungan atau tidak). Dalam representasinya, sebuah graf terdiri dari 2 bagian, yaitu bagian simpul, yang disebut sebagai *list* simpul dan bagian sisi yang disebut dengan *list* sisi.

*List* simpul adalah list yang anggotanya merupakan elemen-elemen simpul / *nodes* penyusun graf. Setiap objek dari sebuah permasalahan akan direpresentasikan menjadi sebuah elemen simpul. Elemen simpul tersebut akan dikaitkan sehingga tersusun menjadi sebuah *list* simpul.

Elemen simpul terbagi menjadi 3 bagian, yaitu informasi, *next*, dan *list* sisi. Informasi merupakan bagian dari elemen simpul yang akan menyimpan setiap informasi yang akan dikaitkan dalam graf tersebut. Informasi merupakan bagian inti dari sebuah graf yang direpresentasikan. *next*, merupakan bagian untuk menyimpan informasi *node* berikutnya, sedangkan *list* sisi adalah bagian untuk menyimpan elemen-elemen yang direlasikan oleh elemen simpul tersebut.

List sisi adalah kumpulan dari elemen yang merupakan representasi dari relasi satu simpul dengan simpul yang lain. Elemen-elemen yang tergabung dalam list sisi disebut sebagai elemen sisi. Setiap elemen pada list sisi, memiliki satu list sisi, yang berarti bahwa elemen tersebut memiliki keterkaitan dengan anggota list sisi yang lain.

### 3.2 Penyimpanan Data Struktur Bangunan

Pada pemodelan sebuah graf, akan digunakan 2 list, yaitu list simpul dan list sisi. List simpul, akan menyimpan informasi tentang vertex, sedangkan list sisi, menyimpan informasi edge. Secara garis besar, elemen simpul dan elemen sisi digambarkan sebagai berikut

Informasi vertex	List sisi
Kaitan ke elemen simpul selanjutnya	

Gambar 4. Elemen Simpul (vertex)

Informasi edge	Kaitan yang menunjukkan elemen berikutnya
----------------	---

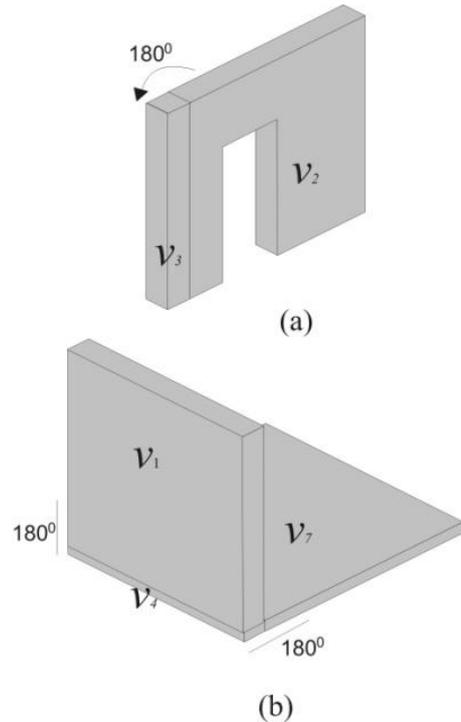
Gambar 5. Elemen Sisi (edge)

Informasi vertex, meliputi informasi geometri dan topologi. Informasi geometri adalah informasi yang berkaitan dengan bentuk komponen individu, sebagai contoh, ukuran. Sedangkan informasi topologi adalah informasi yang mendeskripsikan spesifikasi dalam satu komponen tersebut. Informasi topologi pada satu komponen, akan berbeda dengan komponen bangunan yang lain.

Informasi edge merupakan informasi relasi komponen satu dengan komponen yang lain. Dalam hal ini, komponen diasosiasikan dengan elemen simpul. Informasi edge terdiri dari :

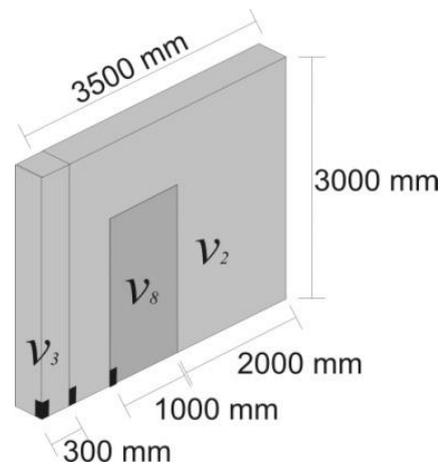
- 1) besar sudut antara antar komponen yang direlasikan, disimbolkan sebagai  $\theta$ ,
- 2) posisi koordinat  $x, y, z$  antar komponen yang direlasikan, dan
- 3) vertex yang dituju.

