

# TRANSITIF KLOSUR DARI GABUNGAN DUA RELASI EKUIVALENSI PADA SUATU HIMPUNAN DENGAN STRUKTUR DATA DINAMIS

Sukmawati Nur Endah  
Program Studi Ilmu Komputer Jurusan Matematika FMIPA UNDIP  
Jl. Prof. H. Soedarto, S.H, Semarang 50275

**Abstract.** A relation  $R$  on set  $A$  is an equivalence relation on  $A$  if and only if  $R$  is reflexive, symmetric and transitive. If two equivalence relations on set  $A$  are combined, the combination of them is not surely an equivalence relation, because it is not surely transitive relation. In this paper is found the smallest transitive relation (transitive closure) of the combination, to be an equivalence relation. Then the steps to determine transitive closure are programmed in Pascal programming language with dynamic data structure that is multilist.

**Keywords:** transitive closure, equivalence relation, dynamic data structure

## 1. PENDAHULUAN

Salah satu konsep dasar bidang matematika adalah relasi. Relasi pada himpunan  $A$  sering memenuhi sifat-sifat refleksif, simetris, dan transitif. Namun ada juga relasi yang tidak memenuhi salah satu dari ketiga sifat di atas. Hal ini dapat ditambahkan pasangan relasi pada  $R$  sampai mendapat relasi dengan sifat yang diinginkan. Untuk menambahkan sesedikit pasangan relasi baru yang mungkin, maka harus dicari relasi terkecil  $R'$  pada himpunan  $A$  yang mengandung  $R$  dan memiliki sifat yang diinginkan. Relasi  $R'$  seperti di atas disebut klosur (*closure*) dari  $R$  [4]. Dengan demikian suatu relasi dapat dicari refleksif klosur (*reflexive closure*), simetris klosur (*symmetric closure*), dan transitif klosur (*transitive closure*).

Salah satu aplikasi transitif klosur yang menarik adalah pada relasi ekuivalensi (*equivalence relation*). Misalkan  $R$  dan  $S$  adalah relasi ekuivalensi pada himpunan  $A$ . Jika  $R$  dan  $S$  digabungkan, maka  $R \cup S$  belum tentu relasi ekuivalensi. Hal ini disebabkan  $R \cup S$  belum tentu transitif. Untuk membentuk relasi ekuivalensi terkecil yang mengandung  $R$  dan  $S$  maka perlu dicari transitif klosur dari  $R \cup S$  tersebut.

Permasalahan yang diambil dalam penulisan ini adalah menentukan relasi ekuivalensi terkecil yang mengandung dua relasi ekuivalensi pada suatu himpunan. Dengan kata lain, permasalahannya adalah menentukan transitif klosur dari gabungan dua relasi ekuivalensi. Kemudian diimplementasikan ke dalam program komputer dengan bahasa pemrograman Turbo Pascal dan menggunakan struktur data dinamis, yaitu senarai berantai.

Untuk membatasi ruang lingkup permasalahan, maka dalam penulisan ini diberikan batasan sebagai berikut:

1. Dua relasi ekuivalensi sebagai input untuk mencari relasi ekuivalensi terkecil yang mengandung keduanya merupakan relasi pada himpunan yang sama.
2. Penentuan transitif klosur menggunakan algoritma Warshall.
3. Program yang dibuat menggunakan struktur data dinamis.

Tujuan penulisan ini adalah

1. Untuk mencari relasi ekuivalensi terkecil yang memuat dua relasi ekuivalensi pada suatu himpunan.
2. Untuk memahami program dengan struktur data dinamis.

## 2. PEMBAHASAN

### A. Transitif Klosur

**Definisi 2.1.** [2] Misalkan  $R$  relasi pada himpunan  $A$ . *Transitif klosur* dari  $R$  adalah relasi  $R'$  sedemikian hingga:

- (i).  $R'$  adalah transitif
- (ii).  $R \subseteq R'$
- (iii). Untuk suatu relasi transitif  $R''$ , jika  $R \subseteq R''$  maka  $R' \subseteq R''$ .

Transitif klosur dari relasi  $R$  dinotasikan dengan  $t(R)$ .

Jika  $R$  relasi pada himpunan  $A$ , maka transitif klosurnya dapat dibentuk dengan menambahkan pada relasi  $R$  semua pasangan terurut yang diperlukan untuk membentuk relasi transitif yang baru. Tetapi bagian (iii) dari definisi 1 menyatakan bahwa tak ada pasangan yang ditambahkan kecuali kalau diperlukan. Jadi  $R'$  adalah relasi transitif terkecil dan  $R \subseteq R'$ . Jika  $R$  transitif, maka relasi transitif terkecil yang mengandung  $R$  adalah  $R$  itu sendiri.

**Teorema 2.1.** [5] Misalkan  $R$  relasi pada himpunan  $A$ .  $R$  transitif jika dan hanya jika  $t(R) = R$ .

**Bukti.**

- ( $\Rightarrow$ ) Jika  $R$  transitif, maka jelas  $R$  adalah relasi transitif terkecil yang mengandung  $R$ . Sehingga  $R = t(R)$ .
- ( $\Leftarrow$ ) Jika  $t(R) = R$ , maka berdasarkan sifat (i) dari definisi 1,  $R$  adalah transitif. ■

**Teorema 2.2.** [2] Misalkan  $R$  relasi pada himpunan  $A$  maka  $t(R) = \bigcup_{i=1}^{\infty} R^i$ .

**Bukti.**

Bukti ini dibagi dalam dua bagian, yaitu :

- (i).  $\bigcup_{i=1}^{\infty} R^i \subseteq t(R)$

Akan ditunjukkan terlebih dulu bahwa  $R^n \subseteq t(R)$  untuk setiap  $n > 0$ .

