

## Analisis Algoritma Definite Integration Menggunakan Metode Desk Check

**Inna Muthmainnah Dalimunthe**

Prodi Pendidikan Matematika Fakultas Ilmu Tarbiyah dan Keguruan  
UIN Sumatera Utara  
*muthmainnahdalimunthe@gmail.com*

**Yahfizham**

Prodi Pendidikan Matematika Fakultas Ilmu Tarbiyah dan Keguruan  
UIN Sumatera Utara  
*Yahfizham@uinsu.ac*

*Korespondensi penulis: muthmainnahdalimunthe@gmail.com*

**Abstrak.** Biasanya kita mengalami kesulitan dalam menentukan keluaran suatu algoritma pseudocode, terutama pada struktur kontrol bersarang atau nested loop. Biasanya rumit. Kita kesulitan menebak logika atau menentukan hasilnya. Misalnya, apa hasil akhir dari siklus tersebut? Apakah itu benar atau salah? Di manakah titik awal dan akhir perulangan? Di manakah batas badan lingkaran? Apakah siklus ini harus terus berlanjut atau dihentikan? Apakah perulangan berlanjut hingga hasil atau kondisi logisnya benar? Apakah perulangan berjalan sampai kondisinya benar? Selain itu, karena pikiran manusia mempunyai keterbatasan untuk mengingat atau mengulang, maka diperlukan suatu alat untuk menganalisisnya. Berdasarkan permasalahan tersebut penulis mencari suatu model atau metode analisis variabel secara sederhana baik variabel masukan maupun variabel keluaran dengan logika algoritmik dalam algoritma psuedo code. Hal inilah yang menjadi maksud penulis merancang sebuah tabel yang dinamakan tabel kontrol. Dalam hal ini, tugas kontrol desktop adalah mendokumentasikan riwayat beberapa perubahan dan logikanya dari berbagai variabel yang dianalisis, serta input dan output, dengan bantuan tabel. Tujuan dari penulisan ini adalah menganalisis respon perancangan Algoritma Integrasi Pasti dengan menggunakan metode flowchart dan memeriksa kemungkinan contoh permasalahan yaitu integrasi eksak.

**Kata Kunci :** Analisis Algoritma

**Abstract.** Normally we experience hardships in deciding the result of a pseudocode calculation, particularly in settled control structures or settled circles. Generally muddled. We experience issues speculating rationale or deciding the result. For instance, what is the final product of the cycle? Is that correct? Where are the beginning and end points of the circle? Where could the limit of the circle's body be? Should this cycle proceed or stop? Does the circle go on until the outcome or intelligent condition is valid? Does the circle run until the condition is valid? Aside from that, in light of the fact that the human psyche has impediments in recollecting or rehashing, a device is expected to examine it. In light of this issue, the creator is searching for a basic model or technique for variable examination, both info factors and result factors with algorithmic rationale in a pseudo code calculation. This is the creator's goal in planning a table called a control table. For this situation, the errand of work area control is to report the historical backdrop of a few changes and their rationale of the different investigated factors, as well as data sources and results, with the assistance of tables. The reason for this paper is to dissect the reaction to the plan of the Specific Reconciliation Calculation utilizing the flowchart technique and inspect potential instances of issues, in particular precise coordination.

**Keyword :** Algoritma Analysis

## PENDAHULUAN

Membuat suatu algoritma melibatkan beberapa langkah utama, termasuk: mendefinisikan masalah; membuat model permasalahan; merancang algoritma model; menguji kebenaran algoritma; diimplementasikan dalam bahasa pemrograman seperti C, Java,

