

Untari Novia Wisesty
Hani Nurrahmi
Prasti Eko Yunanto
Rita Rismala
Febryanti Sthevanie



Struktur Data

Menggunakan

C++



STRUKTUR DATA

MENGGUNAKAN C++

Untari Novia Wisesty
Hani Nurrahmi
Prasti Eko Yunanto
Rita Rismala
Febryanti Sthevanie



PENERBIT KBM INDONESIA

Adalah penerbit dengan misi memudahkan proses penerbitan buku buku penulis di tanah air Indonesia. Serta menjadi media sharing proses penerbitan buku.

STRUKTUR DATA MENGGUNAKAN C++

Copyright @2025 By Untari Novia Wisesty., dkk

All right reserved

Penulis

Untari Novia Wisesty

Hani Nurrahmi

Prasti Eko Yunanto

Rita Rismala

Febryanti Sthevanie

Desain Sampul

Aswan Kreatif

Tata Letak

Sofitahm

Editor

Dr. Muhamad Husein Maruapey, Drs., M.Sc.

Background isi buku di ambil dari <https://www.freepik.com/>

Official

Depok, Sleman-Jogjakarta (Kantor)

Penerbit Karya Bakti Makmur (KBM) Indonesia

Anggota IKAPI/No. IKAPI 279/JTI/2021

081357517526 (Tlpn/WA)

Website

<https://penerbitkbm.com>

www.penerbitbukumurah.com

Email

naskah@penerbitkbm.com

Distributor

<https://penerbitkbm.com/toko-buku/>

Youtube

Penerbit KBM Sastrabook

Instagram

@penerbit.kbmindonesia

@penerbitbukujogja

ISBN: 978-634-202-715-8

Cetakan ke-1, September 2025

15,5 x 23 cm, x +357 halaman

Isi buku diluar tanggungjawab penerbit

Hak cipta merek KBM Indonesia sudah terdaftar di DJKI-Kemenkumham dan isi buku dilindungi undang-undang.

Dilarang keras menerjemahkan, memfotokopi, atau memperbanyak sebagian atau seluruh isi buku ini tanpa seizin penerbit karena beresiko sengketa hukum

Sanksi Pelanggaran Pasal 113
Undang-Undang No. 28 Tahun 2014 Tentang Hak Cipta

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp 100.000.000 (seratus juta rupiah).
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp 500.000.000,00 (lima ratus juta rupiah).
3. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf a, huruf b, huruf e, dan/atau huruf g untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 4 (empat) tahun dan/atau pidana denda paling banyak Rp 1.000.000.000,00 (satu miliar rupiah).
4. Setiap Orang yang memenuhi unsur sebagaimana dimaksud pada ayat (3) yang dilakukan dalam bentuk pembajakan, dipidana dengan pidana penjara paling lama 10 (sepuluh) tahun dan/atau pidana denda paling banyak Rp 4.000.000.000,00 (empat miliar rupiah).

KATA PENGANTAR

Puji syukur Kami panjatkan ke hadirat Allah SWT, karena atas rahmat dan karunia-Nya, buku ajar ini yang berjudul **“Struktur Data menggunakan C++”** dapat diselesaikan dengan baik. Buku ini disusun oleh lima penulis dengan latar belakang dosen pengampu mata kuliah Struktur Data, dengan tujuan utama untuk memberikan pemahaman yang komprehensif kepada mahasiswa Sarjana Informatika maupun bidang *Computer Science*.

Penyusunan buku ini dilandasi oleh kebutuhan mahasiswa akan bahan ajar yang tidak hanya menyajikan teori, tetapi juga memberikan contoh penerapan secara jelas dan sistematis. Oleh karena itu, buku ini dirancang dengan penjelasan tahap demi tahap dalam membangun struktur data, dilengkapi dengan ilustrasi gambar yang membantu visualisasi proses, serta contoh sintaks dalam bentuk *pseudocode* dan bahasa C++. Penyajian *pseudocode* diharapkan dapat memberikan fleksibilitas bagi mahasiswa yang ingin mengimplementasikan konsep struktur data ke dalam bahasa pemrograman lain selain C++.

