

Penerapan Algoritma Branch and Bound dalam Pengambilan Keputusan dengan Metode Analytic Hierarchy Process

Rizki Alif Salman Alfarisy / 13516005

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Bandung, Indonesia
13516005@std.stei.itb.ac.id

Abstraksi—Makalah ini akan membahas tentang penerapan Algoritma Branch and Bound dalam pengambilan keputusan tepatnya dengan metode Analytic Hierarchy Process untuk menentukan pilihan yang memiliki kriteria tertentu dan tiap kriteria memiliki prioritas terhadap tujuan utama. Algoritma Branch and Bound memungkinkan analisis dengan Analytic Hierarchy Process menjadi lebih efektif dengan hanya menganalisis kemungkinan-kemungkinan yang tepat saja dan mengabaikan kemungkinan yang tidak menuju ke tujuan utama.

Kata Kunci— *Analytical Hierarchy Process, Pengambilan Keputusan, Branch and Bound, Efektif dan Efisien.*

I. PENDAHULUAN

Dalam segala kegiatan dalam hidup seseorang pasti akan ditemukan masalah-masalah, baik yang sangat kecil seperti penghapus yang terjatuh dari meja maupun masalah yang sangat besar seperti kehilangan kartu ATM. Manusia pada dasarnya memiliki kemampuan dalam mengambil keputusan dengan cepat dan tepat untuk menyelesaikan persoalan yang dihadapinya. Misalnya pada contoh kasus sebelumnya, seseorang dapat dengan cepat memutuskan bahwa solusi dari permasalahan penghapus yang terjatuh adalah mengambilnya. Namun, semakin kompleks suatu permasalahan dan semakin banyak orang yang terlibat dalam permasalahan tersebut, maka akan semakin sulit pula bagi seseorang untuk menentukan solusi dari masalah tersebut. Misalnya pada kasus kehilangan ATM, seseorang mungkin akan bingung untuk menentukan apakah sebaiknya langsung memblokir kartu ATM-nya atau mencoba mencari kartunya ke pos satpam atau ke ATM terakhir yang dia kunjungi.

Oleh karena itu, mulai banyak orang yang mengemukakan metode-metode untuk mempermudah proses pengambilan keputusan dari suatu masalah yang sangat kompleks seperti *Mult-Voting, Affinity Diagrams, Conjoint Analysis, Cost-Benefit Analysis, Influence Diagrams, S.W.O.T (Strength-Weakness Opportunity Threat) Analysis, Analytic Hierarchy Process*, dan sebagainya. Teknik-teknik tersebut memiliki kelebihan dan kekurangan masing-masing dalam pengambilan keputusan, Tetapi teknik yang akan dibahas pada makalah ini adalah *Analytic Hierarchy Process (AHP)* karena teknik ini

memanfaatkan pohon dan pemberian penilaian pada simpul-simpul pohonnya sehingga menurut penulis, teknik pengambilan keputusan dengan *Analytic Hierarchy Process* dapat dioptimasi dengan menggunakan Algoritma *Branch and Bound*. Selain itu, Algoritma *Branch and Bound* dipilih karena sulitnya menentukan fungsi heuristic dalam proses pengambilan keputusan sehingga Algoritma seperti *Backtracking* dan *A-star* sulit digunakan.

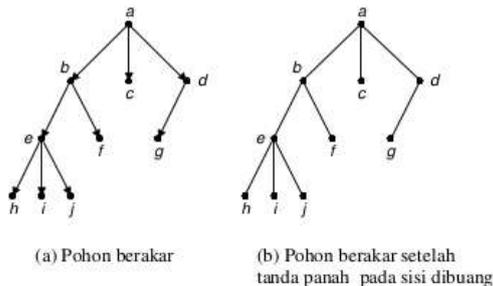
Dengan makalah ini, penulis berharap waktu yang dibutuhkan untuk melakukan *Analytic Hierarchy Process* dapat dipersingkat terutama apabila terdapat banyak kemungkinan keputusan sehingga ukuran *tree* yang akan dibuat sangatlah besar. Selain itu, cara ini juga dapat dilakukan apabila waktu yang tersedia sangatlah sedikit dan dibutuhkan pengambilan keputusan yang cepat tapi tepat.

