

16-02-2023

Algoritma & Struktur Data

Kristina P. Sinaga

I. HEADLINE

A. Summary

1. Membahas definisi algoritma sebagai sekumpulan instruksi yang memecahkan masalah tertentu atau melakukan tugas tertentu.
2. Pentingnya struktur data dalam mengatur dan mengelola data dalam algoritma.
3. Penggunaan Pseudocode untuk mendeskripsikan algoritma dengan cara yang terstruktur.

I. SECTION 1

A. Konsep Dasar Struktur Data

1. Algoritma dan struktur data adalah dua konsep dasar dalam ilmu komputer yang terkait erat dan memainkan peran penting dalam menyelesaikan masalah komputasi secara efisien.
2. Untuk merancang algoritme yang efisien, penting untuk memiliki pemahaman yang baik tentang struktur data yang tersedia dan propertinya, dan untuk memilih struktur data yang paling tepat untuk tugas yang sedang dikerjakan.
3. Untuk merancang struktur data yang efektif, penting untuk mempertimbangkan jenis operasi yang akan dilakukan pada data dan memilih struktur yang akan mengoptimalkan operasi tersebut.

I. SECTION 1

B. Defenisi Algoritma

1. Algoritma adalah seperangkat instruksi langkah demi langkah yang dirancang untuk memecahkan masalah tertentu atau menyelesaikan tugas tertentu.
2. Dalam ilmu komputer, algoritma biasanya diimplementasikan sebagai program atau fungsi yang mengambil data masukan dan menghasilkan data keluaran.
3. Algoritma dapat dianalisis berdasarkan kompleksitas waktu dan kompleksitas ruangnya, yang masing-masing mengacu pada jumlah waktu dan sumber daya memori yang diperlukan untuk menjalankan algoritme.

I. SECTION 1

B. Contoh Algoritma

Beberapa contoh algoritma:

1. Algoritma pengurutan (*sorting*): Algoritma pengurutan digunakan untuk mengatur daftar item dalam urutan tertentu, seperti urutan naik atau turun. Contoh algoritma pengurutan termasuk pengurutan gelembung, pengurutan penyisipan, pengurutan pilihan, pengurutan gabungan, dan pengurutan cepat.
2. Algoritme pencarian: Algoritme pencarian digunakan untuk menemukan item tertentu dalam daftar item. Contoh algoritma pencarian termasuk pencarian linier dan pencarian biner.
3. Algoritme pembelajaran mesin (*Machine learning*): Algoritma pembelajaran mesin digunakan untuk mempelajari pola dan hubungan dalam data, dan dapat digunakan untuk tugas-tugas seperti klasifikasi, regresi, dan pengelompokan. Contoh algoritma pembelajaran mesin termasuk regresi linier, regresi logistik, k-nearest neighbor (KNN), pohon keputusan, dan jaringan syaraf tiruan (ANN).

I. SECTION 1

C. Defenisi Struktur Data

1. Struktur data, di sisi lain, adalah cara mengatur dan menyimpan data dalam program komputer. Struktur data menyediakan cara untuk merepresentasikan dan memanipulasi data, dan struktur data yang berbeda dioptimalkan untuk operasi yang berbeda.
2. Contoh umum dari struktur data meliputi larik, daftar tertaut, tumpukan, antrean, pohon, dan grafik.

I. SECTION 1

D. Contoh Struktur Data

Berikut adalah beberapa contoh umum dari struktur data:

1. Array: Array adalah kumpulan elemen, masing-masing diidentifikasi dengan indeks atau kunci. Array digunakan untuk menyimpan kumpulan elemen sekuensial berukuran tetap dari tipe yang sama, seperti daftar angka.
2. Daftar tertaut (*linked lists*): Daftar tertaut adalah kumpulan elemen, masing-masing berisi nilai dan referensi ke elemen berikutnya dalam daftar. Daftar tertaut dapat digunakan untuk menyimpan kumpulan elemen yang dipesan dengan berbagai ukuran dan jenis, dan sering digunakan untuk mengimplementasikan tumpukan dan antrean.

I. SECTION 1

D. Contoh Struktur Data

Berikut adalah beberapa contoh umum dari struktur data:

1. Antrian: Antrian adalah kumpulan elemen yang mendukung dua operasi: enqueue, yang menambahkan elemen ke belakang antrean, dan dequeue, yang menghapus elemen depan dari antrean. Antrian sering digunakan dalam aplikasi yang memerlukan perilaku first-in-first-out (FIFO), seperti penjadwalan pekerjaan di sistem operasi.
2. Pohon: Pohon adalah struktur data hierarkis yang terdiri dari simpul-simpul yang memiliki hubungan induk-anak. Pohon sering digunakan untuk merepresentasikan hubungan hierarkis, seperti struktur sistem file atau organisasi situs web.