Selain materi utama, buku ini juga menyediakan contoh soal beserta pembahasan, serta latihan soal di setiap bab. Dengan demikian, mahasiswa tidak hanya dapat memahami teori dan implementasi, tetapi juga terbiasa menghadapi variasi permasalahan yang relevan dengan materi yang dipelajari. Karakteristik ini diharapkan menjadikan buku ini sebagai sumber belajar yang sesuai dengan kebutuhan mahasiswa, baik dalam perkuliahan maupun dalam pengembangan kemampuan mandiri.

Penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada seluruh tim dosen Mata Kuliah Struktur Data di Fakultas Informatika, Universitas Telkom, yang telah berkontribusi dalam menyusun, mengevaluasi, serta secara berkesinambungan memperbaiki materi yang menjadi dasar dari buku ini.

Ucapan terima kasih juga kami sampaikan kepada Fakultas Informatika Universitas Telkom yang telah memberikan kesempatan,

masukan, serta dukungan pendanaan sehingga proses penyusunan dan penerbitan buku ini dapat terlaksana dengan baik.

Kami menyadari bahwa buku ini masih jauh dari sempurna. Oleh karena itu, kritik dan saran yang membangun sangat kami harapkan demi penyempurnaan di edisi berikutnya. Masukan dapat disampaikan melalui email salah satu penulis yang tercantum pada bagian akhir buku ini.

Akhir kata, semoga buku ajar ini dapat memberikan manfaat yang sebesar-besarnya bagi mahasiswa, dosen, maupun pembaca lainnya yang tertarik untuk mendalami struktur data menggunakan C++.

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	iii
DAFTAR GAMBAR	v
Bab 1 Pengantar Linked List	1
1.1 Definisi dan Karakteristik dari Struktur Data List	1
1.2 Abstract Data Type.....	2
1.3 Alamat Memori dan Pointer	5
1.4 Soal Latihan	7
Bab 2 Single Linked List	9
2.1 ADT Single Linked List	9
2.2 Implementasi dari ADT Single Linked List	14
2.4 Variasi Single Linked List dengan Last	37
2.5 ADT dan Implementasi Lengkap dari Single Linked List (tanpa Last).....	45
2.6 Contoh Soal dan Pembahasan	51
2.7 Soal Latihan	63
Bab 3 Double Linked List	69
3.1 ADT DLL	69
3.2 Implementasi ADT Double Linked List	72
3.3 ADT dan Implementasi Lengkap dari Double Linked List (tanpa Last).....	96
3.4 Contoh Soal dan Pembahasan	103
3.5 Soal latihan	107
Bab 4 Circular Linked List.....	108
4.1 ADT Circular Single Linked List	108
4.2 Implementasi dari ADT Circular Single Linked List	112
4.3 Variasi Circular Single Linked List dengan Last	137
4.4 ADT dan Implementasi Lengkap dari Single Linked List (tanpa Last).....	145

4.5 Soal Latihan	151
Bab 5 Multi Linked List N-N	153
5.1 Definisi Struktur Data Multi Linked List N-N	154
5.2 Primitif Dasar dari Multi Linked List N-N	157
5.3 ADT dan Implementasi Lengkap dari Multi Linked List N-N	196
5.4 Variasi Multi Linked List N-N	223
5.5 Contoh Soal dan Pembahasan	273
5.6 Soal Latihan	283
Bab 6 Stack.....	285
6.1 ADT Stack.....	285
6.2 Implementasi ADT Stack	287
6.3 Soal Latihan	291
Bab 7 Queue	292
7.1 ADT Queue	292
7.2 Implementasi ADT <i>queue</i>	294
7.3 Soal Latihan	297
Bab 8 Tree.....	299
8.1 Karakteristik dari <i>Binary Search Tree</i>	300
8.2 ADT Struktur Data Binary Tree	301
8.3 Implementasi dari ADT <i>Binary Search Tree</i>	305
8.4 Soal Latihan	330
Bab 9 Graf.....	331
9.1 Definisi Graf	331
9.2 Representasi Graf	334
9.3 Definisi Struktur Data Graf dengan Representasi Multi Linked List.....	335
9.4 Penelusuran Graf.....	350
9.5 Soal Latihan	352
DAFTAR PUSTAKA	354
PROFIL PENULIS	355

