E-ISSN: 3031-8629
P-ISSN: 3032-3258
Halaman 1-10
Volume 2 Nomor 1 Tahun 2025

Algoritma Greedy Untuk Informasi Pasien Rawat Jalan Pada Klinik Dr.XYZ Cilangkap Jakarta Timur

Muhamad Irsan [1], Forkas T.S.B [2], Ahmad Husain[3]

[1]Program Studi Teknik Informatika, Universitas Indraprasta PGRI

e-mail: atstairway@gmail.com[1], fortysant.bb@gmail.com[2], hnr.husen@gmail.com[3]

Abstrak

Proses pengolahan data pasien, dokter dan obat yang bentrok sehingga dapat menyulitkan proses pasien dalam hal pengolahan data pasien dan obat, terutama pada penjadwalan pasien rawat jalan. Proses pasien berobat rawat jalan tidak terkontrol dan juga sering terjadinya kesalahan proses pembuatan laporan dalam perhitungan dan penjadwalan pasien maupun dokter yang masih kurang efektif. Hasil yang akan dicapai diharapkan dapat menemukan penjadwalan pasien rawat jalan yang terdata dan disimpan dengan meingmplementasikan Algoritma Greedy pada penjadwalan pasien rawat jalan.

Kata kunci: Algoritma Greedy,Pasien Rawat Jalan, Penjadwalan

Abstract

Processing patient, doctor and drug data is conflicting which can complicate patient processing in terms of processing patient and drug data, especially in scheduling outpatients. The process of patients seeking outpatient treatment is not controlled and there are also frequent errors in the reporting process in calculating and scheduling patients and doctors which are still less effective. The results that will be achieved are expected to be able to find outpatient scheduling that is recorded and stored by implementing the Greedy Algorithm in outpatient scheduling.

Keywords : *Greedy Algorithm, Outptient, Scheduling.*

LATAR BELAKANG

Dalam perkembangan teknologi manusia harus bisa beradaptasi untuk mengerjakan segala sesuatu pekerjaan dengan sistem terkomputerisasi. Setiap kebutuhan informasi pada saat ini semakin penting dalam berbagai semua kegiatan. Dengan berkembangnya teknologi yang saat ini semakin canggih dapat menghasilkan suatu informasi yang diperoleh akan sangat bermanfaat sekali untuk manusia dalam mengerjakan berbagai hal. Apabila pengolahan data tersebut menggunakan cara yang manual tentunya dapat menyebabkan beberapa kesalahan dan kurangnya efektifitas waktu dalam kegiatan manusia.

Klinik yang masih menggunakan sistem manual menggunakan kertas dalam pengolahan data, terutama pada pengolahan data pasien dan obat sehingga dapat menyulitkan proses berobat dalam hal pengolahan data pasien dan obat pada klinik tersebut.

Dari masalah di atasmaka perlu dibuat sebuah aplikasi yang akan digunakan sebagai proses penjadwalan pasien rawat jalan dengan menerapkan algoritma greedy.

METODE

Algoritma greedy adalah algoritma yang memecahkan masalah dengan tamak yakni mengambil solusi terbaik pada setiap iterasi dengan harapan bahwa solusi optimum lokal tersebut dapat menghasilkan solusi optimum global juga. Algoritma *greedy* cocok untuk digunakan pada persoalan-persoalan yang membutuhkan optimasi solusi yakni memaksimalkan ataupun meminimalkan sesuatu. Jika persoalan merupakan persoalan memaksimalkan, maka strategi algoritma greedy adalah memilih yang paling maksimum pada setiap iterasi. Sedangkan jika persoalan merupakan persoalan meminimalkan, maka strategi algoritma greedy yang diterapkan adalah memilih yang paling minimum pada setiap iterasi.

