



DESAIN PEMBELAJARAN ELEMEN BERPIKIR KOMPUTASIONAL TERINTEGRASI ALGORITMA DAN PEMROGRAMAN

UNTUK GURU TINGKAT SMP



Nusuki Syari'ati Fathimah
Andini Setya Arianti
Enjun Junaeti

DESAIN PEMBELAJARAN ELEMEN BERPIKIR KOMPUTASIONAL TERINTEGRASI ALGORITMA DAN PEMROGRAMAN

UNTUK GURU TINGKAT SMP

**Nusuki Syari'ati Fathimah
Andini Setya Arianti
Enjun Junaeti**



**DESAIN PEMBELAJARAN ELEMEN BERPIKIR KOMPUTASIONAL
TERINTEGRASI ALGORITMA DAN PEMROGRAMAN
UNTUK GURU TINGKAT SMP**

Penulis:

**Nusuki Syari'ati Fathimah
Andini Setya Arianti
Enjun Junaeti**

Desain Cover:

Helmaria Ulfa

Sumber Ilustrasi:

www.freepik.com

Tata Letak:

Handarini Rohana

Editor:

Neng Rismawati

ISBN:

978-634-246-892-0

Cetakan Pertama:

Mei, 2026

Hak Cipta Dilindungi Oleh Undang-Undang

by Penerbit Widina Media Utama

Dilarang keras menerjemahkan, memfotokopi, atau memperbanyak sebagian atau seluruh isi buku ini tanpa izin tertulis dari Penerbit.

PENERBIT:

WIDINA MEDIA UTAMA

Komplek Puri Melia Asri Blok C3 No. 17 Desa Bojong Emas
Kec. Solokan Jeruk Kabupaten Bandung, Provinsi Jawa Barat

Anggota IKAPI No. 519/JBA/2025

Website: www.penerbitwidina.com

Instagram: @penerbitwidina

Telepon (022) 87355370

KATA PENGANTAR

Puji syukur ke hadirat Tuhan Yang Maha Esa atas limpahan rahmat dan karunia-Nya sehingga penulisan buku yang berjudul “Desain Pembelajaran Elemen Berpikir Komputasional Terintegrasi Algoritma dan Pemrograman untuk Guru Tingkat SMP” dapat diselesaikan dengan baik.

Buku ini disusun sebagai kontribusi terhadap pengembangan pengetahuan dalam bidang pendidikan informatika. Fokus utama terletak pada upaya merekonstruksi *taught knowledge* yakni pengetahuan yang secara eksplisit dirancang untuk diajarkan berkenaan dengan elemen-elemen berpikir komputasional yang terintegrasi dalam pembelajaran algoritma dan pemrograman. Melalui pendekatan *Didactic Engineering*, Buku ini menawarkan suatu pemetaan konseptual dan kerangka *a priori* sebagai pijakan awal dalam merancang situasi pembelajaran yang bermakna.

Penulis menyadari bahwa transformasi pembelajaran algoritma dan pemrograman tidak dapat hanya bertumpu pada konten materi semata, tetapi memerlukan penataan ulang atas struktur pengetahuan yang hendak diajarkan agar selaras dengan karakteristik kognitif siswa serta tuntutan berpikir abad ke-21. Oleh karena itu, Buku ini diharapkan dapat memperkaya khazanah pemikiran, baik bagi para perancang kurikulum, pendidik, maupun peneliti pendidikan.

Ucapan terima kasih penulis sampaikan kepada berbagai pihak yang telah memberikan dukungan, masukan, dan inspirasi selama proses penyusunan Buku ini. Secara khusus, apresiasi disampaikan kepada rekan sejawat dan komunitas akademik yang telah menjadi mitra diskusi yang produktif.

Akhir kata, penulis menyadari bahwa Buku ini masih memiliki keterbatasan. Oleh karena itu, saran dan kritik membangun sangat diharapkan demi penyempurnaan di masa mendatang. Semoga buku ini bermanfaat dan dapat menjadi bagian dari ikhtiar kolektif dalam memajukan pendidikan informatika di Indonesia.