II. DASAR TEORI

A. Pohon Berakar

Pohon adalah graf yang khusus. Misalkan $G = (V, E)$ adalah graf tak berarah-sederhana dan jumlah simpulnya n . Maka, syarat agar G merupakan suatu pohon adalah :

1. Setiap pasang simpul di dalam G terhubung dengan lintasan tunggal.
2. G terhubung dan memiliki $n-1$ buah sisi.
3. G tidak mengandung sirkuit.
4. Penambahan satu sisi pada graf hanya akan membuat satu sirkuit
5. Semua sisi graf adalah jembatan (jembatan adalah sisi yang bila dihapus menyebabkan graf terpecah menjadi 2 komponen)



Pada pohon berakar di atas:

- *a* merupakan akar
- *c, d, f, g, h, i, dan j* merupakan daun

Gambar 1 Contoh Pohon Berkar

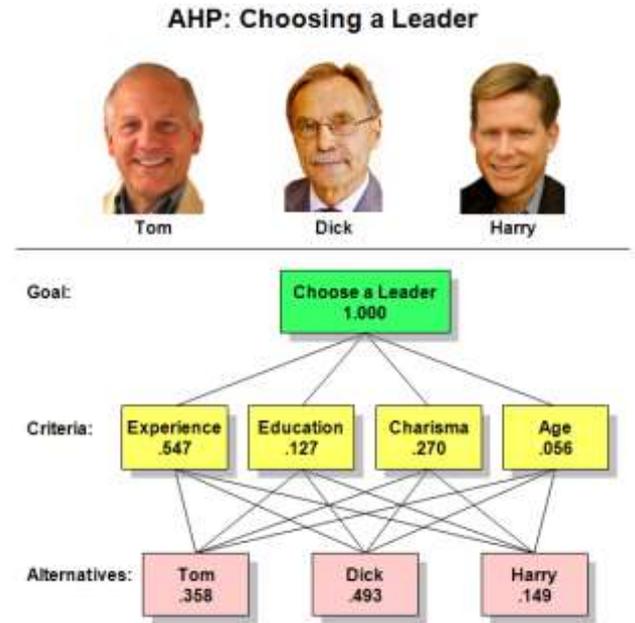
Sumber : <https://www.slideshare.net/septyndid/graf-pohon>

Pohon berakar adalah sebuah pohon yang simpulnya dieprlakukan sebagai akar dan sisi lainnya diberikan arah sehingga menjadi graf berarah. Berikut ini beberapa terminologi penting pada pohon berakar yang akan digunakan di makalah ini :

1. Akar, Simpul yang memiliki derajat 0 (permulaan dari pohon berakar)
2. Anak (*child*) – Orang tua (*Parent*), Misalkan T adalah pohon berakar, dan v adalah simpul pada T selain akar. Maka, orang tua dari v adalah simpul unik u dimana terdapat sisi dari u ke v. Apabila u adalah orang tua dari v maka v adalah anak dari u.
3. Daun (*Leaf*), Daun adalah suatu simpul dalam pohon berakar yang tidak memiliki anak.
4. Derajat (*Degree*), derajat simpul pada suatu pohon berakar ditentukan dari jumlah sisi yang keluar dari suatu simpul (jumlah anak)
5. Kedalam (*Depth*), Panjang lintasan maksimum dari akar ke simpul dari seluruh simpul di pohon.

