

Penerapan Algoritma Runut-Balik dalam Menggambar *Mind Map*

Rinda Nur Hafizha - 13516151

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

rindanurh@gmail.com

Abstrak—Salah satu metode yang dapat memudahkan proses belajar adalah dengan cara membuat *Mind Map* ketika mencatat, karena *Mind Map* selain berisi tulisan, terdapat gambar, variasi warna, serta keterhubungan antar point yang dapat memudahkan kita untuk memahami apa yang kita pelajari. Untuk membuat *Mind Map*, penentuan warna pada tiap point pembahasan merupakan salah satu factor yang penting agar kita dapat dengan mudah mengidentifikasi satu topik dengan topik lainnya. Namun terkadang warna yang tersedia terbatas sehingga penentuan warna pada tiap point pembahasannya tidak bisa menggunakan warna yang benar-benar berbeda, yaitu harus ada warna yang digunakan lebih dari sekali. Masalah ini dapat diselesaikan dengan menggunakan algoritma runut-balik dengan kasus pewarnaan graf. Hasil yang didapatkan dijamin optimal, yaitu menggunakan warna seminimal mungkin.

Keywords—Runut-balik, warna, graf

I. PENDAHULUAN

Sepanjang hidupnya, manusia dituntut untuk selalu belajar. Belajar dalam arti sebenarnya adalah sesuatu yang berlangsung sepanjang kehidupan seseorang. Dalam kenyataannya di kehidupan sehari-hari, sudah dapat dilihat bahwa pada hakikatnya manusia belajar terus menerus meskipun dengan cara yang berbeda dan melalui proses yang tidak sama. Dorongan belajar tersebut terjadi karena dirasa sebagai kebutuhan. Setiap orang merasa butuh untuk mempertahankan hidup dan kehidupannya dalam menghadapi dorongan-dorongan dari dalam dan tantangan alam sekitar yang selalu berubah. Mereka dituntut untuk mampu menyesuaikan diri secara aktif, dinamis, kreatif, dan inovatif terhadap diri dan kemajuan zaman.

Tidak ada batasan usia bagi manusia yang ingin belajar. Tidak terkecuali seorang mahasiswa yang sedang meniti pendidikan di perguruan tinggi. Belajar merupakan kewajiban bagi mahasiswa, bahkan harus dilakukan dengan baik dan tekun. Selain itu, agar bisa mendapatkan ilmu secara maksimal diperlukan strategi belajar.

Agar pembelajaran dapat dilakukan dengan baik, diperlukan metoda dan alat pembelajaran yang tepat. Salah satu alat yang sangat berguna dalam proses pembelajaran adalah dengan menggunakan *Mind Map* (Peta Pikiran). *Mind Map* adalah alat yang berguna untuk memetakan pikiran untuk menghubungkan konsep-konsep permasalahan tertentu dari cabang-cabang membentuk korelasi konsep menuju pada suatu

pemahaman dan hasilnya dituangkan langsung di atas kertas dengan animasi yang disukai dan mudah dimengerti oleh pembuatnya. Terlebih lagi untuk orang yang cara belajarnya secara visual, alat ini dapat membantu proses pembelajaran menjadi lebih mudah dan lebih baik.

Dalam membuat *Mind Map* atau dapat disebut juga saat melakukan *Mind Mapping*, kita menuangkan hasil pemetaan dari pemikiran dalam bentuk akar beserta cabang-cabang. Selain penggambaran pikiran yang tepat, dibutuhkan kreativitas yang melibatkan kombinasi unik antara citra, warna, dan pengaturan visual-spasial.

Untuk masalah pewarnaan *Mind Map*, dibutuhkan pengaturan sedemikian rupa sehingga *Mind Map* dapat terlihat jelas dan menarik. Salah satu pengaturan yang harus dilakukan adalah pewarnaan pada tiap cabangnya. Agar dapat dipahami dengan jelas, pewarnaan pada tiap cabang harus berbeda dengan cabang yang berhubungan langsung dengannya agar jelas bahwa satu cabang berbeda topiknya dengan cabang yang lain. Salah satu cara pewarnaan yang mudah adalah tiap cabangnya dibedakan warnanya. Namun, terkadang karena banyaknya topik yang ada pada *Mind Map*, dibutuhkan warna yang banyak, sedangkan warna yang tersedia sedikit.