dll; dokumentasi; dan analisis kompleksitas algoritma, seperti analisis hasil menggunakan tabel pencarian, kompleksitas ruang, dan kompleksitas waktu. Membuat algoritma Matematika memiliki banyak manfaat, antara lain: membuat atau menyusun perhitungan tidak bergantung pada bahasa pemrograman, hasil yang didistribusikan adalah setara dengan alasan bahwa perhitungannya adalah sesuatu yang serupa. Untuk individu tertentu, masalah logika atau masalah, khususnya analitik, sulit untuk dipahami, misalnya, barisan langkah, urutan yang masuk akal, arah yang independen, dan proses matematika. Oleh karena itu, penulis perlu mengajukan salah satu masalah matematika, yaitu *Conclusive Necessary*, yang akan digunakan sebagai ilustrasi masalah dalam makalah ini untuk membuat flowchart dan strategi pseudocode. Tujuannya adalah untuk berkonsentrasi pada masalah model dengan menggambarkan rangkaian proses penalaran, navigasi, dan penghitungan angka dengan cara yang mudah dengan bantuan gambar. Gambar-gambar ini adalah diagram alir, yang merupakan perangkat atau instrumen yang menunjukkan cara yang harus diambil dalam menangani masalah PC yang dikomunikasikan dalam perkembangan gambar realistis yang luar biasa. Membuat suatu algoritma melibatkan beberapa langkah utama, termasuk: mendefinisikan masalah; membangun model permasalahan; merancang algoritma model; menguji kebenaran algoritma; diterapkan dengan cara yang rumit dan memakan waktu.

Membuat kalkulasi memiliki banyak keuntungan, antara lain: membuat atau menyusun kalkulasi tidak bergantung pada bahasa pemrograman, artinya, membuat kalkulasi tidak bergantung pada bahasa pemrograman dan komputer yang mengeksekusinya; dokumentasi kalkulasi dapat dikonversi ke dalam beberapa dialek pemrograman; apa pun bahasa pemrogramannya, hasil kalkulasi yang didistribusikan akan sama karena kalkulasinya adalah sesuatu yang mirip. Menentukan keluaran suatu algoritma pseudocode biasanya sulit, terutama pada logika struktur kontrol pemutaran atau algoritma nested loop yang cenderung rumit sehingga sulit untuk menebak atau menentukan hasilnya. Misalnya, apa akhir dari perulangan? Dimanakah batas awal (mulai) dan batas akhir (berhenti) siklus tersebut? Di mana Anda memulai dan di mana Anda mencapai batas badan perulangan? Haruskah ia berbalik atau diam? Apakah hasil perulangan itu benar atau salah? Apakah ini perulangan hingga hasilnya boolean atau kondisinya ditafsirkan benar? Apakah itu berulang sampai kondisinya benar? Selain itu, karena keterbatasan pikiran manusia untuk mengingat atau melakukan hal-hal tersebut, maka kebutuhan akan suatu alat dalam analisis ini sangat diperlukan. Berdasarkan kondisi tugas tersebut, penulis mencari suatu model atau metode yang dapat dengan mudah menganalisis variabel-variabel baik input maupun output, serta logika algoritma pada

algoritma pseudocode. tabel yang penulis sebut sebagai tabel kendali. Dalam hal ini, tugas tabel kontrol tabel adalah mendokumentasikan histori setiap perubahan beberapa yang dianalisis, baik input maupun output, serta logikanya menggunakan tabel tersebut. Artikel ini juga merupakan artikel lanjutan dari artikel penulis sebelumnya yang berjudul “Rancangan Algoritma Operasional Perkalian Matriks Menggunakan Metode Flowchart”. Artinya tujuan artikel untuk menganalisis jawaban yang penulis jawab pada artikel tersebut dan pembuktian yang penulis gunakan sebagai contoh soalnya aljabar linier pada perkalian matriks.

## **TINJAUAN PUSTAKA**

### **1. Algoritma**

Nama algoritma ini diambil dari nama sarjana Arab Abu Ja'far Muhammad Ibnu Musa Al Khuwarizmi, yang menulis buku berjudul Al Jabar Wal Muqabala (Kitab Pembaruan dan Reduksi). Orang Barat membaca kata Al Khuwarizmi sebagai Algoritma, yang lambat laun menjadi Algoritma dan menyatu dengan Algoritma Indonesia. Algoritma dapat diartikan sebagai rangkaian langkah (instruksi/tindakan) yang terbatas untuk menyelesaikan suatu masalah. Istilah algoritma Donald E. Knuth, yaitu: Soal atau soal ilmiah khususnya kalkulus sulit dipahami sebagian orang, seperti langkah-langkah secara berurutan, urutan logis, pengambilan keputusan dan proses aritmatika. Penulis ingin mengemukakan salah satu permasalahan kalkulus yaitu Integral Definitif yang akan dijadikan contoh permasalahan dalam makalah untuk membuat flowchart dan pseudocode.