- (1). (Basis). Untuk  $n=1$ , berdasarkan definisi 1 bagian (ii) jelas bahwa  $R^1 = R \subseteq t(R)$ .
- (2). (Induksi). Misalkan  $R^k \subseteq t(R)$  benar untuk setiap  $k \geq 1$ , akan ditunjukkan bahwa  $R^{k+1} \subseteq t(R)$ . Ambil sebarang

$$(a,b) \in R^{k+1}. \Leftrightarrow (a,b) \in R^k R \text{ (sebab } R^{k+1} = R^k R).$$

$$\Leftrightarrow \exists c \in A \ni (a,c) \in R^k \wedge (c,b) \in R.$$

$$\Leftrightarrow (a,c) \in t(R) \wedge (c,b) \in t(R) \text{ (dari langkah induksi } R^k \subseteq t(R) \text{ dan langkah basis)}$$

$$\Leftrightarrow (a,b) \in t(R) \text{ (karena } t(R) \text{ transitif).}$$

Karena sebarang  $[(a,b) \in R^{k+1} \Rightarrow (a,b) \in t(R)]$ , maka  $R^{k+1} \subseteq t(R)$ . Sehingga berdasarkan prinsip induksi matematik  $R^n \subseteq t(R)$  untuk setiap  $n \geq 1$ .

Sesudah terbukti  $R^n \subseteq t(R)$  untuk setiap  $n \geq 1$ , dapat disimpulkan bahwa

$$\bigcup_{i=1}^{\infty} R^i \subseteq t(R).$$

$$(ii). t(R) \subseteq \bigcup_{i=1}^{\infty} R^i$$

Akan ditunjukkan terlebih dulu  $\bigcup_{i=1}^{\infty} R^i$  transitif. Misalkan  $(a,b)$  dan  $(b,c)$  sebarang elemen dari  $\bigcup_{i=1}^{\infty} R^i$ . Untuk suatu integer  $s \geq 1$  dan  $t \geq 1$ ,  $(a,b) \in R^s$  dan  $(b,c) \in R^t$ , sehingga  $(a,c) \in R^{s+t}$ .

Jadi  $(a,c) \in \bigcup_{i=1}^{\infty} R^i$  dan oleh karena itu

$\bigcup_{i=1}^{\infty} R^i$  adalah transitif. Karena setiap relasi transitif yang mengandung  $R$  berisi  $t(R)$ , maka  $t(R) \subseteq \bigcup_{i=1}^{\infty} R^i$ .

Dari (i) dan (ii) terbukti  $t(R) = \bigcup_{i=1}^{\infty} R^i$ . ■

**Definisi 2.2.** [2] Misalkan  $R$  relasi pada himpunan  $A$  dan  $n$  bilangan bulat positif, maka *relasi  $R^n$  pada  $A$*  adalah relasi yang menunjukkan path dengan panjang  $n$  dalam  $R$ . Dengan menuliskan  $aR^n b$  berarti terdapat path dengan panjang  $n$  dari  $a$  ke  $b$  dalam  $R$ .

**Definisi 2.3.** [2] Misalkan  $R$  relasi pada himpunan  $A$ . *Relasi  $R^*$  =  $\{(a,b) \in A \times A \mid$  terdapat suatu path dalam  $R$  dari  $a$  ke  $b\}$*  disebut relasi keterhubungan (connectivity)

relation) pada  $A$ . Panjang dari path ini secara umum tergantung pada  $a$  dan  $b$ . Dengan menuliskan  $aR^*b$  berarti terdapat path dalam  $R$  dengan suatu panjang dari  $a$  ke  $b$ .

**Teorema 2.3.** [2] Jika  $R$  relasi pada himpunan  $A$  maka  $R^* = \bigcup_{i=1}^{\infty} R^i$ .

**Bukti.**

Dari definisi 2, untuk setiap  $n > 0$ ,  $R^n = \{(a,b) \in A \times A \mid \text{terdapat path dalam } R \text{ dengan panjang } n \text{ dari } a \text{ ke } b\}$ . Sehingga berdasarkan definisi 3,  $R^* = \bigcup_{i=1}^{\infty} R^i$ .

Dari teorema 2 dan teorema 3, didapatkan transitif klosur  $t(R) = R^*$ . ■

**Teorema 2.4.** [5] Jika  $R$  relasi pada himpunan  $A$  yang mempunyai  $n$  elemen, maka  $t(R) = R^* = \bigcup_{i=1}^n R^i$ .

**Bukti.**

Teorema ini cukup ditunjukkan bahwa  $R^k \subseteq \bigcup_{i=1}^n R^i$  untuk setiap  $k > 0$ . Misalkan  $(x,y) \in R^k$ , maka terdapat path terhubung dengan panjang  $k$  dari  $x$  ke  $y$  dalam digraphnya dan dengan menghapus cycle dari path ini, dapat dibentuk sebuah path terhubung sederhana dari  $x$  ke  $y$ . Karena path sederhana terpanjang yang mungkin dalam digraph dengan  $n$  vertek mempunyai panjang  $n$ , maka  $(x,y) \in R^i$  untuk suatu  $0 < i \leq n$ . Oleh karena itu  $R^k \subseteq \bigcup_{i=1}^n R^i$  untuk  $k > 0$ . Jadi terbukti  $t(R) = \bigcup_{i=1}^n R^i$ . ■

Ada beberapa metode/cara untuk menentukan transitif klosur yaitu:

- a. Metode graphical
- b. Metode matriks
- c. Algoritma Warshall

Dalam penulisan ini, metode yang digunakan dalam menentukan transitif klosur adalah algoritma Warshall.