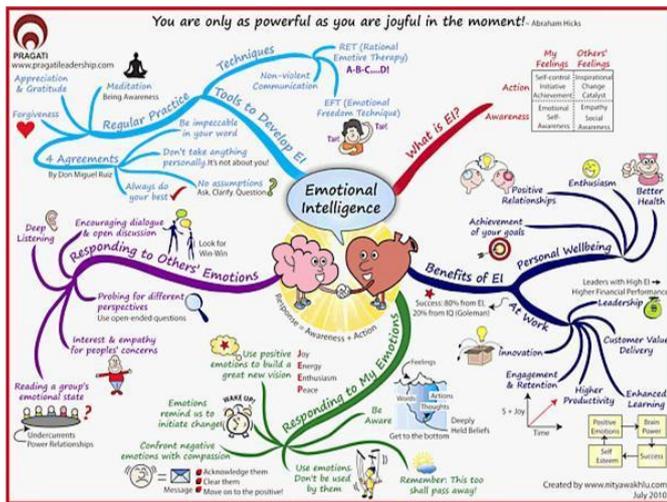
Agar proses pewarnaan dapat dilakukan dengan tepat, penentuan warna tiap cabangnya dapat ditentukan dengan bantuan program yang merupakan aplikasi dari kasus permasalahan pewarnaan graf yang menggunakan algoritma *backtracking* (runut-balik) yang akan dijelaskan di dalam makalah ini. Jenis pewarnaan yang akan dilakukan adalah pewarnaan simpul.

II. DASAR TEORI

A. *Mind Map*

Mind Map atau Peta Pikiran adalah diagram yang digunakan untuk mewakili kata-kata, ide, tugas, atau barang lainnya yang terkait dengan kata kunci sentral atau ide. Secara harfiah, peta pikiran merupakan 'peta' dari pikiran kita, yang menggunakan asosiasi, koneksi, dan memicu untuk merangsang lebih lanjut ide-ide. *Mind Map* digunakan untuk menghasilkan, memvisualisasikan, menstruktur, mengklasifikasikan ide, dan sebagai bantuan untuk belajar dan mengatur informasi, memecahkan masalah, mengambil keputusan, dan menulis. Selain meningkatkan keterampilan

belajar, juga dapat meningkatkan keterampilan hidup. Dalam Mind Map, informasi disusun dengan cara yang mencerminkan tepat bagaimana fungsi otak – memungkinkan otak menggunakan semua gambar dan asosiasinya dalam pola radial dan jaringan sebagaimana otak dirancang, bukan secara linier.



Gambar 1 – Contoh Mind Map

(<https://selvialdriani.wordpress.com/2012/07/23/mind-mapping-pentingnya-mind-mapping-dalam-kehidupan-kita-semua/#jp-carousel-137>)

Mind Mapping adalah sebuah cara untuk mencatat ide maupun gagasan pikiran. Sistem ini jauh lebih efektif dibandingkan bila kita belajar dengan cara mencatat materi dengan sistem linear, yaitu runtut ke bawah menggunakan urutan nomor. Hal ini terjadi karena bila hanya secara linear, kita hanya menggunakan otak kiri. Dengan mind mapping, otak kanan dilibatkan juga, yaitu kreativitas, imajinasi, dan visualisasi sehingga dapat menstimulasi otak kanan dan otak kiri secara sinergis.

Penelitian menunjukkan bahwa otak suka bekerja berdasarkan asosiasi dan akan menghubungkan setiap ide, memori, atau bagian dari informasi kepada puluhan, ratusan, bahkan ribuan ide-ide dan konsep lainnya.

