PELABELAN $L(2,1)$ PADA GRAF *LOLLIPOP* $L_{m,n}$

SKRIPSI

Oleh:

Irham Af'idatul Umam
NIM 161810101028

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2021

PELABELAN $L(2,1)$ PADA GRAF *LOLLIPOP* $L_{m,n}$

SKRIPSI

disusun guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan studi pada Program Studi Matematika (S1)
dan mencapai gelar sarjana Sains

Oleh:

Irham Af'idatul Umam
NIM 161810101028

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2021

PERSEMBAHAN

Dengan mengucapkan rasa syukur atas berkah dari Allah SWT, penulis persembahkan skripsi ini sebagai ungkapan kebahagiaan dan rasa terimakasih kepada:

1. Kedua orang tuaku Ayahanda Mansyur Dasuki dan Ibunda Subanah yang telah memberikan doa, saran serta dukungan penuh selama menempuh pendidikan;
2. Kakak Asrul Huda dan Iib Mubarak, serta Adik Putra Miftakhur Rizky yang telah memberikan motivasi kepada penulis;
3. Kusbudiono, S.Si., M.Si. selaku Dosen Pembimbing Utama dan Dr. Mohamat Fatekurohman, S.Si., M.Si. selaku Dosen Pembimbing Anggota;
4. Seluruh guru dan dosen yang telah memberikan banyak ilmu pengetahuan dan membimbing dengan kesabaran;
5. Almamater Jurusan Matematika dan Ilmu Pengetahuan Alam Universitas Jember, SMA Negeri 20 Jakarta, SMP Negeri 64 Jakarta, dan SD Negeri 01 Kandangtepus

MOTO

Allah akan meninggikan orang-orang yang beriman di antara kamu dan orang-orang yang diberi ilmu pengetahuan beberapa derajat.

(terjemahan Surat *Al-Mujadalah* ayat 11)[*)]

[*)] Departemen Agama Republik Indonesia. 1998. *Al Qur'an dan Terjemahannya*. Semarang: PT Kumudasmoro Grafindo.

PERNYATAAN

Saya yang bertanda tangan di bawah ini

Nama : Irham Af'idatul Umam

NIM : 161810101028

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul "Pelabelan $L(2,1)$ pada Graf *Lollipop* $L_{m,n}$" adalah benar-benar hasil karya ilmiah sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya dan belum pernah diajukan pada institusi manapun, serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, Januari 2021

Yang menyatakan,

Irham Af'idatul Umam

NIM 161810101028

SKRIPSI

PELABELAN $L(2,1)$ PADA GRAF LOLLIPOP $L_{m,n}$

Oleh:

Irham Af'idatul Umam
NIM 161810101028

Pembimbing

Dosen Pembimbing Utama : Kusbudiono, S.Si., M.Si.

Dosen Pembimbing Anggota : Dr. Mohamat Fatekurohman, S.Si., M.Si.

PENGESAHAN

Skripsi berjudul "Pelabelan $L(2,1)$ pada Graf *Lollipop* $L_{m,n}$", telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Tim Penguji:

Ketua,	Anggota I,
Kusbudiono, S.Si., M.Si. NIP. 197704302005011001	Dr. Mohamat Fatekurohman, S.Si., M.Si. NIP. 196906061998031001
Anggota II,	Anggota III,
Dr. Kristiana Wijaya, S.Si., M.Si. NIP. 197408132000032004	Ikhsanul Halikin, S.Pd.,M.Si. NIP. 198610142014041001

Mengesahkan

Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

Universitas Jember

Drs. Achmad Sjaifullah, M.Sc., Ph.D.

NIP. 195910091986021001

RINGKASAN

Pelabelan $\boldsymbol{L(2,1)}$ pada Graf *Lollipop* $\boldsymbol{L_{m,n}}$; Irham Af'idatul Umam, 161810101028; 2021: 28 halaman; Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Jember

Teori graf merupakan cabang ilmu matematika yang telah mengalami banyak perkembangan. Terdapat banyak topik yang dipelajari dalam teori graf, salah satunya adalah pelabelan graf. Secara umum, pelabelan graf membahas tentang pemberian label berupa bilangan bulat pada titik-titik graf, sisi-sisi graf, atau keduanya. Dalam perkembangannya, pelabelan graf ditambah dengan beberapa aturan yang berkaitan dengan jarak. Pelabelan graf dari pengembangan ini biasa disebut pelabelan $L(h,k)$. Kemudian, beberapa peneliti melanjutkan pengembangannya yaitu pelabelan $L(2,1)$ dimana setiap dua titik berjarak satu harus memiliki mutlak selisih label minimal dua dan setiap dua titik berjarak dua harus memiliki mutlak selisih label minimal satu. Dalam penelitian ini, penulis membahas pelabelan $L(2,1)$ pada graf *lollipop* $L_{m,n}$ dengan $m \geq 3$ dan n bilangan bulat positif. Tujuan penelitian ini adalah menentukan nilai minimal *span* (mutlak selisih label terbesar dan terkecil) dari pelabelan $L(2,1)$ pada graf *lollipop* $L_{m,n}$ yang disimbolkan dengan $\lambda_{2,1}(L_{m,n})$, dan membangun program simulasi yang dapat digunakan untuk pelabelan hingga nilai m dan n yang sangat besar.

Terdapat beberapa langkah utama yang dilakukan dlam penelitian ini yaitu pelabelan graf, membentuk fungsi dan teorema, serta membangun program dan dilanjutkan simulasi. Dalam membentuk teorema, langkah pertama yang dilakukan adalah melakukan penotasian titik pada graf. Kemudian dilanjutkan dengan pemberian label dan pendeteksian pola. Selanjutnya adalah membentuk fungsi pelabelan dan menyusun teorema berdasarkan pola yang diperoleh. Langkah terakhir adalah membuktikan teorema. Apabila teorema sudah terbukti, maka langkah selesai. Namun, jika teorema belum terbukti, maka mengulang kembali langkah pelabelan. Program simulasi yang dibangun dalam penelitian ini, dibuat dalam bentuk *Graphical User Interface* (GUI) sehingga mudah dioperasikan.

Graf *lollipop* yang disimbolkan dengan $L_{m,n}$ merupakan graf yang dibentuk dari penggabungan dua graf yaitu graf komplit (K_m) dan graf lintasan (P_n) dengan sebuah sisi penghubung (*bridge*). Graf *lollipop* diberi penotasian himpunan titik dan sisi sebagai berikut, $V = \{u_i \mid 1 \leq i \leq m\} \cup \{v_j \mid 1 \leq j \leq n\}$ dan $E = \{u_i u_k \mid 1 \leq i \leq m-1,\ i+1 \leq k \leq m\} \cup \{u_1 v_1\} \cup \{v_j v_{j+1} \mid 1 \leq j \leq n-1\}$. Dari hasil pelabelan $L(2,1)$ diketahui bahwa untuk sebarang graf *lollipop* ($L_{m,n}$) dengan $m \geq 3$ dan n bilangan bulat positif, minimal *span* pada pelabelan $L(2,1) = 2m - 2$. Fungsi pelabelan yang digunakan untuk memperoleh nilai *span* tersebut adalah $f(u_i) = \{2m - 2i \mid 1 \leq i \leq m\}$ dan $f(v_j) = \{1 \mid j = 1\} \cup \{3 \mid j = 2\} \cup \{2a \mid 3 \leq j \leq n$ dengan a adalah sisa ketika $j - 3$ dibagi $m\}$.