Dalam menyelesaikan suatu masalah dengan algoritma greedy, diperlukan pemetaan masalah tersebut menjadi elemenelemen algoritma greedy. Terdapat 6

elemen dalam algoritma greedy yang perlu dipetakan dari sebuah persoalan yakni himpunan kandidat, himpunan solusi, fungsi solusi, fungsi seleksi, fungsi kelayakan, dan fungsi objektif. Berikut merupakan penjelsan dari masing-masing elemen tersebut

1. Himpunan kandidat (C): berisi langkah-langkah yang mungkin untuk dipilih pada setiap iterasi.
2. Himpunan solusi (S): berisi langkah-langkah yang terpilih dari himpunan kandidat.
3. Fungsi solusi: fungsi untuk menentukan apakah langkah yang sudah terpilih dan dimasukkan ke himpunan solusi sudah memberikan hasil yang diinginkan.
4. Fungsi seleksi: fungsi yang menentukan langkah mana yang dipilih berdasarkan strategi greedy yakni memaksimalkan atau meminimalkan.
5. Fungsi kelayakan: Memeriksa apakah langkah yang terpilih oleh fungsi seleksi tidak melanggar constraint apa pun dan bisa dimasukkan ke himpunan solusi.
6. Fungsi objektif: memaksimalkan atau meminimalkan langkah yang dipilih.

Algoritma greedy berusaha untuk mencari optimum lokal pada setiap iterasi untuk mencapai optimum global. Namun algoritma greedy tidak dapat memastikan bahwa solusi yang didapat merupakan optimum global. Solusi yang diperoleh dapat berupa pseudo-optimum atau sub-optimum yakni optimum semu. Berikut merupakan alasan algoritma greedy tidak selalu menghasilkan solusi yang optimum.

1. Algoritma greedy tidak melakukan iterasi terhadap semua kemungkinan sepret yang dilakukan oleh algoritma brute force.
2. Algoritma greedy menggunakan fungsi seleksi yang berbeda-beda untuk setiap persoalan uang berbeda. Oleh karena itu perlu untuk menemukan fungsi seleksi yang tepat agar solusi persoalan tersebut menjadi optimal.

Karena kedua alasan tersebut, tidak semua solusi algoritma greedy dapat menghasilkan solusi yang optimal. Namun bukan berarti algorima greedy tidak dapat menghasilkan solusi yang optimal. Untuk beberapa persoalan, hasil dari solusi algoritma greedy dapat menghasilkan solusi yang optimal. Berikut beberapa contoh persoalan yang dapat diselesaikan menggunakan algoritma greedy beserta keoptimalan solusinya.

Tidak selalu Optimal:

1. Persoalan penukaran uang
2. Persoalan knapsack 1/0
3. Persoalan lintasan terpendek
4. Persoalan pecahan mesir
5. Persoalan TSP

Selalu Optimal:Persoalan activity selection

1. Persoalan meminimalkan waktu di sistem
2. Persoalan knapsack pecahan
3. Persoalan job dengan tenggat waktu
4. Persoalan pohon merentang minimum
5. Persoalan kode Huffman

Walaupun algoritma greedy tidak selalu menghasilkan solusi optimal, namun ada beberpa kondisi ketika algoritma ini baik untuk digunakan. Kondisi pertama tentunya ketika kita tahu bahwa solusi greedy tersebut merupakan solusi optimal. Kondisi kedua adalah ketika hasil eksak tidak diperlukan. Pada konsisi kedua, solusi dari algoritma greedy dapat dijadikan patokan atau hampiran dari solusi optimal.

HASIL DAN PEMBAHASAN

1. Pembahasan Algoritma

   Algoritma greedy merupakan salah satu algoritma untuk memecahkan persoalan optimasi. Persoalan optimasi yang dimaksud adalah persoalan mencari solusi optimum yaitu maksimasi (maximization) ataupun minimasi

(*minimization*). Algoritma greedy adalah algoritma yang memecahkan masalah langkah per langkah. Tahap Penyelesaian algoritma greedy menurut adalah sebagai berikut:

1. Mengambil pilihan yang terbaik yang dapat diperoleh pada saat itu tanpa memperhatikan konsekuensi ke depan.
2. Berharap bahwa dengan memilih optimum lokal pada setiap langkah akan mbahasanberakhir dengan optimum global.