B. Analytic Hierarchy Process

Analytic Hierarchy Process (AHP) adalah teknik untuk mengatur dan menganalisis keputusan kompleks secara matematis dan psikologis yang dikemukakan oleh Thomas L. Saaty pada tahun 1970. *Analytic Hierarchy* biasa digunakan dalam pengambilan keputusan grup besar dan dapat digunakan di berbagai bidang seperti pemerintahan, bisnis, industri, pendidikan dan kesehatan. Daripada menghasilkan keputusan yang tepat, teknik ini memiliki tujuan untuk menghasilkan keputusan sesuai dengan tujuan dari grup yang menggunakannya dan masalah yang sedang dihadapi oleh grup tersebut. Di dalam teknik ini disediakan cara untuk menstrukturkan masalah dan keputusan, memkuantifikasi elemen-elemennya, menghubungkannya dengan tujuan utama, dan terakhir memberikan evaluasi secara keseluruhan kepada solusi-solusi alternative yang ada.



Gambar 2 Contoh Diagram *Analytic Hierarchy Process* dalam pemilihan pemimpin organisasi

Sumber: https://en.wikipedia.org/wiki/Analytic_hierarchy_process#/media/File:AHP_TDHLeadImage.png

Untuk menggunakan teknik *Analytic Hierarchy Process*, pertama-tama penggunaanya harus mendekomposisi permasalahan yang dihadapinya menjadi suatu hierarki upamasalahan yang saling independen untuk dianalisis lebih jauh. Upamasalahan yang dibuat dapat berhubungan dengan aspek apapun dari masalah utama selama memengaruhi keputusan yang akan diambil grup tersebut. Setelah hierarki dibuat, pengguna baru akan mengevaluasi elemen-elemen yang telah didekompisiasi tadi satu per satu sambil memerhatikan elemen di atasnya dalam hierarki. Dalam melakukan evaluasi, pengguna dapat menggunakan fakta/data konkeret yang sudah ada atau melakukan penilaian tersendiri pada elemen berdasarkan kepentingan dari elemen tersebut menurut pengguna. Evaluasi tersebut kemudian akan diubah menjadi nilai-nilai numerik yang mungkin akan diberikan bobot juga berdasarkan prioritas dari pengguna.

Terakhir, nilai tiap elemen dihitung untuk setiap alternative solusi dan solusi dengan nilai terbaik dianggap sebagai keputusan terbaik yang bisa diambil untuk permasalahan yang sedang dihadapi grup tersebut. Teknik *Analytic Hierarchy Process* membutuhkan perhitungan yang tepat dan waktu yang lumayan lama sehingga biasanya memanfaatkan perangkat lunak untuk membantu analisisnya.

Penggunaan *Analytic Hierarchy Process* dapat dilakukan untuk pemngambilan keputusan oleh individu ataupun

kelompok, tetapi biasanya paling efektif digunakan untuk pengambilan keputusan oleh kelompok karena kemampuannya untuk mendekomposisi dan menilai satu per satu. Teknik ini biasanya digunakan untuk pengambilan keputusan dalam kasus-kasus berikut :

1. Pilihan – Memilih suatu alternatif dari banyak alternatif pilihan, terutama yang melibatkan banyak kriteria yang dinilai
2. Ranking – Mengurutkan kumpulan alternatif dari yang paling diinginkan ke yang tidak diinginkan
3. Prioritas – Menentukan keuntungan dari suatu alternatif dibandingkan kumpulan alternatif lainnya
4. Resource Allocation – Menempatkan *resource* dari beberapa kemungkinan alokasi
5. *Benchmarking* – Membandingkan proses dari suatu organisasi dengan organisasi teladan lainnya
6. *Quality Management* – Segala hal yang berhubungan dengan kualitas dan manajemen kualitas
7. *Conflict Resolution* – Menyelesaikan masalah dari beberapa pihak yang saling berposisi karena ketidakcocokan tujuan atau posisi.