Karena cara kerjanya yang sesuai dengan cara kerja otak manusia, mencatat dengan sistem mind mapping membuat kita mengingat materi lebih lama karena kita tahu gambaran besarnya dan memudahkan untuk memahami konsep sehingga materi sulit yang sedang kita pelajari dapat lebih mudah untuk dipahami dan dikuasai. Metode ini juga dapat meningkatkan kemampuan kita untuk mengingat sesuatu menjadi lebih utuh dan detail. Hal ini terjadi karena dengan mind mapping, kita mengumpulkan semua hal yang penting dalam topik tersebut. Selain itu, metode ini dapat meningkatkan kemampuan manajemen informasi, konsentrasi, imajinasi, dan memori karena dengan membiasakan diri mencatat dengan sistem ini akan melatih otak kita untuk menyusun informasi dengan baik. Metode ini juga dapat menjadikan kegiatan belajar menjadi lebih menyenangkan dan juga menghemat waktu, karena kita

dapat dengan bebas mengekspresikan apa yang ada di pemahaman kita dalam sebuah gambar dan coretan yang menggunakan kombinasi warna.

Beberapa hal penting dalam membuat peta pikiran atau melakukan Mind Mapping adalah:

1. Pastikan tema utama terletak di tengah-tengah
2. Dari tema utama, akan muncul tema-tema turunan yang masih berkaitan dengan tema utama
3. Cari hubungan antara setiap tema dan tandai dengan garis, warna, atau simbol. Pola-pola hubungan ini akan membantu kita memahami topik yang sedang kita baca. Peta pikiran yang telah dimodifikasi dengan simbol dan lambing yang sesuai dengan selera kita akan jauh lebih bermakna dan menarik dibandingkan peta pikiran yang miskin warna sehingga kita dapat lebih mudah mengingat.

B. Algoritma Backtracking (Runut-Balik)

Runut-balik (*backtracking*) adalah algoritma yang berbasis pada DFS untuk mencari solusi persoalan secara lebih mangkus. Runut-balik, yang merupakan perbaikan dari algoritma *brute-force*, secara sistematis mencari solusi persoalan di antara semua kemungkinan solusi yang ada. Dengan metode ini, kita tidak perlu memeriksa semua kemungkinan solusi yang ada. Hanya pencarian yang mengarah ke solusi saja yang selalu dipertimbangkan. Akibatnya, waktu pencarian dapat dihemat. Runut-balik lebih alami dinyatakan dalam algoritma rekursif. Kadang-kadang disebutkan pula bahwa runut-balik merupakan bentuk tipikal dari algoritma rekursif.

Untuk menerapkan metode runut-balik, properti berikut didefinisikan:

1. Solusi persoalan,

Solusi dinyatakan sebagai vector dengan n-tuple:

$$X = (x_1, x_2, \dots, x_n). x_i \in \text{himpunan berhingga } S_i. \text{ Mungkin saja } S_1 = S_2 = \dots = S_n$$

2. Fungsi pembangkit nilai x_k

Dinyatakan sebagai:

$$T(k)$$

$T(k)$ membangkitkan nilai untuk x_k , yang merupakan komponen vektor solusi.

3. Fungsi pembatas

Dinyatakan sebagai

$$B(x_1, x_2, \dots, x_k)$$

Fungsi pembatas menentukan apakah (x_1, x_2, \dots, x_k) mengarah ke solusi. Jika ya, maka pembangkitan nilai untuk x_{k+1} dilanjutkan, tetapi jika tidak, maka (x_1, x_2, \dots, x_k) dibuang dan tidak dipertimbangkan lagi dalam pencarian solusi.

Langkah-langkah pencarian solusi dengan metode runut-balik pada pohon ruang status yang dibangun secara dinamis adalah sebagai berikut:

1. Solusi dicari dengan membentuk lintasan dari akar ke daun. Aturan pembentukan yang dipakai adalah mengikuti metode pencarian mendalam (DFS). Simpul-simpul yang sudah dilahirkan dinamakan **simpul hidup** (*live node*). Simpul hidup yang sedang diperluas dimakan **simpul-E** (*Expand-node*). Simpul dinomori dari atas ke bawah sesuai dengan urutan kelahirannya.
2. Tiap kali simpul-E diperluas, lintasan yang dibangun olehnya bertambah panjang. Jika lintasan yang sedang dibentuk tidak mengarah ke solusi, maka simpul-E tersebut “dibunuh” sehingga menjadi **simpul mati** (*dead node*). Fungsi yang digunakan untuk membunuh simpul-E adalah dengan menerapkan **fungsi pembatas** (*bounding function*). Simpul yang sudah mati tidak akan pernah diperluas lagi.
3. Jika pembentukan lintasan berakhir dengan simpul mati, maka proses pencarian diteruskan dengan membangkitkan simpul anak yang lainnya. Bila tidak ada lagi simpul anak yang dapat dibangkitkan, maka pencarian solusi dilanjutkan dengan melakukan runut-balik ke simpul hidup terdekat (simpul orangtua). Selanjutnya simpul ini menjadi simpul-E yang baru. Lintasan baru dibangun kembali sampai lintasan tersebut membentuk solusi.
4. Pencarian dihentikan bila kita telah menemukan solusi atau tidak ada lagi simpul hidup untuk runut-balik.

Dibawah ini merupakan skema umum algoritma runut-balik dalam dua versi, yaitu versi rekursif dan versi iteratif. Skema dalam versi rekursif lebih tepat karena algoritma runut-balik lebih alami dinyatakan dalam bentuk rekursi.

a. Versi rekursif

```

procedure RunutBalikR(input k:integer)
{mencari semua solusi persoalan dengan
metode runut-balik; skema rekursif

Masukan: k, yaitu indeks komponen
vektor solusi, x[k]

Keluaran: solusi x = (x[1],x[2],...,x[n])
}

Algoritma:

  for tiap x[k] yang belum dicoba
  sedemikian sehingga (x[k] ← T(k)) and
  B(x[1],x[2],...,x[k]) = true do

    if (x[1],x[2],...,x[k]) adalah
    lintasan dari akar ke daun then

      CetakSolusi(x)
  
```

```

  endif

  RunutBalikR(k+1)

endfor

```

Pemanggilan prosedur pertama kali: RunutBalikR(1)

b. Versi iteratif

```

procedure RunutBalikI(input n:integer)
{mencari semua solusi persoalan dengan
metode runut-balik; skema iteratif

Masukan: n, yaitu panjang vektor solusi

Keluaran: solusi x = (x[1],x[2],...,x[n])
}

Deklarasi:

  k : integer

Algoritma:

  k ← 1

  while k>0 do

    if (x[k] belum dicoba sedemikian
    sehingga x[k] ← T(k)) and
    (B(x[1],x[2],...,x[k]) = true) then

      if(x[1],x[2],...,x[k]) adalah
      lintasan dari akar ke daun then

        CetakSolusi(x)

      endif

      k ← k+1 {indeks anggota
      tuple berikutnya}

    else

      k ← k-1 {runut-balik ke
      anggota tuple sebelumnya}

    endif

  endwhile {k=0}

```

Pemanggilan prosedur pertama kali: RunutBalikI(n)

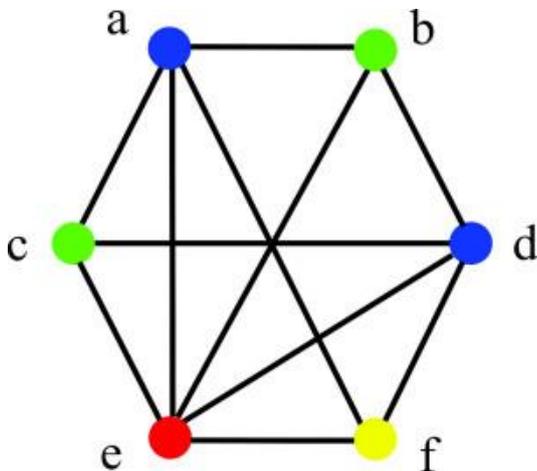
Setiap simpul dalam pohon ruang status berasosiasi dengan sebuah pemanggilan rekursif. Jika jumlah simpul dalam pohon ruang status adalah 2^n atau $n!$, maka untuk kasus terburuk, algoritma runut-balik membutuhkan waktu dalam $O(p(n) 2^n)$ atau $O(q(n) n!)$, dengan $p(n)$ dan $q(n)$ adalah polinom derajat n yang menyatakan waktu komputasi setiap simpul.