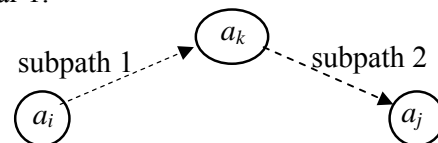
Misalkan  $R$  adalah relasi pada himpunan  $A = \{ a_1, a_2, \dots, a_n \}$ . Jika  $a_1, a_2$

, ...,  $a_n$  adalah path pada  $R$  maka semua vertek selain  $a_1$  dan  $a_n$  disebut vertek-vertek interior (interior vertices) dari path tersebut.

**Definisi 4** [4] Misalkan  $R$  adalah relasi pada himpunan  $A = \{ a_1, a_2, \dots, a_n \}$ . Untuk  $k = 1, 2, \dots, n$ , matriks Boolean  $W_k$  mempunyai nilai 1 pada elemen ke  $(i,j)$  jika dan hanya jika terdapat path dari  $a_i$  ke  $a_j$  dalam  $R$  yang vertek interiornya (jika ada) berasal dari himpunan  $A_k = \{ a_1, a_2, \dots, a_k \}$ .

Dari definisi 4, matriks  $W_n$  mempunyai nilai 1 pada elemen ke  $(i,j)$  jika dan hanya jika terdapat path dalam  $R$  yang menghubungkan  $a_i$  ke  $a_j$ . Dengan kata lain  $W_n = M_{R^*}$ . Jika didefinisikan  $W_0$  sebagai  $M_R$ , maka dipunyai barisan  $W_0, W_1, \dots, W_n$  dengan suku pertama adalah  $M_R$  dan suku terakhir  $= M_{R^*}$  (matriks transitif klosur dari  $R$ ). Masing-masing matriks  $W_k$  ditentukan dari matriks  $W_{k-1}$ . Langkah-langkah dalam mencapai matriks  $R^*$  dari matriks  $R$  inilah yang dinamakan algoritma Warshall. Berikut akan ditunjukkan bagaimana menentukan matriks  $W_k$  dari matriks  $W_{k-1}$ .

Misalkan  $W_k = [t_{ij}]$  dan  $W_{k-1} = [s_{ij}]$ . Jika  $t_{ij} = 1$  maka harus ada path dari  $a_i$  ke  $a_j$  yang vertek interiornya berada dalam himpunan  $A_k = \{ a_1, a_2, \dots, a_k \}$ . Jika vertek  $a_k$  bukan vertek interior dari path ini, maka semua vertek interior harus berasal dari himpunan  $A_{k-1} = \{ a_1, a_2, \dots, a_{k-1} \}$ , sehingga  $s_{ij} = 1$ . Jika  $a_k$  vertek interior dari path, maka situasinya seperti dalam gambar 1.



Gambar 1.  $a_k$  termasuk vertek interior dari path  $a_i$  ke  $a_j$

Diasumsikan bahwa semua vertek interior adalah berbeda. Jadi  $a_k$  hanya terlihat sekali di dalam path ini, sehingga semua vertek interior dari subpath 1 dan subpath 2 harus berasal dari himpunan  $\{ a_1, a_2, \dots, a_{k-1} \}$ . Ini berarti bahwa  $s_{ik} = 1$  dan  $s_{kj} = 1$ .

Sehingga diperoleh  $t_{ij} = 1$  jika dan hanya jika:

$$\left. \begin{array}{l} (i). s_{ij} = 1 \quad \text{atau} \\ (ii). s_{ik} = 1 \quad \text{dan} \quad s_{kj} = 1 \end{array} \right\} \dots \quad (2.1)$$

Persamaan (2.1) merupakan dasar dari algoritma Warshall. Jika matriks  $W_{k-1}$  bernilai 1 pada elemen ke  $(i,j)$  maka dengan persamaan (2.1) (i)  $W_k$  juga bernilai 1 pada elemen ke  $(i,j)$ . Dengan menggunakan persamaan (2.1) (ii), nilai 1 yang baru dapat ditambahkan pada elemen ke  $(i,j)$  pada matrik  $W_k$  jika dan hanya jika kolom ke  $k$  dari  $W_{k-1}$  bernilai 1 di posisi  $i$  dan baris  $k$  dari  $W_{k-1}$  bernilai 1 pada posisi  $j$ . Jadi prosedur untuk menentukan  $W_k$  dari  $W_{k-1}$  sesuai dengan referensi [4] adalah sebagai berikut.

Langkah 1 : Transfer ke  $W_k$  semua nilai 1 dalam  $W_{k-1}$ .

Langkah 2 : Tandai lokasi  $p_1, p_2, \dots$  dalam kolom  $k$  dari  $W_{k-1}$  yang inputnya 1 dan lokasi  $q_1, q_2, \dots$  dalam baris  $k$  dari  $W_{k-1}$  yang inputnya 1.

Langkah 3 : Letakkan 1 pada semua posisi  $p_i, q_j$  yang ditandai dari  $W_k$  (jika elemen pada  $(p_i, q_j)$  bernilai 0)

Contoh 1

$A = \{1, 2, 3, 4\}$  dan relasi pada himpunan  $A$  adalah:

$$R = \{(a,b) \in A \times A \mid a^2 + b^2 = 5 \vee b = a + 1\}, \text{ sehingga } R = \{(1,2), (2,3), (3,4), (2,1)\}.$$

Transitif klosur dari  $R$  dengan algoritma Warshall dapat diselesaikan sebagai berikut.