I. SECTION 1

D. Contoh Struktur Data

Berikut adalah beberapa contoh umum dari struktur data:

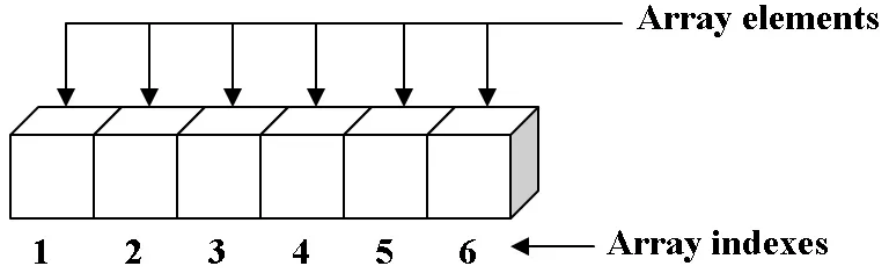
1. Tumpukan: Tumpukan adalah kumpulan elemen yang mendukung dua operasi: push, yang menambahkan elemen ke bagian atas tumpukan, dan pop, yang menghapus elemen teratas dari tumpukan. Tumpukan sering digunakan dalam aplikasi yang memerlukan perilaku last-in-first-out (LIFO), seperti fungsi undo/redo di editor teks.
2. Tabel hash: Tabel hash adalah struktur data yang memetakan kunci ke nilai menggunakan fungsi hash. Tabel hash sering digunakan untuk pencarian cepat dan digunakan di banyak aplikasi, seperti pengindeksan database dan caching.

I. SECTION 1

E. Contoh Penggunaan Algoritma dan Struktur Data

1. Pemilihan struktur data dapat berdampak signifikan pada efisiensi algoritma yang beroperasi pada data tersebut. Misalnya, beberapa algoritme mungkin memerlukan akses cepat ke masing-masing elemen data, sementara yang lain mungkin memerlukan penyisipan atau penghapusan elemen yang efisien. Pemilihan struktur data juga dapat memengaruhi kompleksitas ruang suatu algoritma, karena struktur data yang berbeda membutuhkan jumlah memori yang berbeda untuk menyimpan data yang sama.

I. SECTION 1
F. Subsection B



One-dimensional array with six elements

- Tipe data abstrak (ADT) adalah cara menentukan sekumpulan operasi yang dapat dilakukan pada tipe data, tanpa menentukan bagaimana operasi tersebut diimplementasikan.

I. SECTION 1

G. Notasi ALgoritma

- Notasi algoritma adalah cara untuk menyatakan kompleksitas waktu dan ruang dari suatu algoritma dengan menggunakan notasi matematika. Notasi yang paling umum digunakan adalah notasi O besar, Omega besar, dan Theta besar.
- Notasi O besar, diwakili oleh simbol " O ", digunakan untuk menggambarkan batas atas kompleksitas waktu atau ruang dari suatu algoritma. Ini mengungkapkan skenario terburuk untuk algoritma, atau jumlah maksimum waktu atau ruang yang dapat digunakan algoritma. Misalnya, jika suatu algoritma memiliki kompleksitas waktu $O(n)$, itu berarti waktu yang dibutuhkan untuk menjalankan algoritma tumbuh secara linier dengan ukuran input.

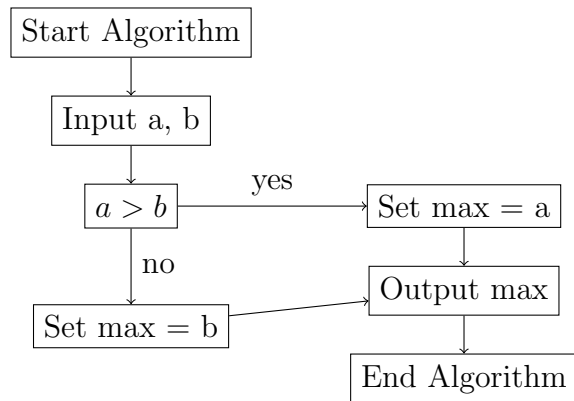
I. SECTION 1

H. Representasi Grafis dari Sebuah Algoritma

- Representasi grafis umum dari suatu algoritma adalah flowchart. Diagram alir menggunakan simbol dan panah yang berbeda untuk mewakili berbagai langkah dan keputusan yang terlibat dalam pemecahan masalah. Berikut adalah contoh diagram alir untuk algoritme sederhana untuk menemukan maksimum dua angka:
- Dalam diagram alir:
 1. kotak persegi panjang mewakili langkah-langkah proses
 2. kotak berbentuk wajik mewakili titik keputusan
 3. Panah mewakili alur algoritma