Analytic Hierarchy Process terbukti berhasil membantu penyelesaian berbagai masalah terkait kasus-kasus di atas. Beberapa di antaranya diaplikasikan oleh berbagai organisasi di seluruh dunia, seperti berikut:

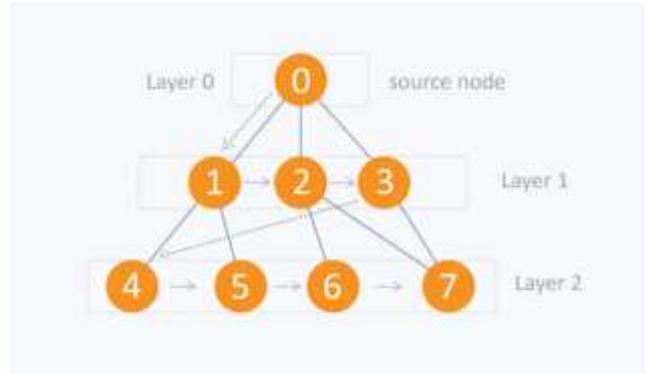
1. Pemilihan Reaktor Nuklir (*Politecnico de Milano*)
2. Menghitung kualitas sebuah perangkat lunak (*Microsoft Corporation*)
3. Menghitung risiko dalam memasang pipa minyak antar Negara (*American Society of Civil Engineer*)
4. Menentukan cara terbaik untuk mengurangi dampak perubahan iklim (*Fondazione Eni Enrico Mattei*)

C. Algoritma Breadth First Search (BFS)

Algoritma BFS adalah algoritma yang dilakukan pada proses traversal atau search pada suatu *graph*. sering dikatakan sebagai algoritma yang lebih ‘pasti’ atau ‘hati-hati’, hal ini dikarenakan BFS menelusuri graf dengan perlahan dari simpul awal. Algoritma BFS akan menelusuri seluruh simpul di kedalaman tertentu baru mulai menjelajahi simpul berikutnya.

Pada penelusuran graf dengan BFS, kita akan memulai dari suatu simpul di *v* dan menandai simpul itu sebagai simpul yang sudah dikunjungi. Saat itu, simpul *v* dapat dikatakan belum dijelajahi oleh algoritma. Simpul *v* baru akan dikatakan telah dijelajahi oleh algoritma apabila semua simpul yang bertetangga dengan *v* telah dikunjungi. Setelah itu, algoritma akan melakukan penjelajahan dengan mengunjungi seluruh simpul yang bertetangga dengan *v* yang belum dijelajahi dan menandai simpul *v* sebagai simpul yang telah dijelajahi. Simpul yang bertetangga dengan *v* akan dimasukkan ke daftar simpul-simpul yang belum dijelajahi. Kemudian, algoritma

akan melakukan penjelajahan ke seluruh simpul yang berada di daftar simpul yang belum dijelajahi dimulai dari simpul yang lebih dulu masuk ke daftar sampai seluruh simpul telah dijelajahi.



Gambar 3 Breadth First Search

Sumber :

<https://www.hackerearth.com/practice/algorithms/graphs/breadth-first-search/tutorial/>

D. Algoritma Branch and Bound

Algoritma Branch and Bound adalah algoritma yang digunakan untuk persoalan optimasi dimana nilai yang akan dioptimasi biasanya berupa bilangan-bilangan yang disimpan dalam struktur berbentuk graf/pohon. Algoritma ini terdiri dari beberapa komponen penting, yaitu :

1. Fungsi Objektif, fungsi yang menghasilkan nilai yang akan kita optimasi, misal pada persoalan menghitung jarak terpendek dari suatu titik, maka fungsi objektif adalah fungsi yang digunakan untuk menghitung jarak.
2. Fungsi Pembatas (*Constraint Function*), fungsi yang menentukan apakah suatu simpul harus dibatasi (*bound*) atau tidak. Beberapa kriteria fungsi pembatas yang umum adalah tidak lebih baik dari solusi terbaik sejauh ini, tidak feasible berdasarkan aturan yang ditentukan. Misalnya, pada persoalan Integer Knapsack, simpul yang menyebabkan tas kelebihan beban akan di-*bound*

Pada dasarnya, algoritma Branch and Bound mirip dengan algoritma BFS. Perbedaannya, algoritma BFS membangkitkan simpul berdasarkan FIFO sedangkan algoritma *Branch and Bound* membangkitkan simpul yang memiliki cost terkecil. Penghitungan dan pencarian cost mana yang terkecil diimplementasikan dengan bantuan fungsi objektif dan fungsi pembatas.