DAFTAR GAMBAR

Gambar 1.1 Jenis-jenis struktur data non-primitive.....	1
Gambar 1.2 Gambar 1.2 Contoh ilustrasi Linked List: (a) List yang berisi 4 elemen, yang setiap elemennya memiliki link untuk terhubung dengan elemen berikutnya; (b) Variasi List yang lain, yang mana setiap elemen memiliki dua link untuk terhubung dengan elemen sebelum dan berikutnya	2
Gambar 1.3 Gambar 1.3 Bagan ilustrasi penyembunyian detail implementasi dari ADT	3
Gambar 2.1 Ilustrasi dari Single Linked List yang terdiri dari 4 elemen list	9
Gambar 2.2 Ilustrasi deklarasi tipe elemen dari Single Linked List	10
Gambar 2.3 Ilustrasi deklarasi tipe dari Single Linked List	10
Gambar 2.4 Ilustrasi kode dan simbol gambar alokasi elemen baru	16
Gambar 2.5 Ilustrasi pengisian data pada elemen baru	16
Gambar 2.6 Ilustrasi pencarian elemen list dengan menggunakan Sekuensial Search	17
Gambar 2.7 Ilustrasi kondisi awal penambahan data dengan list kosong.....	19
Gambar 2.8 Ilustrasi penambahan elemen pada list kosong.....	19
Gambar 2.9 Ilustrasi kondisi awal penambahan data dengan list tidak kosong.....	20
Gambar 2.10 Ilustrasi next elemen baru berisi alamat elemen pertama dari list.....	20
Gambar 2.11 Ilustrasi pemindahan elemen first ke elemen yang baru	20
Gambar 2.12 Ilustrasi kondisi awal list tidak kosong pada penambahan data diakhir	22
Gambar 2.13 Ilustrasi pencarian elemen terakhir dengan menggunakan pointer last.....	22
Gambar 2.14 Ilustrasi menghubungkan elemen terakhir dengan elemen baru	23
Gambar 2.15 Ilustrasi kondisi awal penambahan elemen di tengah list	25
Gambar 2.16 Ilustrasi menghubungkan elemen baru dengan q → next	25