I. SECTION 1

H. Mencari Nilai Maximum 2 Bilangan



- Algoritma dimulai dengan input dua angka, a dan b , dan kemudian memeriksa mana yang lebih besar. Jika a lebih besar dari b , maka algoritme menetapkan nilai maksimum ke a . Jika tidak, itu menetapkan nilai maksimum ke b . Akhirnya, algoritme mengeluarkan nilai maksimum dan berakhir.

I. SECTION 1

I. Simbol flowchart

Berikut adalah beberapa simbol flowchart yang umum digunakan:

- Simbol terminal (oval): digunakan untuk menunjukkan awal dan akhir dari flowchart.
- Simbol proses (persegi panjang): digunakan untuk mewakili proses atau tindakan yang dilakukan.
- Simbol keputusan (diamond): digunakan untuk mewakili titik keputusan atau pertanyaan dengan dua atau lebih kemungkinan jawaban.
- Simbol input/output (jajar genjang): digunakan untuk mewakili operasi input atau output.
- Simbol konektor (lingkaran): digunakan untuk menghubungkan berbagai bagian diagram alur.

I. SECTION 1

I. Simbol flowchart

Berikut adalah beberapa simbol flowchart yang umum digunakan:

- Simbol konektor (lingkaran): digunakan untuk menghubungkan berbagai bagian diagram alur.
- Simbol konektor di luar halaman (dua garis paralel): digunakan untuk menghubungkan bagian-bagian diagram alur yang terletak di halaman atau lembaran berbeda.
- Simbol proses yang telah ditentukan sebelumnya (persegi panjang dengan garis ganda di sisinya): digunakan untuk mewakili proses yang sudah ditentukan atau didokumentasikan di tempat lain.
- Simbol penundaan (bentuk setengah kapsul): digunakan untuk menunjukkan penundaan atau masa tunggu di diagram alur.

I. SECTION 1

I. Simbol flowchart

Berikut adalah beberapa simbol flowchart yang umum digunakan:

- Simbol dokumen (persegi panjang dengan dasar bergelombang): digunakan untuk menunjukkan suatu dokumen atau laporan dalam flowchart.
- Simbol database (silinder): digunakan untuk mewakili database atau lokasi penyimpanan data.

I. SECTION 1

J. Algoritma Pseudocode

- Pseudocode adalah bahasa informal tingkat tinggi yang digunakan untuk menggambarkan logika program komputer. Ini bukan bahasa pemrograman tertentu, melainkan cara untuk merepresentasikan langkah-langkah yang terlibat dalam memecahkan masalah atau mengimplementasikan algoritme menggunakan sintaks mirip bahasa alami.
- Pseudocode dapat ditulis dengan cara terstruktur atau tidak terstruktur.
- Pseudocode terstruktur menggunakan struktur kontrol seperti loop, pernyataan bersyarat, dan subrutin untuk menggambarkan logika suatu program.
- Pseudocode tidak terstruktur menggunakan pernyataan sederhana seperti bahasa Inggris untuk menggambarkan logika suatu program.

I. SECTION 1

J. Algoritma Pseudocode

- Algoritma pseudocode ini menjelaskan proses sederhana untuk menambahkan dua angka bersama dan menampilkan hasilnya.
- 0: $a \leftarrow 1$
0: **if** $a > 0$ **then**
0: $b \leftarrow 2$
0: **if** $b > a$ **then**
0: $c \leftarrow a + b$
0: **else**
0: $c \leftarrow b - a$
0: **end if**
0: **end if**
0: **return** $c = 0$

. KESIMPULAN

.

1. algoritma adalah konsep dasar dalam ilmu komputer dan digunakan untuk memecahkan berbagai macam masalah di berbagai bidang.
2. Memahami konsep kunci dan elemen algoritma, seperti struktur data, pseudocode, dan bagan alur (*flowchart*), dapat membantu dalam pengembangan algoritme yang efisien dan efektif untuk tugas tertentu.