1. Inisialisasi akar, kemudian hitung fungsi objektif dari akar tersebut. Hasilnya akan menjadi solusi terbaik sejauh ini. Kemudian masukkan semua

simpul anak dari B ke Queue dan urutkan berdasarkan nilai fungsi objektifnya.

2. Ambil node dengan cost terkecil dari queue yang telah dibuat. Cek apakah node tersebut memenuhi fungsi pembatas
 - a. Jika memenuhi fungsi pembatas, apabila fungsi objektifnya lebih kecil dari solusi terbaik saat ini, maka ganti nilai solusi terbaik saat ini dengan fungsi objektif ini, kemudian bangkitkan anak-anaknya dan masukkan ke queue.
 - b. Jika tidak, maka *bound* simpul ini.
3. Ulangi hingga queue kosong. Nilai solusi terbaik adalah solusi terbaik dari permasalahan ini.

Algoritma ini selalu memberikan solusi optimum, tetapi tidak seefisien A* yang melakukan penghitungan cost dengan memanfaatkan fungsi heuristic pula. Selain itu, algoritma ini hanya bisa menyelesaikan masalah optimasi saja tidak seperti *backtracking*

III. PENERAPAN ALGORITMA

A. Deskripsi Permasalahan

Algoritma Branch and Bound akan diterapkan untuk mempercepat *Analytic Hierarchy Process*. Pada makalah ini, penulis akan menerapkan penerapan algoritma untuk *Analytic Hierarchy Process* dalam menentukan suatu pilihan berdasarkan multi kriteria.

Studi kasus yang dipilih adalah pengambilan keputusan untuk memilih sekolah dengan kriteria: indeks sekolah, jarak dari rumah, jumlah siswa yang diterima.

Tabel 1 Tabel penilaian sekolah

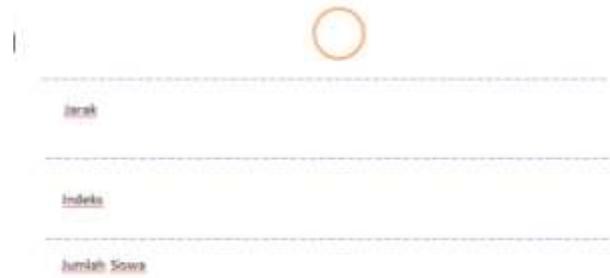
Sekolah	Jarak (50 %)	Jumlah Siswa (30 %)	Indeks (20%)
A	4	100	9
B	2	150	7
C	3	300	8

Tabel di atas merepresentasikan data yang akan dijadikan acuan untuk memilih sekolah berdasarkan kriteria-kriteria tertentu. Selain itu, kriteria juga sudah diberikan prioritas dalam pemilihan.

B. Penyelesaian masalah

Pertama-tama, buatlah graf untuk merepresentasikan permasalahan pengguna. Dalam kasus ini, graf dibuat dimana tiap aras merepresentasikan satu kriteria di mana aras ke 0

adalah akar, kemudian prioritas terbesar, kemudian prioritas kedua tersebut dan seterusnya. Pada kasus ini, level 0 adalah akar, level 1 adalah jarak, level 2 adalah indeks, level 3 adalah jumlah siswa.