Penulis

Universitas Pendidikan Indonesia

Bandung, Mei 2026

DAFTAR ISI

KATA PENGANTAR	iii
DAFTAR ISI	iv
BAB 1 KEMAMPUAN BERPIKIR KOMPUTASIONAL	1
BAB 2 KONSEP BERPIKIR KOMPUTASIONAL DAN TEORI PEMBELAJARAN MENDALAM	5
A. Konsep Berpikir Komputasional dalam Pendidikan Informatika	5
B. <i>Taught Knowledge</i> dalam Perspektif Pendidikan	9
C. <i>Didactic Engineering</i> sebagai Pengembangan <i>Taught Knowledge</i> ..	12
D. Pembelajaran Mendalam sebagai Landasan Rekonstruksi <i>Taught Knowledge</i>	14
E. Strategi Pembelajaran	17
BAB 3 MENGINTEGRASIKAN <i>COMPUTATIONAL THINKING</i> DALAM DESAIN PEMBELAJARAN INFORMATIKA	21
A. Kajian Capaian Pembelajaran (CP) dan Struktur Konsep	21
B. Penempatan Elemen <i>Computational Thinking</i> (CT)	31
C. Representasi: Visual, Verbal, Kode Program	32
D. Analogi dan Konteks Otentik	34
E. Memetakan Konsep dan Hambatan Belajar Siswa	36
F. Rancangan Alur Tujuan Pembelajaran dan Elemen CT yang Terlibat	41
BAB 4 MENGIDENTIFIKASI KESULITAN BELAJAR	45
A. Pengembangan Indikator Kesulitan Belajar	46
B. Mengidentifikasi Kesulitan Belajar Algoritma dan Pemrograman Berdasarkan Persepsi Guru	55
C. Dampak Analisis terhadap Cara Materi Diajarkan	62
BAB 5 RANCANGAN SITUASI PEMBELAJARAN BERBASIS <i>DIDACTIC ENGINEERING</i>	67
A. Rancangan Aktivitas Pembelajaran dan Pemetaan Elemen CT	67
B. Kerangka Praksiologi dalam Situasi Didaktis	78
C. Rancangan Situasi Didaktis untuk Struktur Kontrol Percabangan ...	86
D. Rancangan Situasi Didaktis untuk Pengenalan (Definisi) Algoritma	91
E. Instrumen Validasi Rencana Situasi Didaktis	92
DAFTAR PUSTAKA	95
LAMPIRAN	101
Lampiran 1 Contoh Lembar Kerja Peserta Didik	101
Lampiran 2 Rubrik Penilaian Presentasi	104

Lampiran 3 Instrumen Evaluasi Profil Lulusan	106
Lampiran 4 Perencanaan Pembelajaran Definisi Algoritma	108
Lampiran 5 Instrumen Validasi Rencana Pembelajaran	131
PROFIL PENULIS	146

1

KEMAMPUAN BERPIKIR KOMPUTASIONAL

Berpikir Komputasional (*Computational Thinking-CT*) telah berkembang menjadi salah satu kompetensi esensial abad ke-21. Dalam konteks pendidikan, CT tidak hanya dianggap sebagai landasan untuk pembelajaran informatika dan ilmu komputer, tetapi juga sebagai kerangka berpikir yang dapat memperkuat kemampuan pemecahan masalah dan berpikir logis pada siswa (Cansu & Cansu, 2019; Lyon & Magana, 2020). Sejalan dengan arah reformasi pendidikan global, kurikulum di Indonesia juga mulai mengintegrasikan materi algoritma dan pemrograman dalam pembelajaran di tingkat Sekolah Menengah Pertama (SMP).

Namun, di balik kemajuan kebijakan tersebut, realitas di lapangan menunjukkan bahwa pembelajaran algoritma dan pemrograman masih menghadapi banyak tantangan. Siswa kerap mengalami kesulitan dalam memahami konsep-konsep dasar pemrograman secara mendalam. Tantangan ini terutama bersumber dari tiga faktor utama: (1) tingginya derajat abstraksi materi; (2) kurangnya representasi kontekstual dalam pembelajaran; dan (3) keterbatasan pemahaman guru terhadap implementasi CT dalam aktivitas belajar mengajar yang bermakna.

Pertama, tingkat abstraksi yang tinggi dalam konsep-konsep pemrograman seperti penggunaan variabel, struktur kontrol, dan algoritma sering menjadi hambatan utama dalam proses internalisasi prinsip-prinsip CT oleh siswa (Kakavas & Ugolini, 2019). Kedua, minimnya representasi kontekstual yang relevan dengan dunia nyata membuat siswa kesulitan menghubungkan materi dengan pengalaman mereka sendiri, sehingga menurunkan efektivitas pembelajaran. Ketiga, banyak guru belum memiliki pelatihan yang memadai dalam mengintegrasikan CT secara efektif ke dalam praktik pedagogis, yang pada akhirnya mempengaruhi kualitas pengalaman belajar siswa.

