

Laporan
Algoritma dan Pemrograman
Modul 1 dan 2



NAMA: Rido Juana

NIM: 24241036

PROGRAM STUDI TEKNOLOGI PENDIDIKAN
FAKULTAS SAINS, TEKNIK DAN TERAPAN (FSTT)
UNIVERSITAS PENDIDIKAN MANDALIKA
TAHUN 2024

Modul 1

Pengantar Algoritma dan pemrograman

Adam Bachtiar, S.Kom.,M.MT

Apa itu algoritma?

Algoritma adalah urutan dari beberapa langkah yang logis guna menyelesaikan masalah .

1. Apakah sebuah bilangan adalah bilangan genap atau ganjil ?

Untuk menentukan apakah sebuah bilangan adalah bilangan genap atau ganjil adalah :

Masukkan bilang bulat

Bagikan bilang dengan 2

Jika sisa pembagiannya 0, maka bilangan tersebut genap

Jika sisa pembagiannya bukan 0, maka bilangan tersebut ganjil

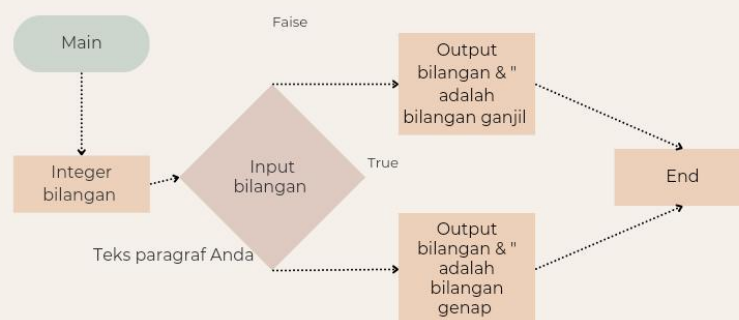
Bilangan genap adalah bilangan asli yang habis dibagi 2 atau kelipatannya. Contoh bilangan genap adalah 2, 4, 6, 8, dan seterusnya.

Bilangan ganjil adalah bilangan asli yang tidak habis dibagi 2. Contoh bilangan ganjil adalah 1, 3, 5, 7, dan seterusnya

Algoritma Menentukan Bilangan Ganjil atau Genap

- Input sebuah bilangan (bilangan bulat)
- Bagi bilangan yang sudah diinputkan dengan angka 2
- Jika sisa pembagian (0) maka bilangan tersebut genap
- Jika sisa pembagian bukan (0) maka bilangan tersebut ganjil

Flowchart Menentukan Bilangan Ganjil atau Genap



Algoritma
menentukan
bilangan ganjil
atau genap

**2. Manakah rute dengan jalur terpendek, jika ada dua rute yang
Akan dibandingkan ?**

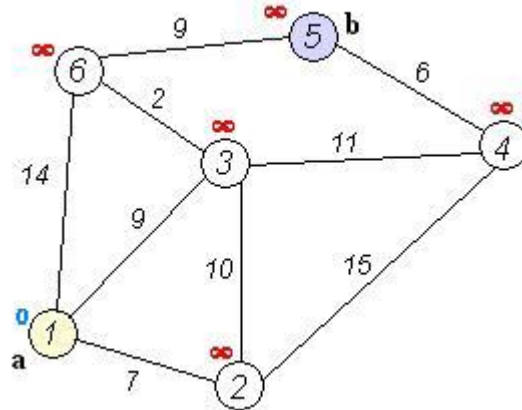
Untuk menentukan rute terpendek antara dua lokasi, Anda bisa menggunakan algoritma untuk mencari jalur terpendek. Beberapa algoritma yang bisa digunakan untuk mencari jalur terpendek, antara lain:

- Algoritma Dijkstra
- Algoritma Bellman-Ford
- Algoritma A*
- Algoritma Floyd-warshall

Rute terpendek adalah jalur antara dua lokasi yang melintasi jarak minimum melalui jaringan jalan atau di peta. Manfaat menggunakan rute terpendek adalah meringkas perjalanan dan menghemat biaya perjalanan