Berdasarkan hasil simulasi dari program yang didasarkan pada fungsi pelabelan, dapat diketahui bahwa mutlak selisih dari setiap dua titik berjarak satu adalah minimal dua dan mutlak selisih dari setiap dua titik berjarak dua adalah minimal satu. Selain itu, nilai *span* atau mutlak selisih dari label terbesar dan label terkecil pada hasil pelabelan merupakan nilai terkecil yang mungkin digunakan dalam pelabelan $L(2,1)$ yaitu $\lambda_{2,1}(L_{m,n}) = 2m - 2$. Nilai minimal *span* graf *lollipop* sama dengan nilai minimal span graf komplit.

PRAKATA

Puji Syukur kehadirat Allah SWT atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul "Pelabelan $L(2,1)$ pada Graf *Lollipop* $L_{m,n}$". Tugas akhir ini disusun untuk memenuhi salah satu syarat dalam menyelesaikan pendidikan strata satu (S1) pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak, oleh karena itu penulis ingin menyampaikan ucapan terima kasih kepada:

1. Bapak Kusbudiono, S.Si., M.Si. selaku Dosen Pembimbing Utama dan Bapak Dr. Mohamat Fatekurohman, S.Si., M.Si. selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, serta perhatian dalam penulisan skripsi ini;
2. Ibu Dr. Kristiana Wijaya, S.Si., M.Si. dan Bapak Ikhsanul Halikin, S.Pd.,M.Si. selaku dosen penguji yang telah memberikan kritik dan saran demi perbaikan tugas akhir ini;
3. Seluruh dosen dan karyawan Fakultas MIPA Universitas Jember yang telah memberikan ilmu serta bimbingannya selama penulis menempuh pendidikan;
4. Kedua orang tua serta seluruh keluarga dan kerabat dekat penulis yang senantiasa memberikan doa dan dukungan;
5. Teman-teman se-angkatan Jurusan Matematika Universitas Jember yang tidak dapat saya sebutkan satu per satu;
6. Semua pihak yang telah memberikan sumbangan tenaga, semangat, dan pikiran yang tidak dapat disebutkan satu persatu oleh penulis.

Penulis juga menerima kritik dan saran dari semua pihak demi kesempurnaan tugas akhir ini. Akhirnya penulis berharap, semoga tugas akhir ini dapat bermanfaat.

Jember, Januari 2021
Penulis

DAFTAR ISI

Halaman

DAFTAR TABEL

DAFTAR GAMBAR

Halaman

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Teori graf merupakan cabang ilmu matematika yang mempelajari sifat-sifat suatu graf. Teori graf telah lama dikenalkan dan berkembang pesat hingga saat ini. Graf digunakan untuk merepresentasikan suatu objek diskrit serta hubungannya dengan objek lain. Sejarah teori graf diawali saat ahli matematika bernama Leonard Euler mampu menyelesaikan permasalahan jembatan Konigsberg pada tahun 1736. Jembatan Konigsberg yang sekarang bernama Kaliningrad berada di kedua sisi sungai Pregal yang mengalir ke pulau Kneiphof dan bercabang menjadi dua sungai. Permasalahan jembatan Konigsberg membahas kemungkinan melalui tujuh buah jembatan tepat satu kali dan kembali ke tempat semula. Leonard Euler menjadi orang pertama yang mampu menyelesaikan permasalahan tersebut kemudian memodelkannya ke dalam graf dengan menyatakan daratan sebagai titik dan jembatan sebagai sisi.

Teori graf merupakan ilmu yang telah banyak dikembangkan, salah satunya yaitu teori pelabelan graf. Pelabelan graf dikenalkan oleh beberapa ilmuwan matematika dengan berbagai pokok bahasan. Secara umum, pelabelan graf membahas tentang suatu graf yang memiliki titik dan sisi serta pelabelan pada titik, sisi atau keduanya dengan menggunakan himpunan bilangan bulat non-negatif.

Pelabelan graf salah satunya dapat digunakan dalam pengaplikasian frekuensi pemancar radio. Pembahasan ini mengkaji penentuan frekuensi di setiap pemancar radio, sehingga jika terdapat lebih dari satu pemancar maka akan memiliki frekuensi yang berbeda. Pengaplikasian ini menggunakan himpunan bilangan non-negatif untuk saluran radio secara terstruktur pada setiap pemancar di lokasi tertentu.

Pembahasan penentuan frekuensi pemancar radio membahas tentang pemberian frekuensi pada setiap pemancar radio sehingga jika terdapat dua pemancar yang berdekatan, maka pemancar tersebut diberikan frekuensi yang berbeda. Pemancar yang berdekatan harus menerima frekuensi dengan selisih yang cukup untuk menghindari gangguan. Permasalahan ini berawal dari pembicaraan

Fred Roberts dan Jerrold Griggs yang berencana menggunakan bilangan non-negatif untuk mewakili saluran radio dalam mempelajari permasalahan penentuan saluran radio secara optimal pada pemancar di lokasi tertentu. Hasilnya, Griggs dan Yeh pada tahun 1992 memperkenalkan pelabelan $L(h,k)$, yaitu pelabelan yang diberikan pada titik suatu graf yang tidak hanya bergantung pada dua titik bertetangga (berjarak satu), tetapi juga berjarak dua.

Kajian dan pembahasan mengenai teori pelabelan $L(2,1)$ telah banyak dikembangkan oleh peneliti sebelumnya. Noviana membahas pelabelan $L(2,1)$ pada graf hasil operasi *comb* antara graf lintasan dan graf sikel pada tahun 2017 yang menghasilkan nilai minimal label terbesar. Fatimah dkk. membahas pelabelan $L(2,1)$ pada operasi beberapa kelas graf di tahun 2016. Prasetyo membahas pelabelan $L(2,1)$ pada graf *cycle*, graf *star* dan graf *wheel* pada tahun 2011. Di tahun 2010, Miryawati membahas tentang pelabelan $L(2,1)$ pada graf bidang ubin reguler dan graf *outerplanar*. Penelitian-penelitian tersebut menghasilkan penentuan nilai minimal *span* dengan aturan pelabelan $L(2,1)$.