Besar sudut antar komponen yang direlasikan, dihitung berdasarkan letak satu vertex terhadap vertex yang lainnya. Dalam gambar 6 (a) dijelaskan bahwa, sudut dihitung berdasarkan posisi  $v_2$  terhadap  $v_3$ . Dengan demikian maka,  $\theta(v_2, v_3) = 180^\circ$  pada gambar (a) dan  $\theta(v_1, v_4) = \theta(v_7, v_4) = 180^\circ$  pada gambar (b).



Gambar 6. Sudut Antara 2 Komponen

Posisi koordinat  $x$  dan  $y$ , merupakan koordinat relatif yang menghubungkan antara komponen satu dengan komponen yang lain. Penentuan posisi koordinat didasarkan pada titik pusat komponen geometri atau disebut sebagai titik acuan. Setiap titik acuan memiliki koordinat yaitu koordinat  $(x, y, z) = (0, 0, 0)$ . Gambar 8 mengilustrasikan posisi setiap  $v_n$  dengan  $v_m$ . Apabila posisi  $v_n$  sejajar dengan  $v_m$ , maka posisi koordinat relatif  $v_n$  terhadap  $v_m$  adalah  $(0, 0, 0)$ , karena  $v_n$  mempunyai posisi yang sejajar dengan  $v_m$ .



Gambar 7. Posisi koordinat relatif dari  $v_8$  terhadap  $v_2$  dan  $v_2$  terhadap  $v_3$

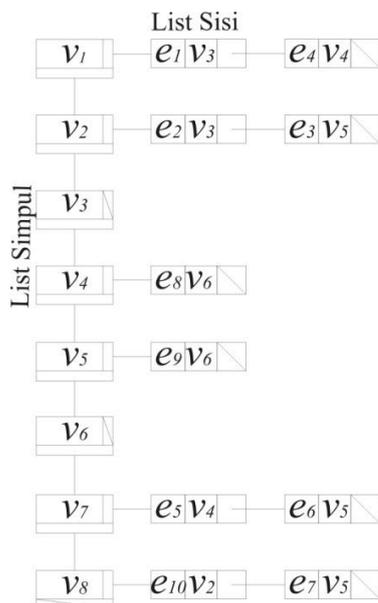
Gambar 7 mengilustrasikan terdapat 3 buah komponen, yaitu  $v_2, v_3$ , dan  $v_8$ , dengan  $e(v_8, v_2)$  dan  $e(v_2, v_3)$ , maka

koordinat relatif antara  $v_3$  terhadap  $v_2$  adalah  $(0, 0, 0)$  dan  $v_8$  terhadap  $v_2$  adalah  $(200, 0, 0)$ . Nilai  $x(v_8) = 200$ , didapat karena  $v_8$  berposisi 200 cm terhadap komponen  $v_2$ . Satuan nilai yang disimpan dalam edge adalah *centimeter (cm)*. Pada gambar 7, titik acuan diberi warna hitam pada sudut bawah setiap komponen. Tabel 1 memuat informasi yang dimiliki *edge* berdasarkan graf  $G$ . **Tabel 1.** Informasi *edge* berdasarkan graf  $G$

**Tabel 1 Informasi *edge* Berdasarkan Graf  $G$**

Edge	$\theta$	Coordinates			$v_n$	$v_m$
		$x$	$y$	$z$		
$e_1$	0	380	0	0	$v_1$	$v_3$
$e_2$	180	30	0	0	$v_2$	$v_3$
$e_3$	180	40	0	0	$v_2$	$v_5$
$e_4$	180	40	0	0	$v_1$	$v_5$
$e_5$	180	40	0	0	$v_7$	$v_4$
$e_6$	180	40	0	0	$v_7$	$v_5$
$e_8$	0	380	0	0	$v_4$	$v_6$
$e_9$	180	30	0	0	$v_5$	$v_6$
$e_{10}$	0	50	0	0	$v_8$	$v_2$

Berdasarkan informasi *vertex* dan *edge* tersebut, graf  $G$  dapat direpresentasikan dalam *linked list*, yaitu sebagai berikut :



**Gambar 8. Representasi *linked list* graf  $G$**

Gambar 8 menunjukkan bahwa setiap elemen pada *list* simpul, memiliki satu *list* sisi.

Dari gambar tersebut, terlihat bahwa sebuah elemen simpul mempunyai penunjuk ke sisi yang berhubungan dengan simpul tersebut serta sebuah penunjuk lain ke simpul berikutnya. Konsep penerapannya sangat mirip dengan *multilist* yang ada pada beberapa bahasa pemrograman.

Dengan adanya penyimpanan data struktur bangunan seperti ini, diyakini akan tercipta kemangkusan dan kesangkilan bagi mereka yang membutuhkan data tersebut semisal arsitek maupun ahli bangunan.