Algoritma Dijkstra

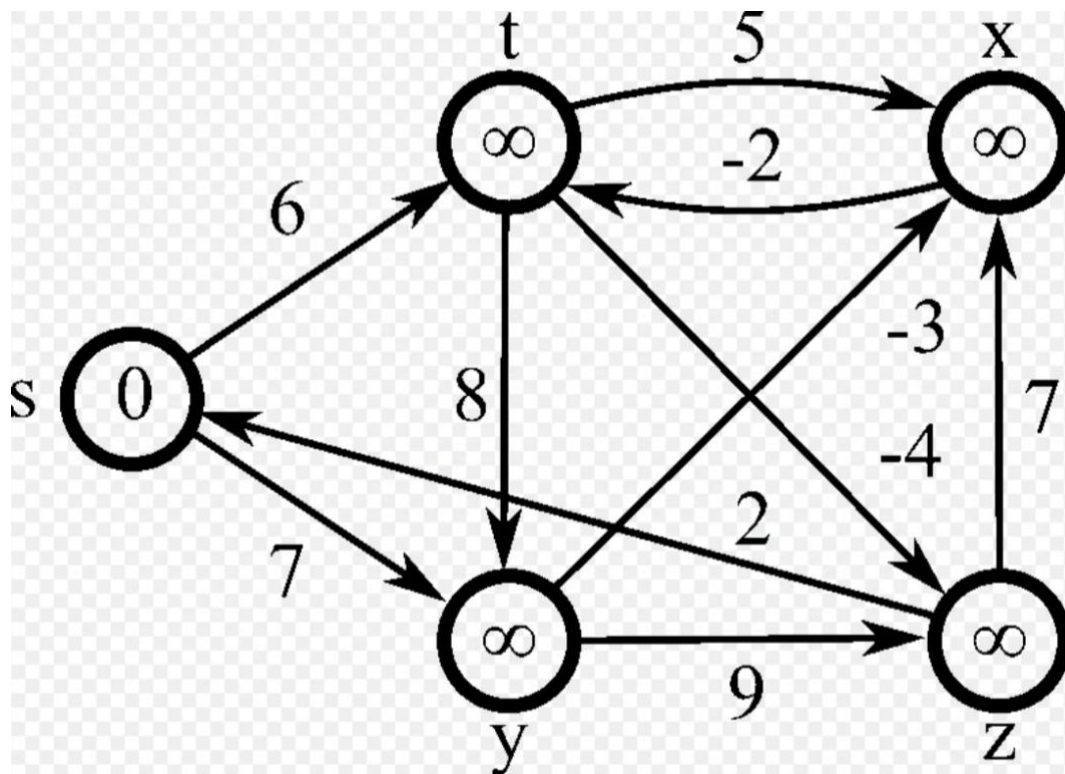
Algoritma Dijkstra: Algoritma ini dapat digunakan untuk menemukan jalur terpendek antara simpul dalam grafik berbobot, seperti jaringan jalan. Algoritma ini akan terus berjalan hingga semua simpul yang dapat dijangkau dalam grafik telah dikunjungi



Algoritma Dijkstra untuk menemukan jalur terpendek antara a dan b .
 Algoritma ini memilih simpul yang belum dikunjungi dengan jarak terpendek, menghitung jarak melalui simpul tersebut ke setiap tetangga yang belum dikunjungi, dan memperbarui jarak tetangga jika lebih kecil. Tandai telah dikunjungi (atur ke merah) setelah selesai dengan tetangga.

Algoritma Bellman-Ford

Algoritme Bellman–Ford menghitung jarak terpendek (dari satu sumber) pada sebuah digraf berbobot. Maksudnya dari satu sumber ialah bahwa ia menghitung semua jarak terpendek yang berawal dari satu titik node. Algoritme Dijkstra dapat lebih cepat mencari hal yang sama dengan syarat tidak ada sisi (edge) yang berbobot negatif. Maka Algoritme Bellman-Ford hanya digunakan jika ada sisi berbobot negatif.



Algoritme Bellman-Ford menggunakan waktu sebesar $O(V.E)$, di mana V dan E adalah banyaknya sisi dan titik.

Algoritma A*

Algoritma A* merupakan algoritma komputer yang menggunakan estimasi jarak dengan menggunakan pencarian jalur terdekat untuk mencapai tujuan dan memiliki fungsi heuristik yang digunakan sebagai dasar pertimbangan untuk menentukan pilihan sejumlah alternatif untuk mencapai sasaran dengan efektif.

Cara kerja algoritma A*

A* menggunakan Best First Search (BFS) dan menemukan jalur dengan biaya terkecil (least-cost path) dari node awal (initial node) yang diberikan ke node tujuan (goal node).