1. Definisi (kekuatan) langkah algoritma harus di amati dengan baik dan tidak boleh menimbulkan makna yang dabel
2. Masukan (masukan) Suatu algoritma mempunyai nol atau lebih masukan yang diberikan kepada algoritma tersebut sebelum dijalankan.
3. Keluar (Keluar) algoritma menghasilkan satu atau lebih.
4. Efisiensi (efisiensi) algoritma dijalankan waktu yang "masuk akal".

Algoritma mempunyai 3 struktur dasar, yaitu

### **1. Pesan**

Urutan adalah setidaknya satu perintah yang dieksekusi dalam permintaan di mana perintah tersebut disusun. Perintah tersebut menentukan hasil akhir dari

perhitungan. Dengan asumsi permintaan di mana mereka disusun berubah, produk akhir juga dapat berubah.

## 2. Pilih

Opsi adalah Arahan yang dijalankan pada kondisi tertentu. Pernyataan hanya akan dieksekusi jika kondisi benar, sebaliknya jika kondisi salah maka pernyataan tidak akan dieksekusi. Pernyataan If (if) dan Then (then) digunakan dalam pernyataan bersyarat.

## 3. Pengulangan

- a. Pengulangan adalah mengulangi operasi yang sama sebanyak yang ditentukan atau dalam kondisi yang diinginkan.

## **METODE PENELITIAN**

### 1. Flowchart

Menurut Edhy Sutanta (2004, hal. 28), diagram alur dicirikan sebagai perangkat atau instrumen yang menunjukkan langkah yang harus diambil dalam menangani masalah komputer dengan dinyatakan sebagai rangkaian simbol-simbol grafik tertentu. Keuntungan menggunakan flowchart untuk menyelesaikan masalah komputer:

1. Terbiasa berpikir sistem dan pengaturannya.
2. Agar lebih gampang memeriksa dan menemukan prosedur yang salah dan panjang
3. Prosedurnya gampang dikembangkan

Robertson, Lesiey Anne (2004:54), Ngoen, Thompshon Susabda (2009:127-148), Bahasa C memberikan tiga panduan untuk melakukan siklus berulang: untuk, sementara, dan lakukan sementara. Ketiga arahan ini memiliki kualitasnya masing-masing.

#### a. Instruksi For (arahan)

Arahan digunakan untuk memulai proses iterasi yang laju iterasinya diketahui sebelum proses iterasi dimulai. untuk ([ekspresi1]; [ekspresi2]; [ekspresi3]) ekspresi; Ekspresi1 untuk memberikan nilai sebuah counter. Ekspresi 2 adalah

ekspresi Boolean yang mengembalikan nilai true (bukan null) atau false (null) saat dijalankan. Expression3 adalah bantuan bekerkerja untuk file statement.

## 2. Struktur pengendalian reproduksi

### a. Selama mengajar

instruksi untuk memulai proses berulang, dimana kondisi pengulangan diperiksa pada awal pemrosesan. Pernyataan while biasanya digunakan untuk memulai siklus perulangan yang pengulangan redundansinya tidak jelas pada awal interaksi perulangan. sementara (artikulasi) artikulasi Artikulasi adalah artikulasi Boolean dan berjalan sebagai kontrol perulangan. Berapa lama pun konsekuensi dari penilaian artikulasi ini menghasilkan nilai yang tidak nol, artikulasi akan dieksekusi berulang-ulang.

### b. Lakukan Selama arahan

Perintah do while merupakan perintah untuk menjalankan proses berulang, dimana kondisi pengulangan diperiksa pada akhir proses. Pernyataan do while biasanya digunakan untuk memulai proses berulang yang frekuensi pengulangannya tidak diketahui tetapi harus dijalankan setidaknya sekali.