Matriks relasi  $R$  adalah:

$$M_R = W_0 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Karena himpunan  $A$  mempunyai 4 anggota, maka  $n = 4$ . Kemudian menentukan  $W_k$  untuk  $k = 1, 2, 3$ , dan 4

a. Menentukan  $W_1 (k = 1)$

$W_0$  mempunyai nilai 1 pada lokasi ke 2 dari kolom 1 dan lokasi ke 2 pada baris 1. Jadi  $W_1$  seperti  $W_0$  ditambah nilai 1 yang baru pada elemen ke  $(2,2)$ .

$$W_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

b. Menentukan  $W_2 (k = 2)$

Perhatikan kolom 2 dan baris 2 dari  $W_1$ . Matriks  $W_1$  mempunyai nilai 1 pada lokasi 1 dan 2 dari kolom 2 dan lokasi 1, 2 dan 3 dari baris 2 sehingga untuk mendapatkan  $W_2$ , nilai 1 harus diletakkan pada elemen ke  $(1,1), (1,2), (1,3), (2,1), (2,2)$  dan  $(2,3)$  dari matriks  $W_1$  (jika pada elemen tersebut belum bernilai 1). Jadi:

$$W_2 = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

c. Menentukan  $W_3 (k = 3)$

Kolom ke 3 dari  $W_2$  bernilai 1 pada lokasi 1 dan 2, dan baris ke 3 dari  $W_2$  bernilai 1 pada lokasi 4. Untuk mendapatkan  $W_3$ , pada elemen ke  $(1,4)$  dan  $(2,4)$  dari  $W_2$  harus bernilai 1, Jadi:

$$W_3 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

d. Menentukan  $W_4 (k = 4)$

$W_3$  mempunyai nilai pada lokasi 1, 2, 3 dari kolom 4 dan tidak ada nilai pada baris ke 4, sehingga tidak ada nilai 1 yang baru yang ditambahkan jadi  $M_R^* = W_4 = W_3$ .

$M_R^*$  merupakan matriks transitif klosur dari  $R$ . Jadi transitif klosur dari  $R$  adalah:

$$t(R) = \{(1,1), (1,2), (1,3), (1,4), (2,1), (2,2), (2,3), (2,4), (3,4)\}.$$

**B. Relasi Ekuivalensi**

**Definisi 2.5.** [1] Relasi  $R$  pada himpunan  $i$  disebut *refleksif* jika dan hanya jika  $aRa$  untuk setiap  $a \in A$ .

Dengan melihat matriks suatu relasi dapat ditentukan apakah relasi tersebut refleksif atau bukan. Diagonal utama dari matriks relasi refleksif semuanya bernilai 1.

**Definisi 2.6.** [1] Relasi pada himpunan  $A$  disebut *simetris* jika dan hanya jika  $aRb$  maka  $bRa$ , untuk semua  $a, b \in A$ . Matriks relasi simetris  $M_R = [m_{ij}]$  mempunyai sifat sebagai berikut :

Jika  $m_{ij} = 1$  maka  $m_{ji} = 1$  atau jika  $m_{ij} = 0$  maka  $m_{ji} = 0$ . [4]

Jadi matriks tersebut simetris terhadap diagonal utama atau  $M_R = M_R^T$ .

**Definisi 2.7.** [1] Relasi  $R$  pada himpunan  $A$  disebut *transitif* jika dan hanya jika  $aRb$  dan  $bRc$  maka  $aRc$ , untuk semua  $a, b, c \in A$ . Dilihat dari matriksnya, menentukan relasi transitif tidaklah semudah pada relasi refleksif dan simetris. Matriks relasi transitif  $M_R = [m_{ij}]$  mempunyai ciri-ciri sebagai berikut:

Jika  $m_{ij} = 1$  dan  $m_{jk} = 1$  maka  $m_{ik} = 1$

**Definisi 8** [1] Relasi  $R$  pada himpunan  $A$  disebut *relasi ekuivalensi* jika dan hanya jika  $R$  mempunyai sifat refleksif, simetris dan transitif.

**C. Transitif Klosur dari Gabungan Dua Relasi Ekuivalensi pada Suatu Himpunan**

Salah satu aplikasi transitif klosur adalah pada relasi ekuivalensi. Misal  $R$  dan  $S$  adalah relasi ekuivalensi pada himpunan  $A$ , akan dicari suatu relasi ekuivalensi terkecil yang di dalamnya memuat  $R$  dan  $S$ . Untuk mencari relasi tersebut berarti sama dengan menentukan transitif klosur dari gabungan dua relasi ekuivalensi yang ada ( $R$  dan  $S$ ).

**Definisi 2.9.** [2] Misalkan  $R$  relasi dari  $A$  ke  $B$ . *Invers dari relasi  $R$*  dinotasikan  $R^{-1}$

adalah relasi dari  $B$  ke  $A$  yang didefinisikan sebagai berikut

$$R^{-1} = \{(y,x) / xRy\}.$$

**Teorema 2.4.** [4] Misalkan  $R_1, R_2$  relasi dari  $A$  ke  $B$  maka  $(R_1 \cup R_2)^{-1} = R_1^{-1} \cup R_2^{-1}$ .