Algoritma ini menggunakan fungsi heuristik jarak ditambah biaya (biasa dinotasikan dengan $f(x)$) untuk menentukan urutan di mana search-nya melalui node-node yang ada pada tree.

Notasi yang dipakai oleh algoritma A* adalah sebagai berikut:

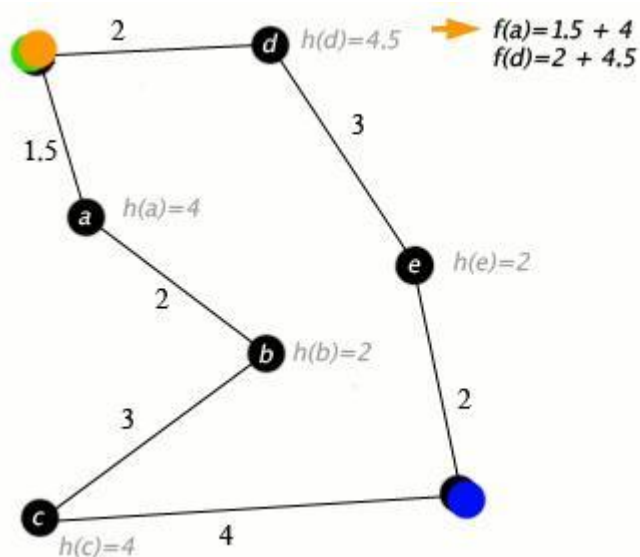
$$F(n) = g(n) + h(n)$$

Dimana

$F(n)$ = biaya estimasi terendah

$G(n)$ = biaya dari node awal ke node n

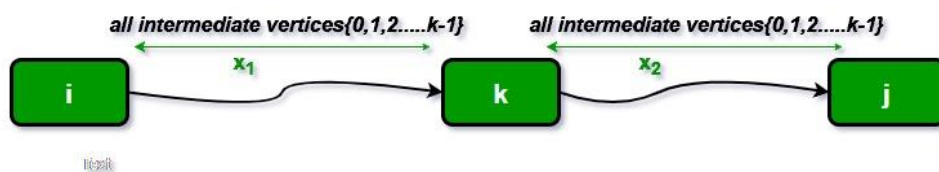
$H(n)$ = perkiraan biaya dari node n ke node akhir



Algoritma Floyd–warshall

Adalah algoritma fundamental dalam ilmu komputer dan teori grafik. Algoritma ini digunakan untuk menemukan jalur terpendek antara semua pasangan simpul dalam grafik berbobot. Algoritma ini sangat efisien dan dapat menangani grafik dengan bobot sisi positif dan negatif, menjadikannya alat serbaguna untuk memecahkan berbagai masalah jaringan dan konektivitas.

Gambar berikut menunjukkan properti substruktur optimal di atas dalam algoritma Floyd Warshall:



Algoritma Kode Semu Floyd Warshall :

Untuk $k = 0$ sampai $n - 1$

Untuk $i = 0$ sampai $n - 1$ Untuk $j =$

0 sampai $n - 1$

$\text{Jarak}[i, j] = \min(\text{Jarak}[i, j], \text{Jarak}[i, k] + \text{Jarak}[k, j])$

Di mana i = Node sumber, j = Node tujuan, k = Node perantara

3. Mengurutkan 3 bilangan yang diinputkan, mulai dari yang terkecil Sampai yang terbesar !

Untuk mengurutkan 3 bilangan yang diinputkan dari yang terkecil sampai yang terbesar, Anda bisa menggunakan model urut naik (ascending):

Bandingkan angka-angka satu sama lain

Tulis angka yang lebih kecil sebelum angka yang lebih besar

Contoh algoritma pengurutan data yang bisa digunakan adalah algoritma bubble sort. Algoritma ini membandingkan data-data yang bersebelahan. Jika data yang bersebelahan tidak berurutan, maka data tersebut akan ditukar.

Algoritma Mengurutkan3Bilangan

{dibaca 3 bilangan bulat (A, B, C) yang pasti berbeda, kemudian menampilkan bilangan tersebut dari yang terkecil}

Deklarasi:

A, B, C : integer

Deskripsi:

Read(A, B, C)

If A<B and A<C then

If B<C then

Write(A, B, C)

Else

Write(A, C, B)

Endif

Else if $B < A$ and $B < C$ then

If $A < C$ then

Write(B, A, C)

Else

Write(B, C, A)

Endif

Else

If $B < A$ then

Write(C, B, A)

Else

Write(C, A, B)