### c. mengirimkan pernyataan sementara (ekspresi)

Ekspresi berbentuk pernyataan atau beberapa pernyataan yang diapit oleh {}. Ekspresinya adalah Sebuah gerakan yang sengaja dikoordinasikan dan masuk akal untuk mengelola sebuah isu, atau bisa juga diartikan bahwa estimasi adalah sebuah gerakan penalaran yang menentukan yang secara sengaja dikoordinasikan dengan menggunakan bahasa yang sadar untuk mengelola sebuah isu.

## **PEMBAHASAN**

Algoritma merupakan akar dari sistem yang terbentuk dalam dunia pemrograman, yaitu susunan langkah-langkah penyelesaian tugas yang sistematis dan logis. Mengetahui struktur dasar algoritma pemrograman sangat berguna dalam pemrograman terutama bagi orang yang sering merencanakan atau menggunakan algoritma.dasar-dasar penulisan algoritma pemrograman ini harus diketahui dan juga dikuasai permasalahan. Struktur dasar

algoritma pemrograman sangat berguna dalam algoritma pemrograman. Algoritma berisi langkah untuk memecahkan suatu masalah. Langkah ini disebut struktur dasar. Strukturnya dibagi menjadi tiga bagian, yaitu. langkah-langkahnya bisa diproses satu per satu langsung ke bawah. Suatu struktur disebut struktur sekuensial atau dapat berpindah ke suatu fasa tertentu apabila memenuhi syarat tertentu atau disebut. struktur pemilih, yaitu counter yang diinginkan atau disebut sesuai dengan struktur loop.

Mendengar kata “algoritma” kita langsung teringat pada istilah matematika, yang tentunya ketika kita ingin menerapkan proses komputer dalam matematika. Namun nyatanya algoritmanya lebih penting atau bahkan berbeda. Pengertian Algoritma sendiri adalah suatu perkembangan langkah-langkah yang disusun secara sengaja dan masuk akal dalam menangani suatu masalah, atau dapat juga diartikan bahwa perhitungan adalah suatu perkembangan pemikiran kritis yang disusun secara metodis dengan menggunakan bahasa yang runtut untuk menangani suatu masalah. Dari penjelasan yang telah diberikan, yang dimaksud dengan algoritma adalah jika langkah penyelesaian suatu masalah berbeda dengan definisi matematika yang dapat digunakan untuk menyelesaikan masalah dari sebuah buku, maka suatu algoritma menyelesaikan suatu masalah dengan menggunakan rumus atau komputer. Dalam matematika dan ilmu komputer, mendefinisikan suatu algoritma adalah proses komputasi multi-langkah. Algoritma juga dapat digunakan untuk perhitungan, penalaran otomatis dan juga pengolahan data. Namun, mendefinisikan suatu algoritma adalah suatu sistem atau metode yang sangat efektif bila digunakan sebagai pemecah masalah yang terdefinisi dengan baik untuk menghitung suatu fungsi. Nanti jika "Algoritma" ditambahkan ke "Data atau Struktur Data", menjadi "Program", yang biasanya digunakan dalam program komputer. Dengan demikian, dapat disimpulkan bahwa “program” merupakan kumpulan beberapa pernyataan dalam suatu komputer, sedangkan prosedur dan langkah sistematis yang terdapat dalam suatu program merupakan algoritma. Perhitungan pemrograman adalah perkembangan langkah-langkah yang dikoordinasikan secara efisien dan sah untuk menangani suatu masalah dengan menggunakan bahasa pemrograman. Kata koheren adalah slogan dalam perhitungan. Metode-metode ini digunakan untuk menjawab pertanyaan-pertanyaan penting spesifik yang diambil dari Garis Besar Schaum Edisi Keempat sebagaimana ditulis di bawah ini:

## a. Soal Table

Tabel 1

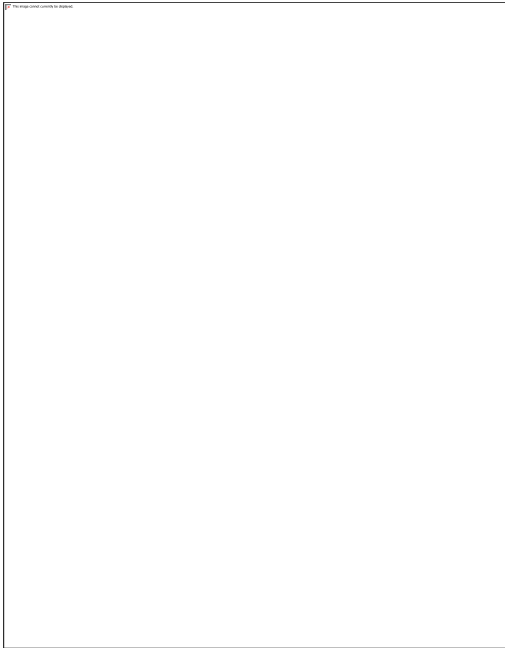
A	B	While $a < 5$	Hasil Boolean	$b = a + 1$	$b = b + a$	Cetak a	Cetal b
0	0	$0 < 5$	T	$1 = 0 + 1$	$1 = 0 + 1$	1	
		$1 < 5$	T	$2 = 1 + 1$	$3 = 1 + 2$	2	
		$2 < 5$	T	$3 = 2 + 1$	$6 = 3 + 3$	3	
		$3 < 5$	T	$4 = 3 + 1$	$10 = 6 + 4$	4	
		$4 < 5$	T	$5 = 4 + 1$	$15 = 10 + 5$	5	
		$5 < 5$	F				15

Perhatikan kolom Boolean Result pada Tabel 1. Selama nilainya True (T), maka iterasi dilakukan pada badan iterasi yang terdiri dari perintah.:  $a = a + 1$ ,  $b = b + 1$ , dan cetak a (perhatikan pula *flowchart* dan *pseudocode*-nya).

Jika disetel ke False, pengulangan akan berhenti. Dimulai dari kondisi pengulangan, terdapat perintah untuk mencetak nilai variabel b (print b), nilai terakhir yang dicetak adalah 15. Mengingat ide faktor, nilai dari sebuah variabel adalah nilai terakhir dari hasil estimasi. Perhatikan bahwa nilai variabel b terus bertambah dari 1 ke 3, 6, 10, dan terakhir 15. Nilai awal variabel b adalah nol ( $b = 0$ ).

Valid atau Palsu, dengan mempertimbangkan efek lanjutan dari uji sementara, dengan pedoman yang menyertainya

Berapakah  $0 < 5$ ?, Maka Hasilnya Benar;



Variabel  $i$  berubah, dari nol, berikutnya 1, 2, 3, 4, 5 karena ada instruksi di dalam badan perulangan, yaitu:  $a = a + 1$ . Variabel  $i$  adalah nilai awal 0 ( $a = 0$ ).

□

Menentukan hasil algoritma pseudocode seringkali sulit, terutama pada logika struktur kontrol berulang atau algoritma loop bersarang, yang seringkali rumit sehingga sulit untuk memprediksi atau menentukan hasilnya. Misalnya, apa akhir dari sebuah loop? Apa batas awal (awal) dan akhir (berhenti) dari sebuah loop? Di mana memulainya dan di mana mencapai batas tubuh dalam satu lingkaran? Haruskah kita menyelesaikannya atau berhenti? Apakah hasil perulangannya benar atau salah. Apakah akan berulang selama hasilnya boolean atau kondisinya benar? Apakah diulang sampai kondisinya benar? Selain itu, karena akal manusia mempunyai keterbatasan dalam mengingat atau melakukan hal-hal tersebut, maka pentingnya suatu alat mutlak diperlukan dalam proses analisis ini. Berdasarkan kondisi permasalahan tersebut, penulis mencari suatu desain atau metode yang dapat dengan mudah menganalisis variabel-variabel baik input maupun output, serta logika algoritma pada algoritma pseudocode. Sebuah panel yang penulis sebut sebagai desktop dashboard. Fungsi panel dokumen yaitu mencatat setiap perubahan pada beberapa variabel yang dianalisis, baik variabel input maupun output. Artikel ini juga merupakan lanjutan dari artikel penulis sebelumnya yang berjudul “Perancangan Algoritma Perkalian Matriks Operasional Menggunakan Metode Flowchart”. Artikel ini bertujuan untuk mengenal bentarnya jawaban yang ditulis oleh sipenulis.