### 3.3 Penggunaan Algoritma Dijkstra

Selain dengan penggunaan graf sebagai media penyimpanan, graf juga dapat digunakan untuk penentuan jalur terpendek dengan penggunaan algoritma Dijkstra. Bentuk algoritma tersebut digunakan untuk mencari *shortest path*. Berikut merupakan pseudo-code dari algoritma Dijkstra.

```

1  function Dijkstra(Graph, source):
2      for each vertex v in Graph:
// Initializations
3          dist[v] := infinity
// Unknown distance function from source to
v
4          previous[v] := undefined
// Previous node in optimal path from
source
5          dist[source] := 0
// Distance from source to source
6          Q := the set of all nodes in Graph
// All nodes in the graph are
unoptimized - thus are in Q
7          while Q is not empty:
// The main loop
8              u := vertex in Q with smallest
dist[]
9              if dist[u] = infinity:
10                 break
// all remaining vertices are inaccessible
from source
11                 remove u from Q
12                 for each neighbor v of u:
// where v has not yet been removed from Q.
13                     alt := dist[u] +
dist_between(u, v)
14                     if alt < dist[v]:
// Relax (u,v,a)
15                         dist[v] := alt
16                         previous[v] := u
17                 return dist[]

```

Pencarian jalur terpendek ini dapat diterapkan pada penyusunan data struktur bangunan. Misalnya saja, seorang ahli bangunan ingin menghitung jarak antara 2 buah dinding yang dalam graf direpresentasikan oleh 2 simpul, maka dengan bantuan algoritma ini, orang tersebut akan dapat menentukan jarak tersebut dengan mudah sehingga dalam perancangan bangunanpun akan menjadi lebih mudah.

### 4. KESIMPULAN

Kesimpulan yang dapat diperoleh dari makalah ini adalah :

- a) Graf merupakan salah satu metode dalam struktur data yang berfungsi untuk menjaga keterikatan satu informasi dengan informasi yang lain.
- b) Dalam permasalahan penyimpanan informasi struktur bangunan, graf dapat digunakan untuk menyimpan informasi yang menyusun elemen bangunan, yaitu informasi geometri dan informasi topologi.
- c) Penggunaan graf bersifat saling menjaga satu elemen dengan elemen yang lain sehingga informasi-informasi yang telah disimpan tersebut dapat di kembalikan lagi ke dalam bentuk permasalahan awal, tanpa mengurangi ketepatan data sebelum disimpan dalam graf.
- d) Dalam realisasinya dalam dunia nyata, konsep graf senarai seperti ini dapat direalisasikan dengan penggunaan list senarai pada bahasa pemrograman C.
- e) Pada bahasa C, sebuah list senarai adalah list yang tiap elemennya memiliki satu nilai elemen serta sebuah penunjuk untuk elemen berikutnya. Konsep elemen yang digunakan pada graf dalam makalah ini dan konsep elemen pada bahasa C adalah sama. Atas kesamaan konsep inilah, realisasinya dapat digunakan dengan bahasa C.
- f) Pada graf penyimpanan data tersebut juga dapat diterapkan algoritma Dijkstra untuk mencari rute jalur terpendek antara 2 simpul yang digunakan.
- g) Penggunaan graf sebagai solusi dalam penyimpanan data struktur bangunan diyakini sebagai salah satu solusi tepat yang dapat dijadikan alternatif permasalahan.

## REFERENSI

- [1] Goodrich, Michael T., Tamssia, Roberto. & Triandopoulos, Nikos. & Cohen, Robert. "Authenticated Data Structures for Graph and Geometric Searching". <http://www.cs.brown.edu/cgc/stms/papers/authDataStr.pdf>. Terakhir diakses tanggal 19 Desember 2009
- [2] Himawan, Bondan, dkk.2008.*Implementasi Graf Dalam Penyimpanan Data Struktur Bangunan*
- [3] van Treeck, Christoph. & Rank, Ernst. "Analysis of Building Structure and Topology Based on Graph Theory"[http://epub.uniweimar.de/volltexte/2004/238/pdf/ccbex\\_204.pdf](http://epub.uniweimar.de/volltexte/2004/238/pdf/ccbex_204.pdf) terakhir diakses tanggal 19 Desember 2009
- [4] Munir, Rinaldi. 2004. Diktat Kuliah Matematika Diskrit. Departemen Teknik Informatika. Institut Teknologi Bandung.
- [5] van Treeck, Christoph. & Rank, Ernst. "Analysis of Building Structure and Topology Based on Graph Theory"[http://epub.uniweimar.de/volltexte/2004/238/pdf/ccbex\\_204.pdf](http://epub.uniweimar.de/volltexte/2004/238/pdf/ccbex_204.pdf) terakhir diakses tanggal 19 Desember 2009