Kajian graf yang digunakan sebagai objek pada teori pelabelan $L(2,1)$ secara umum telah banyak dilakukan oleh beberapa peneliti. Secara garis besar materi pelabelan $L(2,1)$ termasuk ke dalam cabang bidang ilmu teori graf yang dapat diinterpretasikan secara mudah. Dasar-dasar teori pelabelan $L(2,1)$ dapat dianggap mudah dipahami dan dimengerti sehingga banyak beberapa peneliti memilih untuk menyelesaikan permasalahan pada kajian tersebut. Selain itu, terdapat beberapa kelas graf yang belum diinterpretasikan ke dalam teori pelabelan $L(2,1)$, salah satunya adalah graf *lollipop*. Oleh karena itu, maka peneliti akan mengembangkan teori graf di bidang pelabelan $L(2,1)$ pada graf *lollipop* $L_{(m,n)}$. Selain itu, peneliti juga akan membuat program simulasi untuk memudahkan pelabelan $L(2,1)$ pada graf *lollipop* $L_{m,n}$.

1.2 Rumusan Masalah

Berdasarkan latar belakang diperoleh permasalahan yaitu berapa nilai minimal *span* dari pelabelan $L(2,1)$ pada graf *lollipop* $L_{(m,n)}$ dan bagaimana program simulasi pelabelan $L(2,1)$ pada graf *lollipop* $L_{(m,n)}$.

1.3 Tujuan Penelitian

Tujuan yang ingin dicapai dari penelitian ini adalah untuk menentukan nilai minimal *span* dari pelabelan $L(2,1)$ pada graf *lollipop* $L_{(m,n)}$ dan untuk membangun program simulasi pelabelan $L(2,1)$ pada graf *lollipop* $L_{(m,n)}$.

1.4 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat serta dampak yang bernilai positif dengan paparan sebagai berikut.

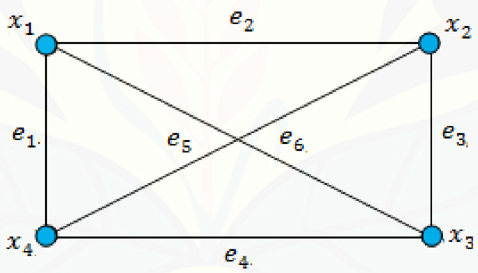
a. Bagi peneliti, diharapkan penelitian ini akan memberikan sebuah manfaat bagi peneliti berupa pembelajaran untuk memperoleh nilai minimal *span* dari pelabelan $L(2,1)$ pada graf *lollipop* $L_{(m,n)}$.
b. Bagi mahasiswa, diharapkan penelitian ini akan memberikan dampak yang bernilai positif serta dapat menjadi referensi tambahan mengenai kajian ilmu teori graf pada teori pelabelan $L(2,1)$.

BAB 2. TINJAUAN PUSTAKA

Pada bab ini, diuraikan beberapa dasar teori yang digunakan dalam penelitian ini yang antara lain meliputi: definisi dan terminologi graf, kelas-kelas graf, jarak dan lintasan pada graf, pelabelan graf, dan pelabelan $L(2,1)$.

2.1 Definisi dan Terminologi Graf

Sebuah graf G yang dinyatakan dalam $(V(G), E(G))$ merupakan pasangan dari dua himpunan. $V(G) = \{v_1, v_2, \dots, v_n\}$ merupakan sebuah himpunan tak kosong dari elemen yang disebut titik dan $E(G) = \{e_1, e_2, \dots, e_n\}$ merupakan sebuah himpunan yang mungkin kosong dari sebuah pasangan tak terurut $\{v_1, v_2\}$ dari titik $v_1, v_2 \in V(G)$ yang disebut himpunan sisi dari G (Slamin, 2009). Berikut ditunjukkan contoh dari sebuah graf G yang disajikan pada Gambar 2.1 dengan 4 titik dan 6 sisi.



Gambar 2.1 Contoh graf G dengan 4 titik dan 6 sisi

Menurut Chartrand dan Oellermann (1993), *order* dari G didefinisikan sebagai banyaknya titik yang terdapat pada graf G, sedangkan *size* dari G didefinisikan sebagai banyaknya sisi yang terdapat pada graf G. Secara umum, penotasian-penotasian dari suatu graf G dinyatakan sebagai p atau $p(G)$ untuk notasi *order* dan q atau $q(G)$ untuk *size* dari graf G. Sebagai contoh (lihat Gambar 2.1), *order* dari graf G tersebut adalah empat, dan *size*-nya adalah enam. Graf berdasarkan keterhubungannya (*connectivity*) dibagi menjadi dua, yaitu graf terhubung (*connected graph*) dan graf tak terhubung (*disconnected graph*). Graf G dikatakan terhubung (*connected*) jika untuk setiap pasangan titik v_i dan v_j di dalam himpunan V terdapat lintasan dari v_i dan v_j, sedangkan graf G dikatakan tak terhubung (*disconnected*) jika ada pasangan titik v_i dan v_j di dalam himpunan V tidak terdapat lintasan dari v_i dan v_j.

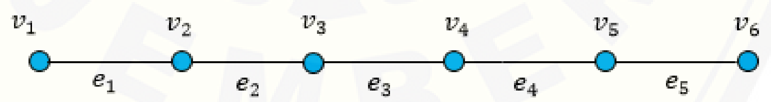
Dua buah titik pada graf G dikatakan bertetangga (*adjacent*) apabila kedua titik tersebut terhubung secara langsung oleh sisi. Hal tersebut, dapat dilihat pada Gambar 2.1, bahwa titik x_1 bertetangga dengan titik x_2, x_3, x_4. Titik v dikatakan bersisian (*incident*) dengan sisi e apabila titik v adalah titik ujung dari sisi e, sama halnya dengan sisi e dikatakan bersisian dengan titik v apabila sisi e memiliki ujung titik v. Sebagai contoh pada Gambar 2.1, sisi e_2 bersisian dengan titik x_1 dan x_2. Selain istilah bertetangga dan bersisian, dalam graf juga dikenal istilah derajat (*degree*). Derajat dari suatu titik u pada graf G yang dinotasikan dengan $d(u)$ adalah banyaknya sisi yang bersisian dengan titik u. Jika suatu titik mempunyai derajat satu, maka titik tersebut disebut daun atau anting (Budayasa, 2007).

2.2 Kelas-kelas Graf

Pada subbab ini diberikan dan dijelaskan beberapa paparan jenis-jenis graf.

a. Graf lintasan

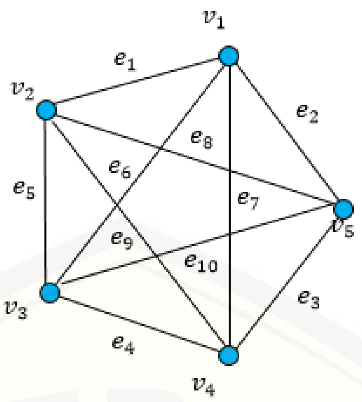
Penotasian graf lintasan (*path graph*) dapat dituliskan dengan P_n dan dapat didefinisikan sebagai graf terhubung (*connected*) dalam suatu lintasan yang mengandung n titik dan $n-1$ sisi dengan dua titik yang berada di ujung memiliki derajat yaitu $\delta(P_n) = 1$ dan $\Delta(P_n) = 2$ (Gross dan Yellen, 2006). Gambar 2.2 merupakan contoh representasi dari graf lintasan dengan $n = 6$.