**Bukti.**

$$\begin{aligned} (x,y) \in (R_1 \cup R_2)^{-1} &\Leftrightarrow (y,x) \in R_1 \cup R_2 \\ &\Leftrightarrow (y,x) \in R_1 \vee (y,x) \in R_2 \\ &\Leftrightarrow (x,y) \in R_1^{-1} \vee (x,y) \in R_2^{-1} \\ &\Leftrightarrow (x,y) \in R_1^{-1} \cup R_2^{-1} \quad \blacksquare \end{aligned}$$

**Teorema 2.5.** [4] Misalkan  $R$  relasi pada  $A$ .  $R$  simetris jika dan hanya jika  $R = R^{-1}$ .

**Bukti.**

( $\Rightarrow$ ) Bukti ini ada dua bagian, yaitu :

(i). Ambil sebarang  $(x,y) \in R$ . Karena  $R$  simetris maka  $(y,x) \in R$ . Dari definisi invers, maka  $(x,y) \in R^{-1}$ . Sehingga  $R \subseteq R^{-1}$ .

(ii). Ambil sebarang  $(x,y) \in R^{-1}$ . Dari definisi invers maka  $(y,x) \in R$ . Karena  $R$  simetris maka  $(x,y) \in R$ . Sehingga  $R^{-1} \subseteq R$ .

Jadi dari (i) dan (ii), terbukti  $R = R^{-1}$ .

( $\Leftarrow$ ) Ambil sebarang  $(x,y) \in R$ . Dari definisi invers maka  $(y,x) \in R^{-1}$ . Karena  $R = R^{-1}$  maka  $(y,x) \in R$ . Sehingga jika  $(x,y) \in R$  maka  $(y,x) \in R$ . Berdasarkan definisi simetris, maka  $R$  simetris.  $\blacksquare$

**Teorema 6** [4] Jika  $R$  dan  $S$  relasi ekuivalensi pada himpunan  $A$  maka relasi ekuivalensi terkecil yang memuat  $R$  dan  $S$  adalah  $(R \cup S)^*$ .

**Bukti.**

Misalkan  $E$  adalah relasi persamaan (equality relation) yang didefinisikan sebagai berikut :  $E = \{(x,x) / x \in A\}$ . Suatu relasi adalah refleksif jika hanya jika relasi tersebut memuat  $E$ .  $E \subseteq R, E \subseteq S$  sebab keduanya refleksif. Jadi  $E \subseteq R \cup S \subseteq (R \cup S)^*$ , sehingga  $(R \cup S)^*$  juga refleksif. Karena  $R$  dan  $S$  simetris, maka  $R = R^{-1}$  dan  $S = S^{-1}$ . Jadi  $(R \cup S)^{-1} = R^{-1} \cup S^{-1} = R \cup S$ , dan  $R \cup S$  juga simetris. Oleh karena itu, semua path dalam  $R \cup S$  mempunyai dua jalur yang arahnya berlawanan, sehingga  $(R \cup S)^*$  juga simetris. Karena telah

diketahui bahwa  $(R \cup S)^*$  transitif, maka  $(R \cup S)^*$  adalah relasi ekuivalensi yang memuat  $R \cup S$ . Relasi  $(R \cup S)^*$  adalah yang terkecil, sebab tidak ada himpunan terkecil yang memuat  $R \cup S$  yang dapat transitif, sesuai dengan definisi transitif klorur. ■

$(R \cup S)^*$  ini merupakan transitif klorur sekaligus relasi ekuivalensi terkecil dari gabungan dua relasi ekuivalensi  $R$  dan  $S$  pada himpunan  $A$ .

Contoh 2

Misalkan  $A = \{1, 2, 3, 4, 5\}$  dan relasi pada himpunan  $A$  adalah:

$$R = \{(a,b) \in A \times A \mid a=b \vee (a \text{ ganjil}, b=a+1) \vee (a \text{ genap}, b=a-1)\}, \text{ dan}$$

$$S = \{(a,b) \in A \times A \mid a=b \vee a^2+b^2=41\},$$

sehingga:

$$R = \{(1,1), (1,2), (2,1), (2,2), (3,3), (3,4), (4,3), (4,4), (5,5)\}$$

$$S = \{(1,1), (2,2), (3,3), (4,4), (4,5), (5,4), (5,5)\}$$

Jelas bahwa  $R$  dan  $S$  adalah relasi ekuivalensi. Akan dicari relasi ekuivalensi terkecil yang memuat  $R$  dan  $S$ . Matriks dari relasi  $R$  dan  $S$  adalah sebagai berikut:

$$M_R = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ dan}$$

$$M_S = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

$$M_{R \cup S} = M_R \vee M_S = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Untuk mencari  $M_{(R \cup S)^*}$  dengan menggunakan algoritma Warshall. Pertama  $W_0 = M_{R \cup S}$ .

a). Menentukan  $W_1$  ( $k = 1$ )

Karena  $W_0$  mempunyai nilai 1 di lokasi 1 dan 2 dari kolom 1 dan di lokasi 1 dan 2 pada baris 1, maka tidak ada nilai 1 yang baru yang ditambahkan pada  $W_0$ . Jadi  $W_1 = W_0$ .

b). Menentukan  $W_2$  ( $k = 2$ )

Karena  $W_1$  mempunyai nilai 1 di lokasi 1 dan 2 dari kolom 2 dan di lokasi 1 dan 2 dari baris 2, maka tidak ada nilai 1 yang baru yang ditambahkan pada  $W_1$ . Jadi  $W_2 = W_1$ .