C. Pewarnaan Graf (Graph Colouring)

Ada tiga macam pewarnaan graf, yaitu pewarnaan simpul, pewarnaan sisi, dan pewarnaan wilayah (*region*). Masalah yang ingin diselesaikan pada makalah ini adalah pewarnaan simpul.

Pewarnaan simpul adalah memberi warna pada simpul-simpul suatu graf sedemikian sehingga tidak ada dua simpul bertetangga mempunyai warna yang sama.

Kita dapat memberikan sembarang warna pada simpul-simpul asalkan berbeda dengan simpul-simpul tetangganya.



Gambar 2 – Contoh Mind Map

(<https://potretsukasuka.wordpress.com/2016/01/19/pewarnaan-graf-dengan-jumlah-minimum-matematika-diskrit/>)

Dalam persoalan pewarnaan graf, kita tidak hanya sekedar mewarnai simpul-simpul dengan warna berbeda dari warna simpul tetangganya saja, namun kita juga menginginkan jumlah macam warna yang digunakan sesedikit mungkin. Pada Gambar 2, kita menggunakan 4 macam warna meskipun terdapat 6 simpul. Hal ini dapat dicapai dengan algoritma runut-balik.

Kita akan menyelesaikan masalah pewarnaan graf ini dengan algoritma runut-balik.

Misalkan warna dinyatakan dengan angka $1, 2, \dots, m$ dan solusi dinyatakan sebagai vektor X dengan n -tuple:

$$X = (x_1, x_2, \dots, x_n). \quad x_i \in \{1, 2, \dots, m\}$$

Warna simpul ke- k , x_k , ditentukan dengan algoritma berikut:

- (i) Bangkitkan warna i
- (ii) Periksa warna simpul-simpul yang bertetangga
- (iii) Jika warna i yang dibangkitkan tidak sama dengan warna simpul tetangga, maka simpul k diberi warna i , kalau tidak, bangkitkan warna berikutnya, ulangi langkah (ii) di atas
- (iv) Jika tidak ada lagi warna yang dapat dibangkitkan (warna habis), maka persoalan pewarnaan graf dengan n simpul dan m warna tidak dapat diselesaikan.

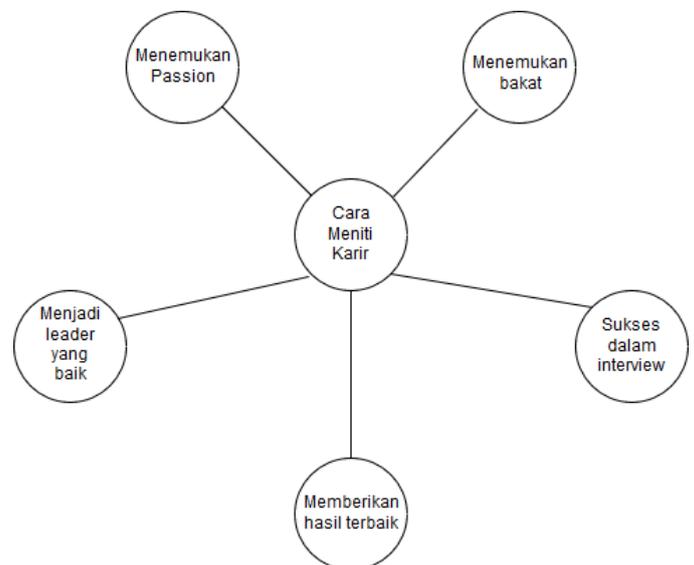
III. PENERAPAN ALGORITMA

Untuk permasalahan pewarnaan Mind Map, dapat kita terapkan persoalan pewarnaan graf menggunakan algoritma runut-balik. Bentuk mind map adalah graf radial di mana topik umumnya berada di tengah graf, dan sub topiknya terhubung ke tengah dan diletakkan melingkar di sekitar tengah.