Gambar 2.17 Ilustrasi menghubungkan elemen dengan info B ke elemen baru	25
Gambar 2.18 Ilustrasi penyimpanan alamat memori elemen yang akan dihapus	27
Gambar 2.19 Ilustrasi membuat first dari list menjadi NIL.....	27
Gambar 2.20 Ilustrasi penyimpanan alamat memori elemen pertama pada pointer p.....	28
Gambar 2.21 Ilustrasi update first dari list menjadi elemen kedua	28
Gambar 2.22 Ilustrasi penghapusan koneksi antara elemen yang dihapus dengan list	28
Gambar 2.23 Ilustrasi penyimpanan alamat memori elemen terakhir dari list.....	30
Gambar 2.24 Ilustrasi penyimpanan alamat memori dua elemen terakhir	30
Gambar 2.25 Ilustrasi penghapusan koneksi elemen terakhir dengan list .	30
Gambar 2.26 Ilustrasi pointer p dan q pada delete after di Single Linked List.....	33
Gambar 2.27 Ilustrasi pembaharuan next dari pointer q menjadi next dari pointer p.....	33
Gambar 2.28 Ilustrasi langkah terakhir delete after pada Single Linked List.....	33
Gambar 2.29 Ilustrasi pointer p yang bergerak dari elemen pertama hingga elemen terakhir	36
Gambar 2.30 Ilustrasi Single Linked List yang menggunakan pointer last ..	37
Gambar 3.1 Ilustrasi DLL yang terdiri dari 4 elemen list.....	69
Gambar 3.2 Ilustrasi deklarasi tipe elemen dari DLL	70
Gambar 3.3 Ilustrasi deklarasi tipe dari DLL	70
Gambar 3.4 Ilustrasi kode dan simbol gambar alokasi elemen baru	73
Gambar 3.5 Ilustrasi pengisian data pada elemen baru	74
Gambar 3.6 Ilustrasi pencarian elemen list dengan menggunakan <i>Sequential Search</i> secara maju	75
Gambar 3.7 Ilustrasi kondisi awal penambahan data dengan list kosong..	76
Gambar 3.8 Ilustrasi penambahan elemen pada list kosong	76
Gambar 3.9 Ilustrasi kondisi awal penambahan data dengan list tidak kosong.....	77
Gambar 3.10 Ilustrasi menghubungkan elemen baru dengan elemen pertama list.....	77
Gambar 3.11 Ilustrasi pemindahan elemen first ke elemen yang baru	78
Gambar 3.12 Ilustrasi kondisi awal list tidak kosong pada penambahan data di akhir	79

Gambar 3.13 Ilustrasi pencarian elemen terakhir dengan menggunakan pointer last.....	80
Gambar 3.14 Ilustrasi menghubungkan elemen terakhir dengan elemen baru.....	80
Gambar 3.15 Ilustrasi kondisi awal penambahan elemen di tengah list	81
Gambar 3.16 Ilustrasi menghubungkan elemen baru dengan elemen setelah q.....	82
Gambar 3.17 Ilustrasi menghubungkan elemen q dengan elemen baru	82
Gambar 3.18 Ilustrasi langkah pertama delete first pada kondisi 2	84
Gambar 3.19 Ilustrasi langkah kedua delete first pada kondisi 2	84
Gambar 3.20 Ilustrasi langkah pertama delete first pada kondisi 3	84
Gambar 3.21 Ilustrasi langkah kedua delete first pada kondisi 3	84
Gambar 3.22 Ilustrasi langkah ketiga delete first pada kondisi 3.....	85
Gambar 3.23 Ilustrasi langkah pertama delete last pada kondisi list berisi banyak elemen.....	86
Gambar 3.24 Ilustrasi langkah kedua delete last pada kondisi list berisi banyak elemen.....	86
Gambar 3.25 Ilustrasi langkah pertama delete after pada kondisi 3.....	88
Gambar 3.26 Ilustrasi langkah kedua delete after pada kondisi 3	88
Gambar 3.27 Ilustrasi langkah terakhir delete after DLL pada kondisi 3	89
Gambar 3.28 Ilustrasi menampilkan info DLL.....	91
Gambar 3.29 Ilustrasi Single Linked List yang menggunakan pointer last	92
Gambar 4.1 Ilustrasi dari Circular Single Linked List yang terdiri dari 4 elemen list	108
Gambar 4.2 Ilustrasi kode dan simbol gambar alokasi elemen baru	113
Gambar 4.3 Ilustrasi pengisian data pada elemen baru	114
Gambar 4.4 Ilustrasi pencarian elemen list dengan menggunakan Sekuensial Search.....	115
Gambar 4.5 Ilustrasi kondisi awal penambahan data dengan list kosong.....	116
Gambar 4.6 Ilustrasi penambahan elemen pada list kosong	117
Gambar 4.7 Ilustrasi kondisi awal penambahan data dengan list tidak kosong.....	117
Gambar 4.8 Ilustrasi next elemen baru berisi alamat elemen pertama dari list	117
Gambar 4.9 Ilustrasi pemindahan elemen first ke elemen yang baru	118
Gambar 4.10 Ilustrasi pemindahan elemen elemen terakhir	118