Gambar 2.2 Graf lintasan P_6

b. Graf lengkap

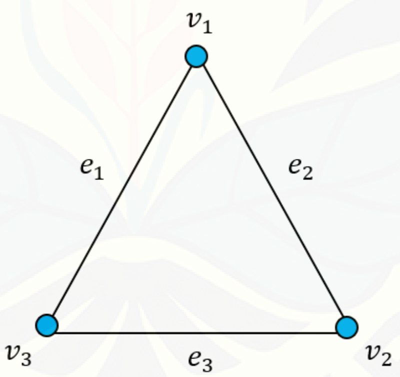
Penotasian graf lengkap (*complete graph*) dapat dituliskan dengan K_n dan didefinisikan sebagai graf yang memiliki n titik yang setiap titiknya bertetangga dengan titik-titik yang lain serta memiliki derajat $\Delta(K_n) = n - 1$. Berikut merupakan sebuah contoh graf lengkap K_5 yang disajikan pada Gambar 2.3.



Gambar 2.3 Graf lengkap K_5

c. Graf *Cycle*

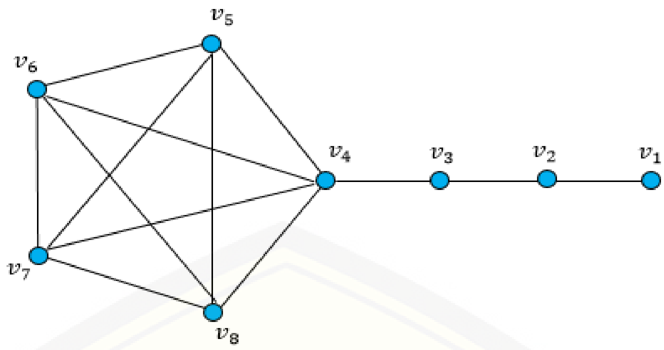
Graf *Cycle* yang dinotasikan dengan C_m adalah graf lintasan P_n yang ujung-ujungnya dihubungkan oleh sebuah sisi. Graf C_m untuk $m \geq 3$ memiliki m titik dan m sisi. Gambar 2.4 merupakan contoh dari graf *cycle* yang memiliki 3 titik dan 3 sisi.



Gambar 2.4 Graf *cycle* C_3

d. Graf *lollipop*

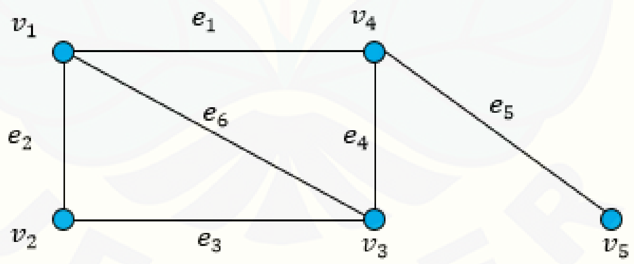
Graf *lollipop* merupakan graf yang diperoleh dengan menggabungkan graf lengkap (*complete graph*) K_n dengan graf lintasan (*path graph*) P_n dengan suatu jembatan (sisi). Penotasian graf *lollipop* dapat dituliskan dengan $L_{m,n}$ (Vijayan dan Nagarajan, 2015). Gambar 2.4 merupakan contoh dari graf *lollipop* $L_{5,3}$.



Gambar 2.4 Graf *lollipop* $L_{5,3}$

2.3 Jarak dan Lintasan pada Graf

Misalkan G merupakan suatu graf dengan himpunan titik $V(G)$ dan himpunan sisi $E(G)$. Jarak dari titik u ke v pada graf G didefinisikan sebagai panjang suatu lintasan terpendek dari u ke v dan dinyatakan dengan $d(u,v)$. Lintasan merupakan suatu barisan berselang-seling pada suatu titik-titik dan sisi-sisi yang berbentuk $v_0, e_1, v_1, e_2, \dots, v_{n-1}, v_0, e_n, v_n$ sedemikian sehingga $e_1 = (v_0, v_1), e_2 = (v_1, v_2), \dots$, $e_n = (v_{n-1}, v_n)$ merupakan sisi-sisi dari graf G. Dari Gambar 2.7, contoh lintasan adalah $v_1 - e_2 - v_2 - e_3 - v_3 - e_4 - v_4$.



Gambar 2.5 Graf dengan 5 titik dan 6 sisi

Misalkan dari Gambar 2.5 dicari jarak dari titik v_2 ke titik v_2. Terdapat empat lintasan yang mungkin dilalui yaitu v_2, e_2, v_1, e_1, v_4; $v_2, e_2, v_1, e_6, v_3, e_4, v_4$; v_2, e_3, v_3, e_4, v_4 dan $v_2, e_3, v_3, e_6, v_1, e_1, v_4$. Berdasarkan definisi jarak, maka dari empat lintasan tersebut diketahui bahwa jarak titik v_2 ke titik v_4 sama dengan dua, yaitu panjang lintasan v_2, e_2, v_1, e_1, v_4 dan v_2, e_3, v_3, e_4, v_4 karena hanya melewati dua sisi.

2.4 Pelabelan Graf

Pelabelan pada suatu graf merupakan sebuah fungsi yang memetakan antara titik ataupun sisi atau keduanya pada suatu bilangan bulat non negatif dengan sifat tertentu. Pelabelan graf berdasarkan sebuah domainnya dibagi menjadi tiga jenis yaitu pelabelan sisi, pelabelan titik, dan pelabelan total. Pelabelan titik pada suatu graf didefinisikan sebagai pelabelan yang menggunakan domain himpunan titik (*vertex labeling*). Pelabelan pada sisi dinyatakan dengan pelabelan pada domain himpunan sisi (*edge labeling*). Pelabelan total berlaku kedua-duanya dan pelabelan ini menggunakan domain himpunan titik dan himpunan sisi (*total labeling*).