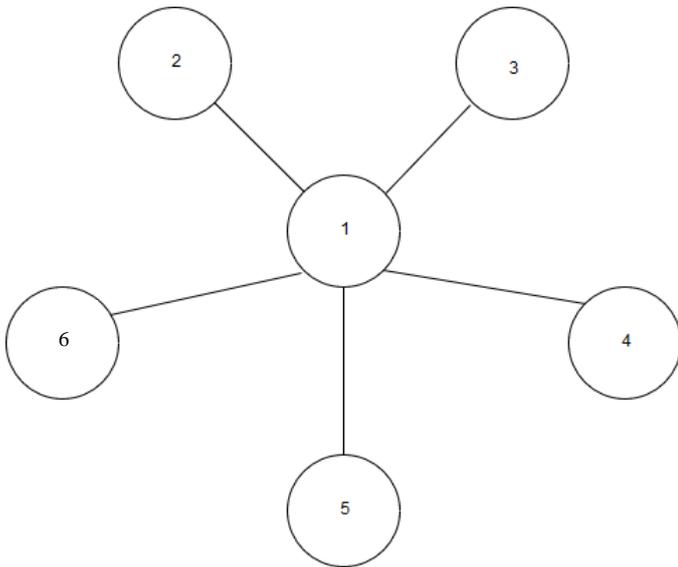
Contohnya adalah kita ingin membuat mind map sebuah buku tentang meniti karir yang berjudul Cara Meniti Karir. Buku tersebut memiliki 5 bab, yaitu

- Menemukan passion
- Menemukan bakat
- Sukses dalam Interview
- Memberikan hasil terbaik
- Menjadi leader yang baik

Maka topik di atas adalah Cara Meniti Karir, dan subtopiknya adalah kelima bab tersebut. Dalam menyelesaikan masalah pewarnaan mind map, mind map kita anggap sebagai graf dan tiap topik dan subtopiknya direpresentasikan sebagai simpul pada graf. Hasil penggambaran mind map diilustrasikan pada Gambar 3. Lalu untuk penentuan warna menggunakan algoritma yang akan diterapkan, mind map diubah di mana tiap topic dan sub topic direpresentasikan dengan simpul dan diberi nomor yang menandakan simpul ke- i , yang diilustrasikan pada Gambar 4.



Gambar 3 – Ilustrasi Mind Map pada contoh buku tentang Meniti Karir



Gambar 4 – Hasil Konversi Mind Map menjadi Graf bernomor

Pada persoalan tersebut, terdapat 6 simpul, dan misal kita hanya menyediakan dua warna. Misal mind map yang kita anggap sebagai graf direpresentasikan ke dalam matriks ketetanggaan. Maka,

Masukan:

1. Matriks ketetanggaan $GRAF[1..n,1..n]$ $\{n = \text{jumlah simpul graf; } n=6\}$
 $GRAF[i,j] = \text{true}$ jika ada sisi dari simpul i ke simpul j
 $GRAF[i,j] = \text{false}$ jika tidak ada sisi dari simpul i ke simpul j
2. Warna
 Dinyatakan dengan integer $1, 2, \dots, m$
 Di mana m dalam persoalan ini adalah 2

Keluaran:

1. Tabel $X[1..n]$, yang dalam hal ini, $x[i]$ adalah warna untuk simpul i .

Algoritma:

1. Inisialisasi $x[1..n]$ dengan 0
 $\text{for } i \leftarrow 1 \text{ to } n \text{ do}$
 $\quad x[i] \leftarrow 0$
 endfor
2. Panggil prosedur $\text{PewarnaanGraf}(1)$

{Kamus global}

Deklarasi

$\text{const } n = 6$ {jumlah simpul graf}

```

const m = 2 {jumlah warna yang
disediakan}

type matriks = array[1..n,1..n] of
boolean

type tabel = array[1..n] of integer

GRAF : matriks

x : tabel