Gambar 4.11 Ilustrasi menghubungkan pointer next elemen terakhir ke elemen baru.....	119
Gambar 4.12 Ilustrasi kondisi awal list tidak kosong pada penambahan data diakhir	120
Gambar 4.13 Ilustrasi pencarian elemen terakhir dengan menggunakan pointer last.....	121
Gambar 4.14 Ilustrasi menghubungkan elemen terakhir dengan elemen baru.....	121
Gambar 4.15 Ilustrasi kondisi awal penambahan elemen di tengah list ..	123
Gambar 4.16 Ilustrasi menghubungkan elemen baru dengan q → next ..	124
Gambar 4.17 Ilustrasi menghubungkan elemen dengan info B ke elemen baru.....	124
Gambar 4.18 Ilustrasi penyimpanan alamat memori elemen yang akan dihapus	126
Gambar 4.19 Ilustrasi membuat first dari list menjadi NIL.....	126
Gambar 4.20 Ilustrasi penyimpanan alamat memori elemen pertama pada pointer p	127
Gambar 4.21 Ilustrasi update first dari list menjadi elemen kedua	127
Gambar 4.22 Ilustrasi penghapusan koneksi antara elemen yang dihapus dengan list	127
Gambar 4.23 Ilustrasi pencarian elemen terakhir	128
Gambar 4.24 Ilustrasi pointer next elemen terakhir ke elemen pertama..	128
Gambar 4.25 Ilustrasi penyimpanan alamat memori dua elemen terakhir	130
Gambar 4.26 Ilustrasi penghapusan koneksi elemen terakhir dengan list	131
Gambar 4.27 Ilustrasi pointer p dan q pada delete after di Single Linked List	133
Gambar 4.28 Ilustrasi pembaharuan next dari pointer q menjadi next dari pointer p	133
Gambar 4.29 Ilustrasi langkah terakhir delete after pada Single Linked List	134
Gambar 4.30 Ilustrasi pointer p yang bergerak dari elemen pertama hingga elemen terakhir	135
Gambar 4.31 Ilustrasi Single Linked List yang menggunakan pointer last	137
Gambar 5.1 Contoh Multi Linked List N-N untuk Relasi Mahasiswa dan Mata Kuliah	154
Gambar 5.2 Ilustrasi Multi Linked List Relasi N-N	155
Gambar 5.3 Ilustrasi pencarian relasi pada Multi Linked List N-N	164

Gambar 5.4 Ilustrasi pembuatan elemen relasi baru pada Multi Linked List N-N	174
Gambar 5.5 Ilustrasi penambahan elemen relasi baru pada Multi Linked List N-N	174
Gambar 5.6 Ilustrasi delete first pada list relasi. Panah abu-abu menandakan kondisi awal list sebelum prosedur delete first	177
Gambar 5.7 Ilustrasi delete last pada list relasi. Panah abu-abu menandakan kondisi awal list	181
Gambar 5.8 Ilustrasi delete after pada list relasi. Panah abu-abu menandakan kondisi awal list	182
Gambar 5.9 Ilustrasi delete first pada List_1	186
Gambar 5.10 Variasi Multi Linked List relasi N-N	223
Gambar 5.11 Ilustrasi pencarian relasi pada variasi Multi Linked List N-N	230
Gambar 5.12 Ilustrasi penghapusan elemen List_1 pada variasi Multi Linked List N-N.....	248
Gambar 5.13 Ilustrasi penghapusan elemen List_2 pada variasi Multi Linked List N-N.....	257
Gambar 5.14 Multi Linked List N-N dengan info List_1 dan List_2 yang terurut secara ascending	274
Gambar 5.15 Graf tak berarah yang merepresentasikan ketetanggaan antar kota	283
Gambar 6.1 Ilustrasi Stack dengan Representasi Single Linked List	286
Gambar 7.1 Ilustrasi Queue dengan Representasi Single Linked List	293
Gambar 8.1 Ilustrasi hirarki dari struktur data Tree	299
Gambar 8.2 Contoh Binary Search Tree	300
Gambar 8.3 Ilustrasi node dasar yang terdapat di dalam Binary Search Tree	301
Gambar 8.4 Ilustrasi deklarasi tipe data node dari struktur data Tree	301
Gambar 8.5 Ilustrasi deklarasi tipe data BST atau variabel Root	302
Gambar 8.6 Contoh array yang digunakan untuk membentuk BST	308
Gambar 8.7 Ilustrasi node 10 menjadi root dari BST	308
Gambar 8.8 Ilustrasi penambahan 5 menjadi node pada BST	309
Gambar 8.9 Ilustrasi penambahan 14 menjadi node pada BST	309
Gambar 8.10 Ilustrasi penambahan 8 menjadi node pada BST	309
Gambar 8.11 Contoh BST yang dibentuk dari sekumpulan nilai dari Array.....	310
Gambar 8.12 Contoh BST yang akan digunakan dalam Tree Traversal.....	312
Gambar 8.13 Ilustrasi subpohon pada step 1	313