2.4.1 Pelabelan $L(2,1)$

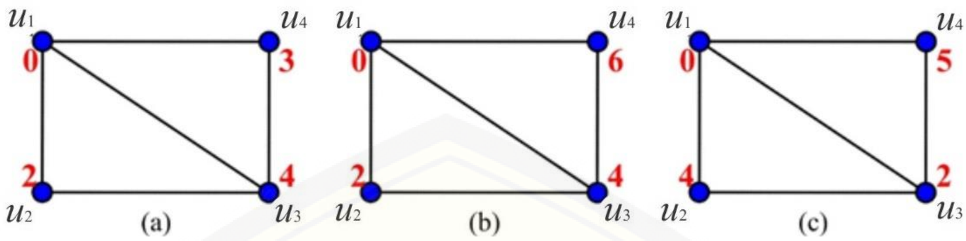
Pelabelan $L(2,1)$ pada graf G menurut Griggs dan Yeh (1992) didefinisikan sebagai suatu fungsi f yang memetakan himpunan titik $V(G)$ ke bilangan bulat non-negatif sedemikian sehingga $|f(u) - f(w)| \geq 2$ jika $d(u,w) = 1$ dan $|f(u) - f(w)| \geq 1$ jika $d(u,w) = 2$ dengan $d(u,w)$ adalah jarak antara titik u dan w. Mutlak selisih dari label terbesar dan label terkecil dari graf disebut dengan *span*. Dalam pelabelan $L(2,1)$ pada graf G, *span* minimum yang diambil dari semua label dikenal dengan istilah $L(2,1)$-*labeling number*, dan dinotasikan dengan $\lambda_{2,1}(G)$. Menurut Shao dkk. (2008), dengan mempertimbangkan bahwa pelabelan $L(2,1)$ pada graf G memuat bilangan nol (0), didefinisikan pelabelan $k - L(2,1)$ yaitu pelabelan $L(2,1)$ dengan nilai terbesar label yang tidak melebihi k, maka $\lambda_{2,1}(G)$ adalah nilai minimum dari k sedemikian sehingga graf G dilabeli dengan pelabelan $k - L(2,1)$.

Teorema 2.1 *(Lum, 2007) Jika H adalah subgraf dari G, maka $\lambda_{2,1}(H) \leq \lambda_{2,1}(G)$.*

Teorema 2.2 *(Lum, 2007) Untuk sebarang graf komplit K_n dengan n bilangan bulat positif, $\lambda_{2,1}(K_n) = 2n - 2$.*

Teorema 2.3 *(Lum, 2007) Untuk sebarang graf lintasan P_n dengan $n \geq 5$, $\lambda_{2,1}(P_n) = 4$.*

Misalkan diberi sebuah graf G dengan himpunan titik $V(G) = \{u_1, u_2, u_3, u_4\}$ dan himpunan sisi $E(G) = \{u_1u_2, u_1u_3, u_1u_4, u_2u_3, u_3u_4\}$. Berikut diberikan contoh pencarian nilai minimal *span* dari pelabelan $L(2,1)$ pada graf G tersebut dan direpresentasikan dalam Gambar 2.6.



Gambar 2.6 Contoh pelabelan $L(2,1)$ pada graf G

Dari Gambar 2.6 (a) terlihat bahwa mutlak selisih dari label titik u_3 dan u_4 adalah satu, sedangkan berdasarkan syarat pelabelan $L(2,1)$ yaitu mutlak selisih dua titik berjarak satu adalah minimal dua. Hal ini berarti bahwa pelabelan graf pada Gambar 2.6 (a) bukan merupakan pelabelan $L(2,1)$. Selanjutnya, apabila diperhatikan Gambar 2.6 (b) dan (c) telah memenuhi kedua syarat pelabelan $L(2,1)$ yaitu mutlak selisih dua titik berjarak satu adalah minimal dua dan mutlak selisih dua titik berjarak dua adalah minimal satu. Pada Gambar 2.6 (b) terlihat bahwa label terkecilnya adalah nol (0) dan label terbesarnya adalah enam (6), sehingga *span* yang dihasilkan sama dengan enam (6). Pada Gambar 2.6 (c) dapat dilihat bahwa label terkecil dan terbesarnya berturut-turut adalah nol (0) dan lima (5), sehingga *span* yang didapatkan adalah lima (5). Berdasarkan hasil pelabelan pada Gambar 2.6 (b) dan (c), dapat disimpulkan bahwa nilai *span* minimal dari pelabelan $L(2,1)$ yaitu lima (5).

2.4.2 Hasil Penelitian Terdahulu

Berdasarkan penelitian sebelumnya, terdapat beberapa teorema mengenai nilai minimal *span* dari pelabelan $L(2,1)$ pada beberapa kelas graf. Berikut hasil penelitian-penelitian terdahulu tersebut yang telah dirangkum dan disajikan pada Tabel 2.1.

Tabel 2.1 Hasil pelabelan $L(2,1)$ penelitian terdahulu

Graf	**Nilai Minimal *Span***	**Peneliti**
Graf Bintang $S_{1,n}$	$\lambda_{2,1}(S_{1,n}) = n + 1$	Griggs dan Yeh (1992)
Graf Sikel C_n	$\lambda_{2,1}(C_n) = 4, n \geq 3$	Lum (2007)
Graf Lintasan P_n	$\lambda_{2,1}(P_n) = 4, n \geq 5$	Lum (2007)
Graf Komplit K_n	$\lambda_{2,1}(K_n) = 2n - 2$	Lum (2007)
Graf *Super Cycle* $Sc_{n,r}$	$\lambda_{2,1}(Sc_{n,r}) = 6, r \geq 2$	Karimah (2016)
Graf Kipas F_n	$\lambda_{2,1}(F_n) = n + 1$	Fatimah dkk. (2016)
Graf Teratai T_n	$\lambda_{2,1}(T_n) = 2n + 2$	Fatimah dkk. (2016)
Graf Roda W_n	$\lambda_{2,1}(W_n) = n + 1$	Fatimah dkk. (2016)
Graf $K_1 \odot (P_n \cup F_n)$	$\lambda_{2,1}(K_1 \odot (P_n \cup F_n)) = 2n + 2$	Fatimah dkk. (2016)
Graf Sierpinski $S_{n,2}$	$\lambda_{2,1}(S_{n,2}) = 4,\ n \geq 3$	Sagala (2016)
Graf Sierpinski $S_{n,3}$	$\lambda_{2,1}(S_{n,3}) = 4,\ n \geq 3$	Sagala (2016)
Graf Petersen $P(n,1)$	$\lambda_{2,1}(S_{n,3}) = \begin{cases} 6 & ; n = P(3k,1) \\ 7 & ; n = P(l + 12k, 1) \end{cases}$	Aminullah (2019)
Graf Buku Segitiga $K_{1,1,n}$	$\lambda_{2,1}(K_{1,1,n}) = n + 1$	Komarullah (2020)
Graf Kerucut $C_{m,n}$	$\lambda_{2,1}(C_{m,n}) = n + 5,\ m \geq 3$	Komarullah (2020)
Graf *Tadpole* $T_{m,n}$	$\lambda_{2,1}(T_{m,n}) = 4,\ m \geq 3$	Komarullah (2020)
Graf *Dumbbell* $D_{m,n,o}$	$\lambda_{2,1}(D_{m,n,o}) = 4$	Komarullah (2020)
Graf $K_{1,1,m} \odot P_n$	$\lambda_{2,1}(K_{1,1,m} \odot P_n) = m + 3$	Komarullah (2020)

BAB 3. METODE PENELITIAN

3.1 Jenis dan Objek Penelitian

Penelitian ini tergolong dalam penelitian eksploratif yang dimaksudkan untuk menambah wawasan serta memunculkan ide-ide baru, dan juga hasil dari penelitian ini dapat digunakan sebagai tambahan rujukan. Penelitian ini menggunakan metode pendeteksian pola, yaitu dengan mencari pola pelabelan $L(2,1)$ sehingga diperoleh bentuk umumnya. Objek yang digunakan dalam penelitian ini adalah graf *lollipop* $L_{m,n}$. Graf ini di-*expand* dengan $m \geq 3$ dan n bilangan bulat positif.