Gambar 4 Rancangan Graf untuk Studi Kasus Memilih Sekolah

Setelah graf dibuat, tentukan fungsi pembatas dan fungsi objektif. Karena setiap aras memiliki kriteria berbeda, maka fungsi pembatasnya pun harus disesuaikan agar berbeda-beda setiap aras. Selain itu, fungsi pembatas juga harus dibuat agar skalanya sama misal dari skala 0-1 sehingga perbedaan antar aras tidak akan menyebabkan gap yang sangat besar.

Pada kasus ini, fungsi pembatas hanyalah cost lebih besar dari solusi minimum karena tidak ada permintaan khusus dari pengguna. Sedangkan fungsi objektifnya adalah:

$$\text{Untuk jarak, } f(x) = x/\max(\text{jarak})$$

$$\text{Untuk Siswa, } f(x) = f_{i-1}(x) + 1 - (x/\max(\text{siswa}))$$

$$\text{Untuk Indeks } f(x) = f_{i-1} + 1 - (x/10)$$

Pada kasus lainnya, fungsi objektif dapat dimodifikasi sesuai dengan keinginan pengguna.

Pada langkah pertama. Solusi maksimum diset negative tajam hingga lalu anak-anak dari simpul akar dibangkitkan yaitu jarak untuk tiap sekolah dan dihitung fungsi objektifnya. Kemudian dimasukkan ke dalam queue

Tabel 1 Tabel fungsi objektif jarak

Sekolah	f1(x)
A	1
B	0.5
C	0.75

Queue :

- Sekolah A : 1
- Sekolah B : 0.5
- Sekolah C : 0.75

Solusi Minimum : 0.5

Langkah 2, Dari seluruh fungsi objektif yang dibangkitkan node sekolah B lah yang memiliki cost minimum sehingga berikutnya node B yang dibangkitkan dihitunglah fungsi Objektif yang kedua yaitu, fungsi indeks. Dari hasil penghitungan, nilai yang didapat adalah $0.5 + 1-150/300$ sehingga hasilnya adalah 1. Kemudian node tersebut masuk ke dalam queue

Queue :

- Sekolah A : 1
- Sekolah B : 1
- Sekolah C : 0.75

Solusi Minimum : 0.75

Di langkah ketiga, kali ini sekolah C yang dipilih untuk dikerjakan karena costnya paling minimum yaitu 0.75. Anakanya dibangkitkan lalu dilakukan penghitungan untuk fungsi objektifnya yaitu $0.75 + 1- 300/300$ yang menghasilkan 0. Kemudian node tersebut masuk ke dalam queue kembali.

Queue :

- Sekolah A : 1
- Sekolah B : 1
- Sekolah C : 0.75

Solusi Minimum : 0.75

Pada langkah keempat, kali ini sekolah C kembali yang dipilih untuk dievaluasi karena costnya masih paling minimum yaitu 0.75. Anakanya dibangkitkan lalu dilakukan penghitungan fungsi objektif yaitu $0,75 + 1 - 8/10$ sehingga didapatkan hasil 0.95 dan dimasukkan ke dalam queue

Queue :

- Sekolah A : 1
- Sekolah B : 1
- Sekolah C : 0.95

Solusi Minimum : 0.95

Pada langkah kelima, sekolah yang memiliki cost terkecil tetap sekolah C dengan cost 0.95. Namun, sekolah C tidak memiliki anak lagi sehingga tidak ada simpul yang dibangkitkan.

Queue :

- Sekolah A : 1
- Sekolah B : 1

Solusi Minimum : 0.95

Pada langkah keenam, sekolah yang memiliki cost terkecil adalah sekolah A dan B dengan cost 1. Sekolah A dipilih (karena aturan FIFO) namun, karena nilai 1 lebih besar dari cost minimum maka simpul ini dibound.