Gambar 8.14 Ilustrasi subpohon pada step 2	313
Gambar 8.15 Ilustrasi subpohon pada step 3	314
Gambar 8.16 Ilustrasi subpohon pada step 4	315
Gambar 8.17 Ilustrasi penghapusan node 34 dari BST	323
Gambar 8.18 Ilustrasi penghapusan node 15 dari BST	323
Gambar 8.19 Contoh daun pada suatu BST.....	324
Gambar 8.20 Path dari node 20 adalah 15 - 34 - 20	326
Gambar 8.21 Height of Tree dari BST adalah 3, yaitu jarak terpanjang antara root menuju daun.....	327
Gambar 8.22 Depth dari node 20 adalah 2, karena terdapat dua edge untuk bisa sampai menuju root	328
Gambar 8.23 Level dari Tree di atas adalah 4 (mulai dari level 0 s.d 3)	329
Gambar 9.1 Contoh graf berdasarkan arah dan bobotnya, (a) graf tak berarah, (b) graf berarah, (c) graf berbobot	332
Gambar 9.2 Contoh graf sederhana (a), multigraf (b), dan graf lengkap (c)	333
Gambar 9.3 Contoh matriks ketetanggaan (b) untuk graf (a).....	334
Gambar 9.4 Contoh matriks insiden (b) untuk graf (a)	335
Gambar 9.5 Contoh representasi Multi Linked List 1-N dengan info (b) untuk graf (a)	336
Gambar 9.6 Ilustrasi connecting node	341
Gambar 9.7 Ilustrasi disconnecting node	344
Gambar 9.8 Ilustrasi penelusuran Graf menggunakan prosedur prosedur BFS.....	350
Gambar 9.9 Ilustrasi penelusuran Graf menggunakan prosedur prosedur DFS	351

DAFTAR PUSTAKA

Weiss, M. A. (2014). *Data structure and algorithm analysis in C++ (4th ed.)*. Addison-Wesley.

Das, V. V. (2006). *Principle of data structure using C and C++*. New Age International Publishers.

PROFIL PENULIS



Dr. Untari Novia Wisesty, S.T., M.T., lahir di Bondowoso pada tanggal 8 November 1987. Beliau meraih gelar sarjana dari Program Studi Informatika Universitas Telkom pada tahun 2010, kemudian melanjutkan pendidikan magister di bidang Informatika di Universitas Telkom (2012), serta menuntaskan program doktor di Institut Teknologi Bandung pada tahun 2023.