## KESIMPULAN

Algoritma adalah serangkaian langkah pemecahan masalah yang dikoordinasikan secara efisien dan masuk akal. Kata sah adalah semboyan. Cara-cara tersebut harus diperiksa, menyiratkan bahwa nilai realitasnya harus dapat ditentukan apakah valid atau menyesatkan. Perhitungan juga merupakan dasar pemrograman. Kemudian lagi, perhitungan pemrograman adalah sebuah pendekatan untuk mencoba mengatasi masalah dengan menggunakan bahasa pemrograman disusun secara menyeluruh. Struktur dasar algoritma pemrograman menjadi landasan pemrograman itu sendiri. Struktur dasar algoritma ini adalah urutan, pemilihan dan pengulangan setiap tindakan atau dapat dicapai dengan metode campuran.

Dalam membuat atau menulis algoritma pemrograman, notasi bahasa dan struktur dasar diperlukan agar algoritma pemrograman dapat dinilai benar dan bebas kesalahan. Agar suatu program ini dapat dijalankan dengan komputer, Program harus ditulis dalam bahasa yang dapat dimengerti oleh komputer. Bahasa komputer yang digunakan untuk menulis program disebut bahasa pemrograman. Rangkaian langkah sistematis untuk menyelesaikan masalah disebut algoritma. Metode penelitian dokumen adalah metode ini pengumpulan beberapa data dan informasi yang melalui pemeriksaan dan informasi berupa dokumen internal/eksternal suatu perusahaan.

## DAFTAR PUSTAKA

- Jr., Frank Ayres, et.al. (1999). *Schaum's Outlines of Theory and Problems of Calculus. Fourth Edition*. Kalkulus. Edisi Keempat. Alih Bahasa: Nur Danarjaya, M.Sc. Editor: Amalia Safitri, S.TP., M.Si. Jakarta: Penerbit Airlangga.
- Ngoen, Thompson Susabda. (2009). *ALGORITMA DAN STRUKTUR DATA Bahasa C*. Jakarta: Penerbit Mitra Wacana Media. Edisi Pertama.
- Robertson, Lesiey Anne. 2004. *Simple Program Design. A Step-by-Step Approach. Fourth Edition*. Hongkong: CourseTechnology.
- Sutanta, Edhy. (2004), *Algoritma Teknik Penyelesaian Permasalahan Untuk Komputasi*. Graha Ilmu. Yogyakarta. <http://www.akmi-baturaja.ac.id/wp-content/uploads/2012/07/Logika-dan-Algoritma.pdf>
- Agusnawar. (2004). *Resepsionis Hotel*. Jakarta: Gramedia Pustaka Utama AA Eryanto, Moekijat. 2019 *Tinjauan Pustaka Definisi Supervisor*. 1-29. Diakses pada 19 juli 2023 dalam <http://repository.um-surabaya.ac.id/3737>

- Bagyono. (2005). *Pariwisata dan Perhotelan*. Bandung: Alfabeta
- Bagyono. (2012). *Teori dan Praktik Hotel Front Office*. Bandung: Alfabeta
- Darsono, Agustinus. (2011). *Front Office Hotel*. Jakarta: Grasindo
- Sambodo, Agus dan Bagyono. (2006). *Dasar – Dasar Kantor Depan Hotel*. Yogyakarta: Andi Sihite,
- Richard. (2000). *Hotel Management: Pengelolaan Hotel*. Surabaya: SIC Soenarno, Adi. (2006). *Front Office Management*. Yogyakarta: Andi Offcet