3.2 Langkah-langkah Penelitian

Langkah-langkah yang dilakukan untuk mencapai tujuan dalam penelitian ini adalah sebagai berikut.

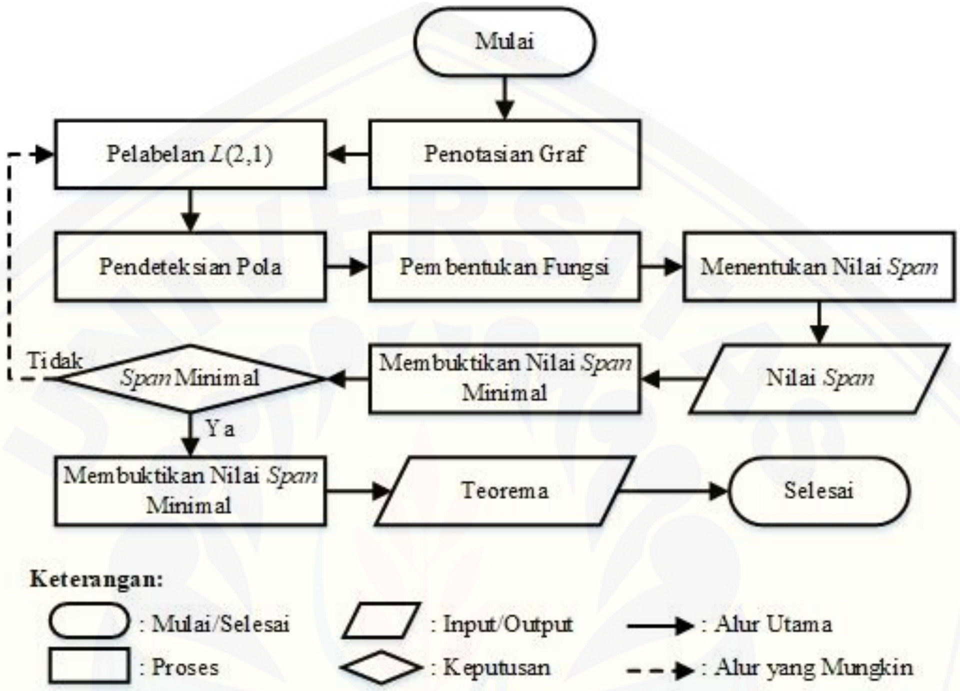
a. Pelabelan $L(2,1)$ pada graf *lollipop*

Langkah pelabelan merupakan langkah utama yang dilakukan penulis dalam penelitian ini. Terdapat beberapa kegiatan yang dilakukan pada tahap ini, antara lain sebagai berikut.

1) Melakukan penotasian titik pada graf *lollipop* $L_{m,n}$ dengan $m \geq 3$ dan n bilangan bulat positif.
2) Melakukan pelabelan $L(2,1)$ terhadap graf *lollipop* $L_{m,n}$.
3) Mendeteksi pola pelabelan sebagai dasar pembentukan fungsi dan teorema.
4) Membentuk fungsi pelabelan dan menentukan nilai *span*.
5) Membuktikan *span* bernilai minimal berdasarkan batas bawah dan batas atas. Pembuktian batas bawah dapat dilakukan dengan menggunakan Teorema 2.1. Namun, untuk pembuktian batas atas dapat dilakukan melalui pengecekan selisih mutlak dari setiap label titik berjarak satu dan dua berdasarkan fungsi pelabelan yang telah dibuat.
6) Apabila nilai *span* sudah minimal, maka langkah pelabelan selesai. Namun, jika nilai *span* belum minimal, maka mengulang langkah 2) sampai langkah 5) hingga nilai *span* sudah minimal.

7) Membentuk teorema nilai minimal *span* pelabelan $L(2,1)$ pada graf *lollipop*.

Untuk memperjelas proses dari langkah-langkah pelabelan $L(2,1)$ pada graf *lollipop* $L_{m,n}$, maka diilustrasikan dalam diagram alir Gambar 3.1.



Gambar 3.1 Diagram alir pelabelan $L(2,1)$ graf *lollipop* $L_{m,n}$

b. Pembuatan program

Pada tahap ini, penulis membuat program simulasi sebagai alat bantu pelabelan $L(2,1)$ pada graf *lollipop* $L_{m,n}$. Program simulasi ini dibuat dengan bentuk *Graphical User Interface* (GUI). Pelabelan yang diterapkan dalam program ini disesuaikan dengan fungsi dan teorema yang telah diselesaikan pada tahap sebelumnya. Tujuan pembuatan program ini adalah agar pelabelan graf *lollipop* dengan *expand* nilai m dan n yang besar tidak perlu dilakukan secara manual.

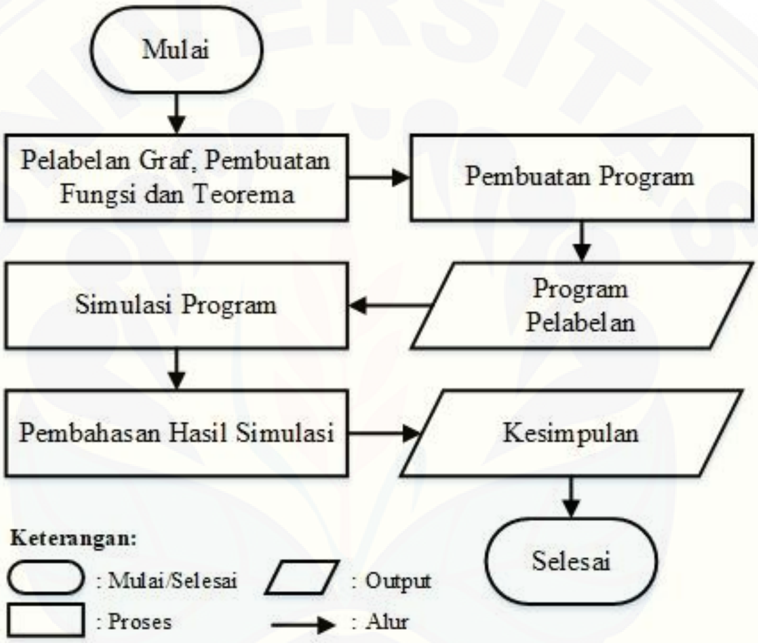
c. Simulasi program

Setelah program selesai dibuat, pada tahap ini penulis melakukan simulasi pelabelan $L(2,1)$ pada graf *lollipop* $L_{m,n}$ dengan beberapa nilai m dan n. Pemilihan nilai m dan n yang digunakan merepresentasikan tiga hubungan dari dua nilai tersebut, yaitu $m < n$, $m = n$ dan $m > n$. Pada simulasi ini juga digunakan nilai m dan n yang besar untuk menguji bahwa teorema dan fungsi pelabelan yang telah terbentuk sebelumnya berlaku pada *expand* graf yang besar.

d. Pembahasan hasil simulasi

Pada tahap ini, hasil simulasi yang dilakukan dibahas kembali untuk menjawab rumusan masalah serta tujuan penelitian, hingga mendapatkan kesimpulan.