Queue :

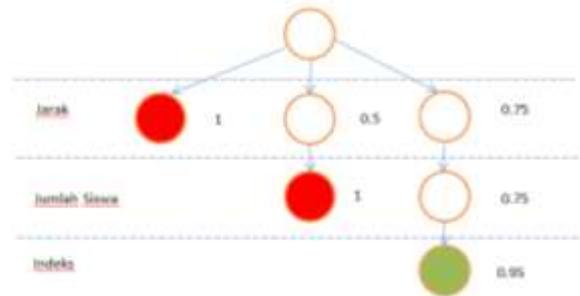
- Sekolah B : 1

Solusi Minimum : 0.95

Pada langkah ketujuh, sekolah yang tersisa hanyalah sekolah B dengan cost 1. Tetapi, simpul ini juga dibound karena costnya lebih besar dari cost minimum sekarang.

Queue :

Solusi Minimum : 0.95



Gambar 5 Hasil akhir dari pohon pemilihan sekolah

Karena Queue kosong, maka algoritma sudah selesai dan solusi minimum adalah 0.95 dari sekolah C sehingga sekolah yang sebaiknya pengguna datangi adalah sekolah C. Dapat dilihat dari hasil akhir graf bahwa dengan algoritma Branch and Bound tidak seluruh kemungkinan perlu dievaluasi. Pada sekolah A, yang indeks dan jumlah siswa tidak dihitung dan pada sekolah B hanya jarak dan indeks yang dihitung.

IV. KESIMPULAN

Algoritma Branch and Bound terbukti dapat mempercepat proses pengambilan keputusan dengan *Analytic Hierarchy Process*. Hal ini dapat dilakukan baik dengan menggunakan program atau tanpa program dengan keuntungan dan kekurangan tersendiri.

Apabila algoritma dilakukan dengan menggunakan program maka penentuan fungsi objektif akan lumayan sulit, dibutuhkan waktu untuk menentukan fungsi objektif yang tepat. Selain itu, semakin banyak kriteria yang terlibat semakin kompleks juga fungsi yang harus dibuat.

Sedangkan apabila algoritma dilakukan tanpa program atau langsung dikerjakan manusia, kekurangannya adalah evaluasi yang dilakukan berubah-ubah dengan cepat dan mungkin membuat bingung. Misalnya, pada saat sedang membahas tentang jarak tiba-tiba di langkah berikutnya harus membahas tentang indeks sekolah. Hal tersebut mungkin membuat beberapa orang bingung. Akan tetapi, kelebihanannya adalah evaluasi bisa

dilakukan dengan lebih natural tanpa rumus dan semacamnya asalkan pengguna dapat menyepakati system penilaian tertentu

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Tuhan Yang Maha Esa yang telah memampukan penulis untuk menyelesaikan makalah ini dengan baik. Selanjutnya penulis juga mengucapkan terima kasih kepada orang tua penulis serta rekan-rekan di sekitar penulis yang terus memberikan dukungan dan nasihat sehingga menjadi motivasi bagi penulis dalam menyelesaikan kewajiban dalam perkuliahan di ITB. Penulis berterima kasih pada Dr.Ir. Rinaldi Munir, MT ; Dr. Masayu Leylia Khodra, ST.MT ; Dr.Nur Ulfa Maulidevi, ST., M.Sc., sebagai dosen mata kuliah strategi algoritma yang telah membimbing penulis, dalam penulisan makalah ini melalui perkuliahan Strategi Algoritma. Penulis juga mengucapkan terima kasih kepada seluruh pihak yang berperan dalam penyusunan makalah ini sehingga penulis dapat menyelesaikan makalah ini tepat waktu.

[1] Munir, Rinaldi. 2006. Strategi Algoritma. Bandung : Penerbit Informatika

[2] Drake, P.R. (1998). "Using the Analytic Hierarchy Process in Engineering Education" (PDF). International Journal of Engineering Education. 14 (3): 191–196. Archived from the original (PDF) on 2007-11-28. Retrieved 2007-08-20..

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



ttd

Rizki Alif Salman Alfarisy - 13516005