Perhitungan banyaknya pasien. Berikut ini langkah–langkah dalam melakukan penghitungan shift:

1. Cek jumlah pasien dalam 1 bulan, contoh: bulan November 2024 terdapat 300 pasien.
2. Pada penjadwalan ini terdapat perbedaan banyaknya pasien pada hari minggu dan non-minggu, maka cek jumlah hari minggu dan non-minggu:
3. Hitung jumlah banyaknya pasien yang dibutuhkan selama 1 bulan.

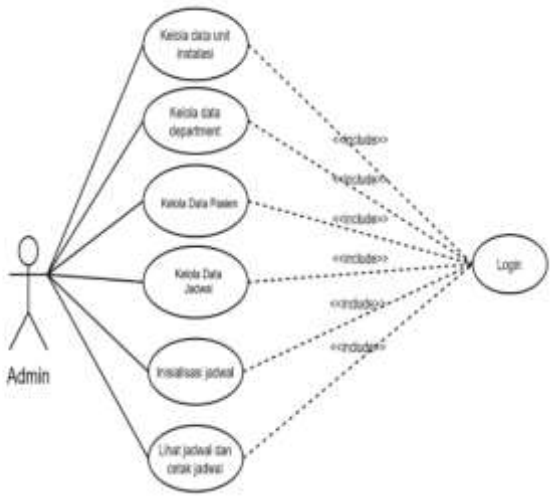
Penugasan penjadwalan pasien ini memiliki jadwal yang berbeda, yaitu pada hari minggu terdapat 15 pasien dan pada non–minggu terdapat 10 Pasien. Sehingga dibutuhkan proses random atau acak untuk pasien. Terdapat 4 orang pasien, C ={1, 2, 3, 4}. Sehingga kelompok–kelompok yang dihasilkan dari kedua proses acak dan merupakan solusi dari grpeeedy permutasi adalah sebagai berikut:

(S) = {(1,2) (3,4) (1,3) (2,4) (1,4) (2,3) (2,1) (3,2) (4,3) (4,1) (3,1) (4,2)} (S) = {(1,2,3) (4,1,2) (3,4,1) (2,3,4) (1,2,4) (3,1,2) (3,4,2) (1,3,4) (1,3,2)
(2,4,1) (3,1,4) (4,2,3) (1,4,2) (2,3,1) (3,2,4) (4,1,3) (1,4,3) (2,1,4)
(3,2,1) (2,4,3) (2,1,3) (4,2,1) (4,3,1) (4,3,2)}

Berikut merupakan langkah–langkah dalam proses penjadwalan:

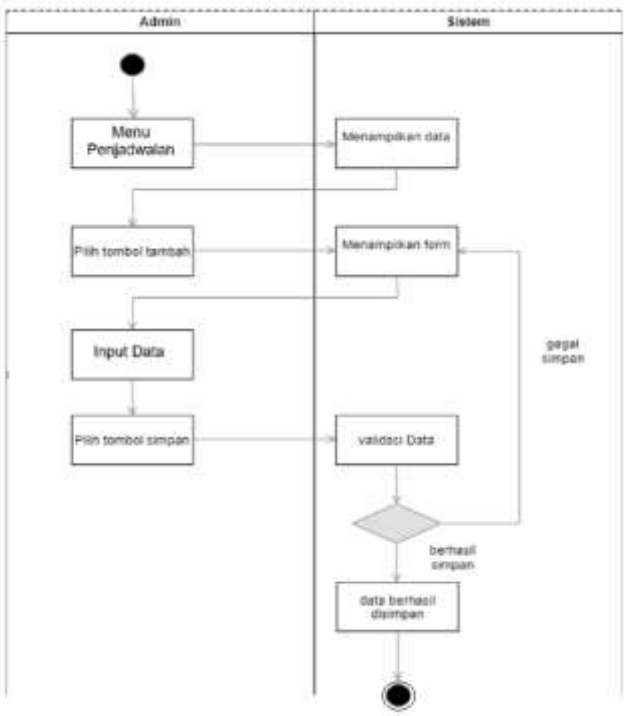
1. Cek tanggal penugasan untuk memperoleh jadwal yang masih kosong. Kemudian cek hari minggu dan non–minggu untuk menentukan penggunaan kelompok.
2. Untuk mengisi jadwal yang kosong digunakan proses greedy permutasi.
3. Ubah kedalam bentuk array of integer, sehingga solusi berupa kelompok yang terbentuk akan menjadi kandidat jadwal.
4. Untuk penjadwalan pada hari minggu digunakan hasil random
5. Untuk penjadwalan pada hari non–minggu, indeks solusinya adalah 26, karena jumlah hari non–minggu adalah 26 hari berarti 52 jadwal yang harus diisi.
6. Jika kandidat (C) telah kosong dan solusi belum terpenuhi dalam arti masih terdapat shift yang kosong, maka isi ulang (C) dan lanjutkan penelusuran dari penjadwalan awal.
7. Untuk memilih jadwal yang akan diisi dan masih kosong, berlaku aturan yang berfungsi sebagai uji kelayakan, yaitu: pasien tidak sedang off dan tidak memiliki jadwal dihari yang sama.