c). Menentukan  $W_3$  ( $k = 3$ )

Karena  $W_2$  bernilai 1 di lokasi 3 dan 4 dari kolom 3 dan di lokasi 3 dan 4 dari baris 3, maka tidak ada nilai 1 yang baru yang ditambahkan pada  $W_2$ . Jadi  $W_3 = W_2$ .

d). Menentukan  $W_4$  ( $k = 4$ )

Karena  $W_3$  bernilai 1 di lokasi 3, 4 dan 5 dari kolom 4 dan di lokasi 3, 4 dan 5 dari baris 4, maka untuk membentuk  $W_4$  harus ditambahkan nilai 1 baru pada  $W_3$  di elemen ke (3,5) dan (5,3). Jadi :

$$W_4 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

e). Menentukan  $W_5$  ( $k = 5$ )

Karena  $W_4$  mempunyai nilai 1 pada lokasi 3, 4 dan 5 dari kolom 5 dan lokasi 3, 4 dan 5 dari baris 5, maka tidak ada nilai 1 baru yang ditambahkan pada  $W_5$ . Jadi  $W_5 = W_4$ .

Jadi  $t(R \cup S) = \{(1,1), (1,2), (2,1), (2,2), (3,3), (3,4), (3,5), (4,3), (4,4), (4,5), (5,3), (5,4), (5,5)\}$ . Hasil  $t(R \cup S)$  ini merupakan relasi ekuivalensi terkecil yang memuat  $R$  dan  $S$ .

Penyelesaian dengan menggunakan program dapat dilihat pada contoh output di halaman selanjutnya.

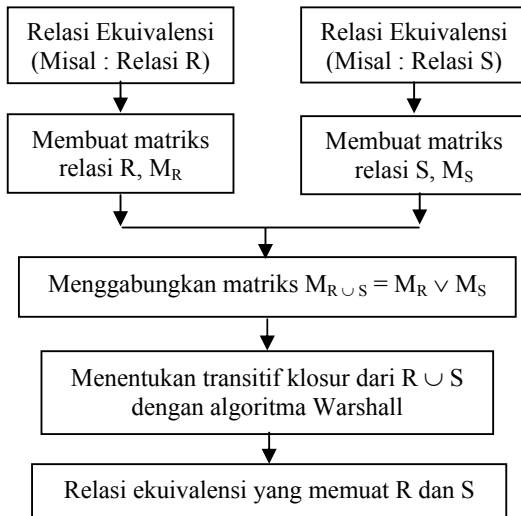
#### D. Implementasi dalam Program Pascal dengan Struktur Data Dinamis

Dari pembahasan sebelumnya, langkah-langkah dalam menentukan transitif klorur dari gabungan dua relasi ekuiva-

lensi pada suatu himpunan agar mendapatkan relasi ekuivalensi terkecil dapat digambarkan pada Gambar 2.

Langkah-langkah tersebut kemudian dibuat ke dalam program dengan bahasa pemrograman Turbo Pascal dan menggunakan struktur data dinamis.

Dalam membuat program, senarai yang digunakan adalah senarai banyak (multilist), sebab terkadang relasi yang ada pada suatu himpunan, hanya melibatkan beberapa elemen yang ada dalam himpunan tersebut. Sehingga struktur simpul matriks relasi dapat dilihat pada Gambar 3.



Gambar 2. Langkah-langkah menentukan relasi ekuivalensi terkecil dari gabungan dua relasi ekuivalensi pada suatu himpunan

Baris	Kolom
Bawah	Kanan

Gambar 3. Struktur simpul untuk matriks relasi

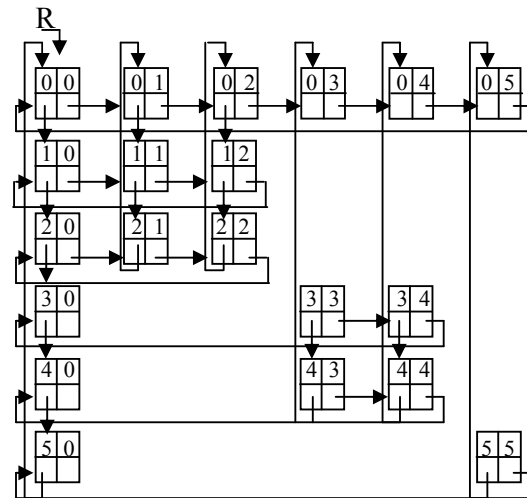
Baris dan Kolom menunjukkan nomor baris dan nomor kolom yang bernilai 1. Tipe Baris dan Kolom adalah integer. Bawah dan Kanan masing-masing adalah pointer yang bertipe sama seperti gambar 3. Pointer Bawah menunjuk ke nomor baris berikutnya, pointer Kanan menunjuk ke kolom berikutnya.

Contoh 3

Matriks relasi R pada contoh 2 =

$$\begin{bmatrix}
 1 & 1 & 0 & 0 & 0 \\
 1 & 1 & 0 & 0 & 0 \\
 0 & 0 & 1 & 1 & 0 \\
 0 & 0 & 1 & 1 & 0 \\
 0 & 0 & 0 & 0 & 1
 \end{bmatrix}$$

Penyajian matriks di atas dengan senarai berantai banyak adalah sebagai berikut.