Langkah-langkah penelitian secara sistematis diilustrasikan dalam diagram alir pada Gambar 3.2.



Gambar 3.2 Diagram alir langkah penelitian

BAB 5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dan pembahasan yang telah diuraikan pada bab 4, dapat disimpulkan bahwa nilai minimal *span* dari pelabelan $L(2,1)$ pada graf *lollipop* $(L_{m,n})$ adalah $2m - 2$ dengan $m \geq 3$ dan n bilangan bulat positif. Nilai minimal *span* graf *lollipop* sama dengan nilai minimal span graf komplit. Penelitian ini juga menghasilkan program simulasi pelabelan $L(2,1)$ pada graf *lollipop* $(L_{m,n})$. Program tersebut dapat digunakan dengan memasukkan nilai m dan n kemudian menekan tombol, maka graf *lollipop* digambarkan beserta notasi dan pelabelannya.

5.2 Saran

Saran untuk penelitian selanjutnya adalah mengembangkan pelabelan $L(2,1)$ titik maupun sisi pada beberapa kelas graf atau operasi graf lainnya. Selain itu, penelitian selanjutnya juga dapat mengembangkan pelabelan $L(3,2,1)$ serta menerapkan pelabelan $L(2,1)$ maupun $L(3,2,1)$ dalam permasalahan di kehidupan sehari-hari.

DAFTAR PUSTAKA

Aminullah, M.R.A. 2019. Minimal Label Terbesar dari Pelabelan Titik dan Sisi $L(2,1)$ pada Graf Petersen $P(n,1)$. *Skripsi*. Malang: Universitas Islam Negeri Maulana Malik Ibrahim.

Budayasa, K. 2007. *Teori Graph dan Aplikasinya*. Surabaya: Unesa University Press.

Chartrand, G., dan O.R. Oellermann. 1993. *Applied and Algoritmic Graph Theory*. Mcgraw: Hill College.

Fatimah, S., I.W. Sudarsana, dan S. Musdalifah. 2016. Pelabelan $L(2,1)$ pada Operasi Beberapa Kelas Graf. *Jurnal Ilmiah Matematika dan Terapan*. 13(2): 73-84.

Griggs, J.R., dan R.K. Yeh. 1992. Labelling Graphs with a Condition at Distance 2. *SIAM Journal on Discrete Mathematics*. 5(4): 586-595.

Gross, J.L., dan J. Yellen. 2006. *Graph Theory and Its Applications*. Boca Raton: Chapman & Hall/CRC

Karimah, L.N. 2016. Pelabelan $L(2,1)$ pada Graf Super Cycle. *Skripsi*. Malang: Universitas Islam Negeri Maulana Malik Ibrahim.

Komarullah, H. 2020. Pelabelan $L(2,1)$ pada Graf Buku Segitiga, Graf Kerucut, Graf Tadpole dan Graf Dumbbell serta Graf Hasil Identifikasi Titik dari Graf Buku Segitiga dan Graf Lintasan. *Skripsi*. Jember: Universitas Jember.

Lum, A. 2007. *Upper Bounds on the $L(2,1)$-labeling Number of Graphs with Maximum Degree Δ*.Walla-Walla, Washington: Whitman Colllege.

Miryawati, L. 2010. Pelabelan $L(2,1)$ pada Graf Bidang Ubin Reguler dan Graf Outerplanar. *Skripsi*. Semarang: Universitas Diponegoro.

Noviana, G. 2017. Pelabelan $L(2,1)$ pada Graf Hasil Operasi Comb antara Graf Lintasan dan Graf Lingkaran. *Skripsi*. Bandung: Universitas Islam Negeri Sunan Gunung Djati.

Sagala, Y. 2016. Pelabelan $L(2,1)$ pada Graf Sierpinski $S(n,k)$. *KARISMATIKA: Kumpulan Artikel Ilmiah, Informatika, Statistik, Matematika dan Aplikasi*. 3(2): 130-139.

Shao, Z., R.K. Yeh, K.K. Poon, dan W.C. Shiu. 2008. The $L(2,1)$-labeling of $K1$, n-Free Graphs and Its Applications. *Applied Mathematics Letters*. 21(11): 1188-93.

Slamin. 2009. *Desain Jaringan: Pendekatan Teori Graf*. Jember: Universitas Jember Press.

Vijayan, A., dan T. Nagarajan. 2015. Vertex-Edge Domination Polynomials of Lollipop Graphs. *International Journal of Scientific and Innovative Mathematical Research*. 3(4): 39-44.