Sebagai dosen di Fakultas Informatika Universitas Telkom, beliau telah mengampu sejumlah mata kuliah seperti Pembelajaran Mesin, Kecerdasan Artificial, dan Desain Algoritma Lanjut. Dengan pengalaman tersebut, beliau merasa perlu menghadirkan buku ajar ini agar mahasiswa mendapatkan penjelasan yang lebih runtut, lengkap dengan ilustrasi dan contoh penerapan nyata.



Hani Nurrahmi, S.Kom., M.Kom., lahir di Bandung pada tanggal 8 Desember 1992. Beliau meraih gelar sarjana dari Program Studi Informatika Universitas Telkom pada tahun 2016, kemudian melanjutkan pendidikan magister di bidang Informatika di Universitas Telkom tahun 2018. Sebagai dosen di Fakultas Informatika Universitas Telkom, beliau telah mengampu sejumlah mata kuliah seperti Struktur Data, Algoritma dan Pemrograman, dan Teori Bahasa

dan Automata. Buku ajar ini disusun bersama tim dosen sebagai upaya beliau menghadirkan bahan pembelajaran yang sistematis, disertai contoh praktis, agar mahasiswa lebih mudah memahami konsep-konsep penting dalam struktur data.



Prasti Eko Yunanto, ST., M.Kom lahir di Sungai Talang, Kabupaten Solok Selatan, Sumatera Barat tahun 1989. Menyelesaikan pendidikan dasar hingga menengahnya di kota kelahiran. Pendidikan tingginya diselesaikan pada S-1 Teknik Informatika, Institut Teknologi Telkom (2007-2012) dan S-2 Informatika, Universitas Telkom (2013-2015). Sejak tahun 2019 secara resmi mengabdikan sebagai dosen S-

1 Informatika di Universitas Telkom, dengan Mata kuliah yang biasa diampu adalah Algoritma Dasar, Struktur Data, dan Teori Bahasa dan Automata. Selanjutnya untuk fokus penelitian di luar pengajaran adalah Biometrics Security dan juga topik-topik lainnya yang masih berkaitan dengan Kecerdasan Buatan.



Dr. Rita Rismala, S.T., M.T., lahir di Ciamis pada tanggal 15 Desember 1986. Penulis meraih gelar Sarjana Teknik (S.T.) dari Program Studi S1 Teknik Informatika, Fakultas Informatika, Institut Teknologi Telkom (sekarang Universitas Telkom) pada tahun 2010, kemudian meraih gelar Magister Teknik (M.T.) dari Program Studi S2 Teknik Informatika, Fakultas Informatika, Institut

Teknologi Telkom pada tahun 2013, serta meraih gelar Doktor (Dr.) dari Program Studi S3 Teknik Elektro dan Informatika, Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung pada tahun 2024. Penulis merupakan dosen di Fakultas Informatika, Universitas Telkom dan telah mengampu sejumlah mata kuliah seperti Algoritma dan Pemrograman, Struktur Data, Dasar Kecerdasan Artificial, Kecerdasan Artificial, Pembelajaran Mesin, dan Sistem Rekomendasi. Saat ini penulis juga aktif sebagai peneliti di bidang Pembelajaran Mesin dan Sistem Rekomendasi.



Febryanti Sthevanie ST., MT., merupakan akademisi yang lahir pada tahun 1988. Gelar sarjana Informatika diraihnya dari Institut Teknologi Telkom pada tahun 2010, dilanjutkan dengan gelar magister Teknik Informatika di Universitas Telkom pada tahun 2013. Sejak awal karir sebagai dosen di Fakultas Informatika Universitas Telkom, beliau banyak berkontribusi dalam pembelajaran mata kuliah dasar seperti Struktur Data dan Algoritma dan Pemrograman.

Melalui keterlibatan dalam penulisan buku ajar ini, beliau berharap dapat memberikan kemudahan kepada mahasiswa untuk mengimplementasikan struktur data baik di dunia akademik perkuliahan maupun dalam menyelesaikan masalah di dunia kerja kelak.