```

```

procedure          PewarnaanGraf(input
k:integer)

{mencari semua solusi-solusi pewarnaan
graf; rekursif

Masukan: k adalah nomor simpul graf

Keluaran: jika solusi ditemukan, solusi
dicetak ke piranti keluaran

}

Deklarasi

stop : Boolean

Algoritma

stop <- false

while not stop do

    {tentukan semua nilai untuk x[k]}
    WarnaBerikutnya(k) {isi x[k]
dengan sebuah warna}

    if x[k]=0 then {tidak ada warna
lagi, habis}

        stop ← true

    else

        if k = n then {apakah
seluruh simpul sudah diwarnai?}

            CetakSolusi(X,n)

        else

            PewarnaanGraf(k+1)
{warnai simpul berikutnya}

        endif

    endif

endwhile

```

```

procedure WarnaBerikutnya (input
k:integer)
{menentukan warna untuk simpul k
Masukan: k
Keluaran: nilai untuk x[k]
K.awal: x[1],x[2],...,x[k-1] telah diisi
dengan warna dalam himpunan {1,2,...,m}
sehingga setiap simpul bertetangga
mempunyai warna berbeda-beda
K.akhir: x[k] berisi dengan warna
berikutnya apabila berbeda dengan warna
simpul-simpul tetangganya. Jika tidak
ada warna yang dapat digunakan, x[k]
diisi dengan nol
}

```

Deklarasi

stop, keluar : boolean

j : integer

Algoritma

```

stop <- false
while not stop do
    x[k] ← (x[k] + 1) mod (m+1)
    {warna berikutnya}
    if x[k]=0 then
        stop ← true
    else
        {periksa warna simpul-
simpul tetangganya}
        j ← 1
        keluar ← false
        while (j≤n) and (not
keluar) do
            if (GRAF[k,j]) and
(x[k] = x[j]) then {jika ada sisi dari
simpul k ke simpul j dan warna simpul k
= warna simpul j}
                keluar ← true
        {keluar dari kalang}
    else

```

```

j ← j+1
{periksa simpul berikutnya}
    endif
endwhile {j>n or keluar}

    if j = n+1 then {seluruh
simpul tetangga telah diperiksa dan
ternyata warnanya berbeda dengan x[k]}
        stop ← true {x[k]
sudah benar, keluar dari kalang}
    endif
endif
endwhile

```

IV. PEMBAHASAN

Ketika penentuan warna untuk graf mind map tersebut ditentukan oleh algoritma runut-balik, akan keluar hasil pada Gambar 5.

```

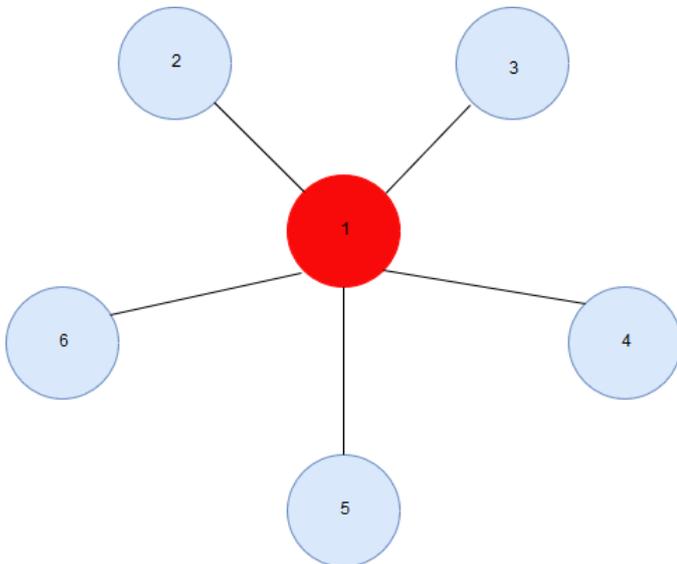
D:\rinda\SEMESTER 4\Strategi Algoritma\belajar uas>py stimamakalah.py
Masukkan jumlah warna:
2
Masukkan warna
1: merah
2: biru

simpul 1 berwarna merah
simpul 2 berwarna biru
simpul 3 berwarna biru
simpul 4 berwarna biru
simpul 5 berwarna biru
simpul 6 berwarna biru