LAMPIRAN

```
function varargout = Lollipop(varargin)
% LOLLIPOP MATLAB code for Lollipop.fig
%      LOLLIPOP, by itself, creates a new LOLLIPOP or raises the
existing
%      singleton*.
%
%      H = LOLLIPOP returns the handle to a new LOLLIPOP or the
handle to
%      the existing singleton*.
%
%      LOLLIPOP('CALLBACK',hObject,eventData,handles,...) calls
the local
%      function named CALLBACK in LOLLIPOP.M with the given input
arguments.
%
%      LOLLIPOP('Property','Value',...) creates a new LOLLIPOP or
raises the
%      existing singleton*.  Starting from the left, property
value pairs are
%      applied to the GUI before Lollipop_OpeningFcn gets called.
An
%      unrecognized property name or invalid value makes property
application
%      stop.  All inputs are passed to Lollipop_OpeningFcn via
varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows
only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help Lollipop

% Last Modified by GUIDE v2.5 06-Dec-2020 14:44:12

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Lollipop_OpeningFcn, ...
                   'gui_OutputFcn',  @Lollipop_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
```

```
end
% End initialization code - DO NOT EDIT


% --- Executes just before Lollipop is made visible.
function Lollipop_OpeningFcn(hObject, eventdata, handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Lollipop (see VARARGIN)

% Choose default command line output for Lollipop
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
clc;
movegui(gcf,'center');
set(gcf,'Name','Pelabelan L(2,1) Graf Lollipop');
set(handles.edit1,'String','');
set(handles.edit2,'String','');
set(handles.text5,'String','L_m,n');
set(handles.text7,'String','2m-2');
cla(handles.axes1,'reset');
set(handles.axes1,'XTick','','YTick','');
% UIWAIT makes Lollipop wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = Lollipop_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
set(handles.text5,'String','L_m,n');
cla(handles.axes1,'reset');
set(handles.axes1,'XTick','','YTick','');
```

```
set(handles.pushbutton2,'UserData',[]);

% Parameter
m = str2num(get(handles.edit1,'String'));
n = str2num(get(handles.edit2,'String'));

% Membentuk titik
U(:,1) = [0;0];
V = [(1:n);zeros(1,n)];
Theta1 = 90 * (m+2) / m;
Theta2 = 180 * (m-2) / m;
U(:,2) = [cosd(Theta1) -sind(Theta1);sind(Theta1) cosd(Theta1)] *
(V(:,1)-U(:,1)) + U(:,1);
for i = 3 : m
    U(:,i) = [cosd(Theta2) sind(Theta2);-sind(Theta2)
cosd(Theta2)] * (U(:,i-2)-U(:,i-1)) + U(:,i-1);
end

% Plot garis
hold on
for i = 1 : m-1
    for k = i+1 : m
        plot([U(1,i),U(1,k)],[U(2,i),U(2,k)],'k');
    end
end
plot([U(1,1),V(1,1)],[U(2,1),V(2,1)],'k');
for j = 1 : n-1
    plot([V(1,j),V(1,j+1)],[V(2,j),V(2,j+1)],'k');
end

% Plot titik
for i = 1 : m
    line(U(1,i),U(2,i),'Marker','o','MarkerFaceColor','b');
    if m >= 20 || n >= 20
        text(U(1,i),U(2,i),sprintf(['u' num2str(i)
'\n']),'Color','black','FontWeight','bold','FontSize',8);
    elseif (m >= 10 && m < 20) || (n >= 10 && n < 20)
        text(U(1,i),U(2,i),sprintf(['u' num2str(i)
'\n']),'Color','black','FontWeight','bold','FontSize',10);
    else
        text(U(1,i),U(2,i),sprintf(['u' num2str(i)
'\n']),'Color','black','FontWeight','bold','FontSize',12);
    end
end
for j = 1 : n
    line(V(1,j),V(2,j),'Marker','o','MarkerFaceColor','g');
    if m >= 20 || n >= 20
        text(V(1,j),V(2,j),sprintf(['v' num2str(j)
'\n']),'Color','black','FontWeight','bold','FontSize',8);
    elseif (m >= 10 && m < 20) || (n >= 10 && n < 20)
        text(V(1,j),V(2,j),sprintf(['v' num2str(j)
'\n']),'Color','black','FontWeight','bold','FontSize',10);
    else
        text(V(1,j),V(2,j),sprintf(['v' num2str(j)
'\n']),'Color','black','FontWeight','bold','FontSize',12);
    end
```

```
end
xmin = min(U(1,:))-0.5;
xmax = n+0.5;
ymin = min(U(2,:))-0.5;
ymax = max(U(2,:))+0.5;
set(handles.axes1,'XLim',[xmin xmax],'YLim',[ymin ymax]);
set(handles.pushbutton2,'UserData',{U,V});
set(handles.text5,'String',['L_' num2str(m) ',' num2str(n)]);


% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
set(handles.text7,'String','2m-2');
cla(handles.axes1,'reset');
set(handles.axes1,'XTick','','YTick','');

% Parameter
m = str2num(get(handles.edit1,'String'));
n = str2num(get(handles.edit2,'String'));

% Data titik
Titik = get(handles.pushbutton2,'UserData');
U = Titik{1};
V = Titik{2};

% Plot garis
hold on
for i = 1 : m-1
    for k = i+1 : m
        plot([U(1,i),U(1,k)],[U(2,i),U(2,k)],'k');
    end
end
plot([U(1,1),V(1,1)],[U(2,1),V(2,1)],'k');
for j = 1 : n-1
    plot([V(1,j),V(1,j+1)],[V(2,j),V(2,j+1)],'k');
end

% Plot titik
for i = 1 : m
    line(U(1,i),U(2,i),'Marker','o','MarkerFaceColor','b');
    if m >= 20 || n >= 20
        text(U(1,i),U(2,i),sprintf(['u' num2str(i)
'\n']),'Color','black','FontWeight','bold','FontSize',8);
    elseif (m >= 10 && m < 20) || (n >= 10 && n < 20)
        text(U(1,i),U(2,i),sprintf(['u' num2str(i)
'\n']),'Color','black','FontWeight','bold','FontSize',10);
    else
        text(U(1,i),U(2,i),sprintf(['u' num2str(i)
'\n']),'Color','black','FontWeight','bold','FontSize',12);
    end
```

```
end
for j = 1 : n
    line(V(1,j),V(2,j),'Marker','o','MarkerFaceColor','g');
    if m >= 20 || n >= 20
        text(V(1,j),V(2,j),sprintf(['v' num2str(j)
'\n']),'Color','black','FontWeight','bold','FontSize',8);
    elseif (m >= 10 && m < 20) || (n >= 10 && n < 20)
        text(V(1,j),V(2,j),sprintf(['v' num2str(j)
'\n']),'Color','black','FontWeight','bold','FontSize',10);
    else
        text(V(1,j),V(2,j),sprintf(['v' num2str(j)
'\n']),'Color','black','FontWeight','bold','FontSize',12);
    end
end
xmin = min(U(1,:))-0.5;
xmax = n+0.5;
ymin = min(U(2,:))-0.5;
ymax = max(U(2,:))+0.5;
set(handles.axes1,'XLim',[xmin xmax],'YLim',[ymin ymax]);

% Label
for i = 1 : m
    text(U(1,i),U(2,i),sprintf(['\n' num2str(2*m-
2*i)]),'Color','red','FontWeight','bold','FontSize',12);
end
for j = 1 : n
    if j == 1
        text(V(1,j),V(2,j),sprintf(['\n'
num2str(1)]),'Color','red','FontWeight','bold','FontSize',12);
    elseif j == 2
        text(V(1,j),V(2,j),sprintf(['\n'
num2str(3)]),'Color','red','FontWeight','bold','FontSize',12);
    else
        text(V(1,j),V(2,j),sprintf(['\n' num2str(2*(mod(j-
3,m)))]),'Color','red','FontWeight','bold','FontSize',12);
    end
end

set(handles.text7,'String',['2m-2 = ' num2str(2*m-2)]);


% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
movegui(gcf,'center');
set(gcf,'Name','Pelabelan L(2,1) Graf Lollipop');
set(handles.edit1,'String','');
set(handles.edit2,'String','');
set(handles.text5,'String','L_m,n');
set(handles.text7,'String','2m-2');
cla(handles.axes1,'reset');
set(handles.axes1,'XTick','','YTick','');
```

```
function edit2_Callback(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit2 as text
%        str2double(get(hObject,'String')) returns contents of
edit2 as a double


% --- Executes during object creation, after setting all
properties.
function edit2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
%        str2double(get(hObject,'String')) returns contents of
edit1 as a double


% --- Executes during object creation, after setting all
properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```