2. Use case Diagram



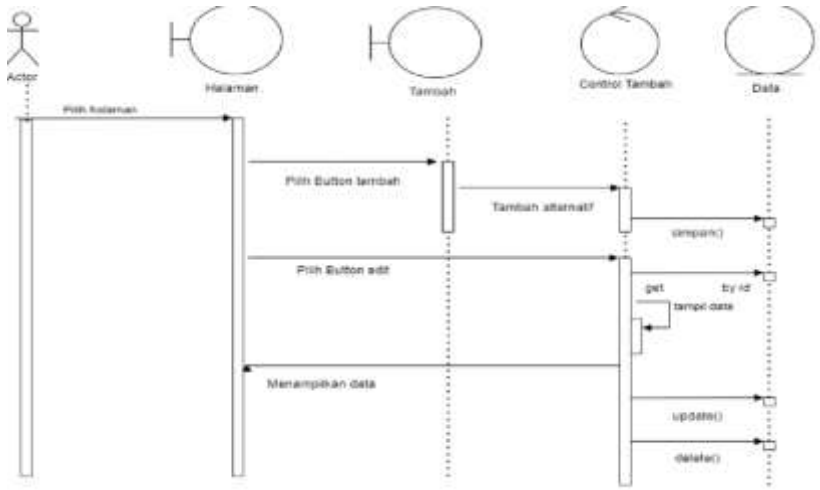
Gambar 1. Use Case Diagram

3. Activity Diagram Penjadwalan



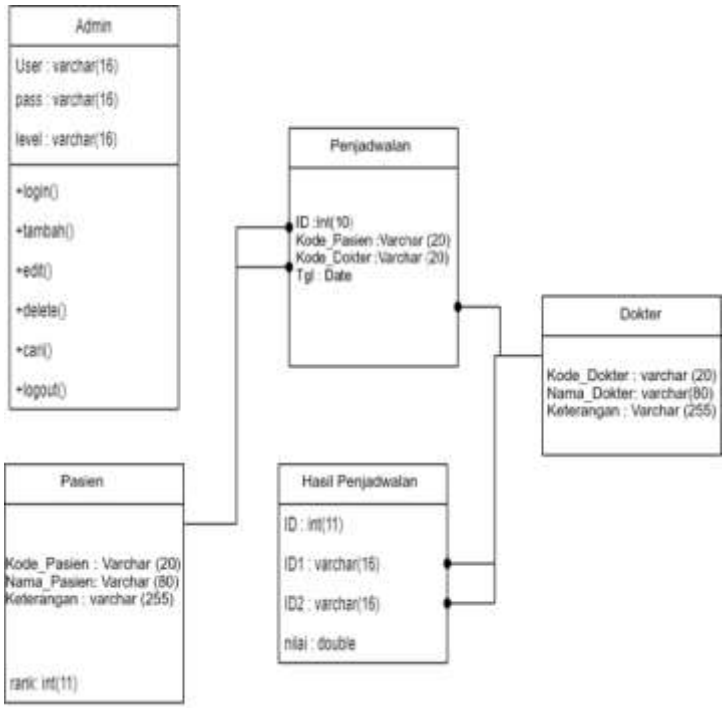
Gambar 2. Activity Diagram Penjadwalan

4. Sequence Diagram Penjadwalan



Gambar 3. Sequence Diagram Penjadwalan

5. Class Diagram Penjadwalan



Gambar 3. Sequence Diagram Penjadwalan

6. Tampilan Aplikasi Penjadwalan Pasien Menggunakan Algoritma Greedy



Gambar 4. Tampilan Aplikasi Penjadwalan Pasien

SIMPULAN

1. Untuk mengatasi permasalahan penjadwalan pasien rawat jalan yang bentrok, dibutuhkan sistem terkomputerisasi dengan menerapkan algoritma greedy.
2. Dengan menerapkan algoritma greedy pada Aplikasi dapat lebih meningkatkan mutu pelayanan kepada pasien dalam hal proses penjadwalan pasien rawat jalan.
3. Penghematan waktu dan tenaga dalam pengolahan data pada proses penjadwalan pasien rawat jalan, serta proses pencatatan data pasien dan dokter menjadi lebih cepat.
4. Dengan menerapkan algoritma greedy, pencarian data, informasi dan Laporan-laporan yang dibutuhkan dapat diterima dengan cepat dan rapi oleh bagian-bagian yang membutuhuhkan.

DAFTAR PUSTAKA

Ferbrian Jack, 2014, **Pengetahuan Komputer dan Teknologi Informasi**, Informatika, Bandung.

G. Sri Hartai, B. Herry Suharto, M. SoesiloWijono, 2013, **Pemrograman GUI Swing Java dengan Netbeans 5,** Andi, Yogyakarta.

Helmy, Ridwan Sanjaya, 2013, **Pengolahan Database SQL dengan Java**, Elex Media Komputindo, Jakarta.

Helmy, Ridwan Sanjaya, 2013, **Pengolahan Database SQL Server 2000 dengan Java2**, Elex Media Komputindo, Jakarta.

Jogianto, 2015.**Analisis dan Desain Sistem Informasi.** Yogyakarta : Graha Ilmu.

Kadir, 2013, **Sistem Informasi dengan Basis Data**, Elex Media Komputindo, Jakarta.

Kristanto, Andi. 2013. **Perancangan Sistem Informasi dan Aplikasinya.** Yogyakarta : Graha Ilmu.

Kristanto, Andi. 2013. **Perancangan Sistem Informasi dan Aplikasinya.** Yogyakarta : Graha Ilmu.

Kusrini, 2013, **Strategi Perancangan dan Pengelolaan Basis Data**, Andi, Yogyakarta.

Nugroho, Bunafit. 2014. **Database Relasional dengan *MySQL.*** *Yogyakarta: Andi Offset.*

Sigit W, Aloysius. 2015. ***7 Proyek aplikasi dengan Java.*** *Jakarta : Elex Media Komputindo.*

Simon, Judith, 2014.**Sistem Informasi Databases.** Yogyakarta : Graha Ilmu.

Suarga, 2017, **Dasar Pemrograman Komputer dalam Bahasa Java**, Andi, Yogyakarta.

Sutabri, Tata. 2014. **Analisa Sistem Informasi.** Yogyakarta : Graha Ilmu.

Sutanta, Edhy. 2014. **Sistem Database**. Yogyakarta : Graha Ilmu.