Gambar 4. Penyimpanan matriks relasi dengan senarai berantai banyak

Deklarasi simpul seperti di atas seperti dalam referensi [3] adalah:

```

type Mat = ^ Elemen;
Elemen = record
    Baris, Kolom : integer;
    Kanan, Bawah : Mat;
end;
  
```

Procedure yang digunakan dalam program adalah:

- Procedure BUAT\_SIMPUL: Digunakan untuk membuat simpul baru.
- Procedure ADA\_BARIS: Digunakan untuk mengecek apakah nomor baris sudah ada.
- Procedure ADA\_KOLOM: Digunakan untuk mengecek apakah nomor kolom sudah ada.
- Procedure SISIP\_ELEMEN: Digunakan untuk menyisipkan elemen matriks pada posisinya yang tepat.

- e) Procedure TAMBAH\_DATA:  
Digunakan untuk menambah data pada relasi yang telah ada.
- f) Procedure CETAK\_MATRIKS:  
Digunakan untuk mencetak bentuk matriks yang bernilai 1 dan 0.
- g) Procedure CETAK\_RELASI:  
Digunakan untuk mencetak relasi dari suatu bentuk matriks.
- h) Procedure GABUNGAN\_MATRIKS:  
Digunakan untuk menggabungkan dua buah matriks relasi.
- i) Procedure TRANSITIF\_KLOSUR:  
Digunakan untuk menentukan transitif klorur dari sebuah matriks.
- j) Procedure CEK\_REFLEKSIF:  
Digunakan untuk mengecek sifat refleksif dari suatu relasi.
- k) Procedure CEK\_SIMETRIS:  
Digunakan untuk mengecek sifat simetris dari suatu relasi.
- l) Procedure CEK\_TRANSITIF:  
Digunakan untuk mengecek sifat transitif dari suatu relasi.

```
|2. Jika Anda ingin mengakhiri |
| pemasukan data, ketikkan   |
| bilangan 0 sesudah kata     |
| Nomor baris                  |
+-----+

```

**Tampilan contoh pemasukan data relasi ekuivalensi (misal untuk data di contoh 2 halaman sebelumnya)**

```
PROGRAM TRANSITIF KLOSUR DARI
GABUNGAN DUA RELASI EKUIVALENSI
DALAM SUATU HIMPUNAN
DENGAN STRUKTUR DATA DINAMIS
```

```
MASUKKAN POSISI ELEMEN YANG
BERNILAI 1 PADA MATRIKS RELASI
EKUIVALENSI PERTAMA:
```

```
Nomor baris : 1   Nomor kolom : 1
Nomor baris : 1   Nomor kolom : 2
Nomor baris : 2   Nomor kolom : 1
Nomor baris : 2   Nomor kolom : 2
Nomor baris : 3   Nomor kolom : 3
Nomor baris : 3   Nomor kolom : 4
Nomor baris : 4   Nomor kolom : 3
Nomor baris : 4   Nomor kolom : 4
Nomor baris : 5   Nomor kolom : 5
Nomor baris : 0
```

Berikut contoh output program:

**Tampilan menu pilihan**

```
PROGRAM TRANSITIF KLOSUR DARI
GABUNGAN DUA RELASI EKUIVALENSI
DALAM SUATU HIMPUNAN
DENGAN STRUKTUR DATA DINAMIS
```

```
>>> MENU PILIHAN <<<
```

- A. INPUT DATA (DUA RELASI EKUIVALENSI)
- B. MENAMBAH DATA
- C. OUTPUT MATRIKS RELASI EKUIVALENSI
- D. OUTPUT MATRIKS GABUNGAN DUA RELASI EKUIVALENSI
- E. TRANSITIF KLOSUR DARI GABUNGAN DUA RELASI EKUIVALENSI
- F. KESIMPULAN
- G. SELESAI

```
MASUKKAN POSISI ELEMEN YANG
BERNILAI 1 PADA MATRIKS RELASI
EKUIVALENSI KEDUA:
```

```
Nomor baris : 1   Nomor kolom : 1
Nomor baris : 2   Nomor kolom : 2
Nomor baris : 3   Nomor kolom : 3
Nomor baris : 4   Nomor kolom : 4
Nomor baris : 4   Nomor kolom : 5
Nomor baris : 5   Nomor kolom : 4
Nomor baris : 5   Nomor kolom : 5
Nomor baris : 0
```

```
>> CEK TERPENUHINYA SYARAT RELASI
EKUIVALENSI <<
```

```
Ingat : Relasi dikatakan relasi
ekuivalensi jika dan hanya
jika relasi tersebut
mempunyai sifat refleksif,
simetris, dan transitif
```

PILIH SALAH SATU (A..G atau a..g):

**Tampilan menu pilihan A (Input Data)**

```
+-----+
|                PERHATIAN                |
|1. Pemasukan data relasi                 |
| ekuivalensi berdasarkan                 |
| matriks relasinya yaitu hanya          |
| pada elemen yang bernilai 1.          |
+-----+

```

```
Relasi PERTAMA merupakan relasi
refleksif
Relasi PERTAMA merupakan relasi
simetris
Relasi PERTAMA merupakan relasi
transitif
Relasi PERTAMA merupakan relasi
ekuivalensi
```



Relasi KEDUA merupakan relasi  
refleksif  
Relasi KEDUA merupakan relasi  
simetris  
Relasi KEDUA merupakan relasi  
transitif  
Relasi KEDUA merupakan relasi  
ekuivalensi

**Tampilan menu pilihan C (Matriks  
Relasi Ekuivalensi)**

PROGRAM TRANSITIF KLOSUR DARI  
GABUNGAN DUA RELASI EKUIVALENSI  
DALAM SUATU HIMPUNAN  
DENGAN STRUKTUR DATA DINAMIS  
  
>> MATRIKS RELASI EKUIVALENSI <<  
=====

MATRIKS RELASI EKUIVALENSI  
PERTAMA:

1	1	0	0	0
1	1	0	0	0
0	0	1	1	0
0	0	1	1	0
0	0	0	0	1

MATRIKS RELASI EKUIVALENSI KEDUA:

1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	1
0	0	0	1	1

**Tampilan menu pilihan D (Matriks  
Gabungan Dua Relasi Ekuivalensi)**

PROGRAM TRANSITIF KLOSUR DARI  
GABUNGAN DUA RELASI EKUIVALENSI  
DALAM SUATU HIMPUNAN  
DENGAN STRUKTUR DATA DINAMIS  
  
>> GABUNGAN DUA BUAH MATRIKS  
RELASI EKUIVALENSI <<  
=====

MATRIKS HASIL GABUNGAN:

1	1	0	0	0
1	1	0	0	0
0	0	1	1	0
0	0	1	1	1
0	0	0	1	1

**Tampilan menu pilihan E (Transitif  
Klosur dari Gabungan Dua Relasi  
Ekuivalensi)**

PROGRAM TRANSITIF KLOSUR DARI  
GABUNGAN DUA RELASI EKUIVALENSI  
DALAM SUATU HIMPUNAN  
DENGAN STRUKTUR DATA DINAMIS

>>TRANSITIF KLOSUR DARI  
GABUNGAN DUA BUAH MATRIKS  
RELASI EKUIVALENSI<<  
=====

Matriks W(1):

1	1	0	0	0
1	1	0	0	0
0	0	1	1	0
0	0	1	1	1
0	0	0	1	1

Matriks W(2):

1	1	0	0	0
1	1	0	0	0
0	0	1	1	0
0	0	1	1	1
0	0	0	1	1

TEKAN ENTER UNTUK MELANJUTKAN !!

Matriks W(3):

1	1	0	0	0
1	1	0	0	0
0	0	1	1	0
0	0	1	1	1
0	0	0	1	1

Matriks W(4):

1	1	0	0	0
1	1	0	0	0
0	0	1	1	1
0	0	1	1	1
0	0	1	1	1

TEKAN ENTER UNTUK MELANJUTKAN !!

Matriks W(5):

1	1	0	0	0
1	1	0	0	0
0	0	1	1	1
0	0	1	1	1
0	0	1	1	1

MATRIKS HASIL TRANSITIF KLOSUR :

1	1	0	0	0
1	1	0	0	0
0	0	1	1	1
0	0	1	1	1
0	0	1	1	1

Jadi transitif klossur dari gabungan dua relasi ekuivalensi (R dan S) adalah

$$t(R \cup S) = \{(1,1), (1,2), (2,1), (2,2), (3,3), (3,4), (3,5), (4,3), (4,4), (4,5), (5,3), (5,4), (5,5)\}$$

### Tampilan menu pilihan F (Kesimpulan)

```
PROGRAM TRANSITIF KLOSSUR DARI
GABUNGAN DUA RELASI EKUIVALENSI
DALAM SUATU HIMPUNAN
DENGAN STRUKTUR DATA DINAMIS
>> KESIMPULAN <<
=====
```

Dua relasi ekuivalensi R dan S dari matriks relasi yang diketahui adalah :

$$R = \{(1,1), (1,2), (2,1), (2,2), (3,3), (3,4), (4,3), (4,4), (5,5)\}$$
$$S = \{(1,1), (2,2), (3,3), (4,4), (4,5), (5,4), (5,5)\}$$

Relasi ekuivalensi terkecil yang mengandung R dan S adalah

$$\{(1,1), (1,2), (2,1), (2,2), (3,3), (3,4), (3,5), (4,3), (4,4), (4,5), (5,3), (5,4), (5,5)\}$$

### 3. PENUTUP

Dari pembahasan yang telah dikemukakan sebelumnya, maka dapat diambil kesimpulan sebagai berikut:

1. Transitif klossur dari suatu relasi merupakan relasi transitif terkecil yang mengandung R, sedangkan transitif klossur dari gabungan dua relasi ekuivalensi

merupakan relasi ekuivalensi terkecil yang mengandung dua relasi yang diketahui.

2. Algoritma Warshall dapat digunakan untuk menentukan relasi ekuivalensi terkecil dari gabungan dua relasi ekuivalensi pada suatu himpunan dengan cara menentukan transitif klossur dari gabungan kedua relasi ekuivalensi tersebut.
3. Penyimpanan matriks relasi dengan senarai berantai banyak dapat menghemat memori sebab elemen nol pada matriks tersebut tidak ikut tersimpan.

### 4. DAFTAR PUSTAKA

- [1] Barnier, W., Feidman, N. (1990), *Introduction to Advanced Mathematics*, Prentice-hall International, New Jersey: 116 - 153.
- [2] Fletcher, Hoyle, Patty (1991), *Foundations of Discrete Mathematics*. PWS Kent Publishing Company, Boston: 319 - 409.
- [3] Insap Santosa, P. (1992) *Struktur Data Menggunakan Turbo Pascal 6.0*, Andi Offset, Yogyakarta: 221-246.
- [4] Kolman, B. (1987), *Discrete Mathematical Structures for Computer Science*, Prentice-hall Inc, New Jersey: 85 - 151.
- [5] Ross, K A., Wright, C.R.B. (1992), *Discrete Mathematics*, Third Edition, Prentice-hall International, New Jersey: 563 - 571.