```

Gambar 5 – Hasil Eksekusi Program

Permasalahan penentuan warna pada 6 simpul dengan 2 warna dapat diselesaikan, dengan simpul 1 berwarna merah dan simpul sisanya berwarna biru. Hal ini bisa terjadi karena simpul 1 bertetangga dengan seluruh sisa simpul, sedangkan seluruh simpul kecuali simpul 1 hanya bertetangga dengan simpul 1. Sehingga hasil pewarnaan mind map adalah sebagai berikut.



Gambar 6 – Hasil Pewarnaan Graf

Berdasarkan hasil pewarnaan, kurang tepat bila pewarnaan dilakukan menggunakan pewarnaan graf pada simpul, karena tujuan pewarnaan pada mind map adalah agar tiap topik yang berbeda dapat diidentifikasi dengan mudah dengan cara warnanya yang berbeda. Hal ini terjadi karena tetangga dari simpul 1 terletak melingkar disekitar simpul 1. Dapat dilihat pada hasil ilustrasi bahwa pada topik (simpul) yang berbeda namun letaknya bersebelahan, simpul berwarna sama sehingga hasil dapat salah diinterpretasikan bahwa simpul 2,3,4,5, dan 6 bukan merupakan topik yang berbeda, sehingga informasi yang ingin disampaikan tidak tersampaikan. Maka, untuk pewarnaan mind map sebaiknya menggunakan pewarnaan graf pada sisi, sehingga sisi yang bersebelahan dapat berbeda warnanya dan penyampaian informasi menjadi lebih tepat.

V. KESIMPULAN

Algoritma Runut-balik sangat baik untuk diaplikasikan pada persoalan untuk mencari solusi persoalan secara lebih mangkus. Selain itu, pada permasalahan pewarnaan pada simpul graf yang penyelesaiannya baik bila menggunakan algoritma Runut-balik, kita dapat menemukan solusi optimum, yaitu menggunakan warna sesedikit mungkin.

Namun, untuk permasalahan pewarnaan pada Mind Map, kurang tepat bila permasalahan ini diselesaikan seperti dengan kasus pewarnaan graf pada simpul.

UCAPAN TERIMA KASIH

Pertama-tama, penulis ingin mengucapkan terima kasih kepada Tuhan yang Maha Esa karena atas ridha dan rahmatnya penulis dapat menyelesaikan makalah ini meskipun saat mengerjakan makalah ini penulis memiliki kesulitan yaitu sedang sakit. Selain itu diucapkan juga terima kasih kepada orang tua penulis yang selalu memberikan dukungan dan doa. Tak lupa juga ucapan terimakasih kepada Bu Masayu selaku dosen pengajar mata kuliah Strategi Algoritma yang telah mengajarkan mata kuliah ini dengan sangat baik. Terakhir, penulis juga ingin mengucapkan terimakasih kepada teman, tim pengajar Mata Kuliah Strategi Algoritma, dan pihak-pihak lainnya yang telah membantu saya secara langsung maupun tidak langsung.

REFERENSI

- [1] Munir, Rinaldi. Diktat Kuliah IF2211 Strategi Algoritma. 2018. Bandung : Penerbit Informatika
- [2] Munir, Rinaldi. Diktat Kuliah IF2120 Matematika Diskrit. 2017. Bandung : Penerbit Informatika
- [3] <http://bacakilat.com/tips-mudah-membuat-mind-map/> [diakses 13 Mei 2018 pukul 11.20]
- [4] <http://mindmapclubindonesia.blogspot.co.id/2012/02/manfaat-mind-mapping-dalam-peningkatan.html> [diakses 13 Mei 2018 pukul 11.15]
- [5] <https://makalahkumakalahmu.wordpress.com/2008/10/01/konsep-belajar-sepanjang-hayat/> [diakses 13 Mei 2018 pukul 10.30]

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung 14 Mei 2018

Rinda Nur Hafizha
13516151