Berbagai strategi telah diajukan untuk mengatasi masalah-masalah tersebut. Di antaranya adalah peningkatan kapasitas guru melalui pelatihan yang lebih sistematis (Lockwood & Mooney, 2017), penggunaan pendekatan pembelajaran berbasis masalah dan konteks dunia nyata (Jonasen & Gram-Hansen, 2019; Bai & Zhao, 2021), serta pemanfaatan teknologi inovatif seperti robotika dan *augmented reality* guna

2

KONSEP BERPIKIR KOMPUTASIONAL DAN TEORI PEMBELAJARAN MENDALAM

A. KONSEP BERPIKIR KOMPUTASIONAL DALAM PENDIDIKAN INFORMATIKA

Berpikir komputasional (*computational thinking*, CT) merupakan proses berpikir yang melibatkan formulasi masalah dan penyusunan solusi yang dapat dieksekusi oleh agen komputasi, baik manusia maupun mesin (Wing, 2006). Dalam konteks pendidikan, CT menjadi kerangka berpikir penting untuk menumbuhkan kemampuan *problem-solving*, penalaran logis, dan pengembangan algoritma pada peserta didik (Grover & Pea, 2013). Bahkan, beberapa kajian menekankan bahwa CT perlu dipahami sebagai kerangka berpikir yang menyeluruh, bukan hanya seperangkat keterampilan teknis. Ini menjadikan CT sebagai kompetensi kunci yang perlu dikembangkan sejak pendidikan dasar hingga menengah (Voogt *et al.*, 2015; Wing, 2006).

CT tidak hanya relevan untuk pembelajaran informatika, tetapi juga untuk mendukung kompetensi lintas bidang, seperti matematika, sains, dan bahkan humaniora. Integrasi CT ke dalam pembelajaran terbukti mampu mendorong siswa untuk berpikir secara logis, analitis, dan kreatif. Dalam bidang pendidikan STEM, penerapan CT telah meningkatkan performa siswa dalam menyelesaikan masalah non rutin serta memperkuat kemampuan berpikir tingkat tinggi (Selamat *et al.*, 2024; Srinivasa, 2022; Asunda *et al.*, 2023). Di Indonesia, elemen CT telah termuat dalam capaian pembelajaran mata pelajaran Informatika pada Kurikulum, khususnya terintegrasi pada materi algoritma dan pemrograman dan analisis data.

1. Elemen CT: *Decomposition, Pattern Recognition, Abstraction, Algorithmic Thinking*

CT mencakup empat elemen utama yang saling berhubungan, yaitu dekomposisi, pengenalan pola, abstraksi, dan berpikir algoritmik. Keempat elemen ini tidak hanya relevan dalam konteks pemrograman atau pengembangan perangkat lunak, tetapi juga telah diakui sebagai kompetensi lintas disiplin. Empat elemen utama berpikir komputasional menurut banyak literatur adalah sebagai berikut.

3

MENGINTEGRASIKAN *COMPUTATIONAL THINKING* DALAM DESAIN PEMBELAJARAN INFORMATIKA

Analisis konseptual dan epistemologis dilakukan untuk menelaah secara mendalam struktur pengetahuan yang melandasi materi algoritma dan berpikir komputasional (*Computational Thinking/CT*) yang akan diajarkan kepada siswa SMP. Langkah ini menjadi bagian penting dalam kerangka *Didactic Engineering* karena berfungsi sebagai landasan dalam proses transposisi didaktik, yaitu mengubah pengetahuan ilmiah (*scholarly knowledge*) menjadi pengetahuan yang siap diajarkan di kelas (*taught knowledge*).

Melalui analisis ini, pendidik dapat memahami bagaimana konsep-konsep utama dalam algoritma dan CT tersusun serta bagaimana konsep tersebut dapat disajikan secara lebih mudah dipahami oleh siswa. Selain itu, analisis ini juga membantu mengidentifikasi kemungkinan munculnya *epistemological obstacles* atau hambatan epistemologis yang dapat dialami siswa ketika mempelajari konsep-konsep yang bersifat abstrak dan teknis, sehingga strategi pembelajaran yang dirancang dapat lebih tepat dan efektif.

A. KAJIAN CAPAIAN PEMBELAJARAN (CP) DAN STRUKTUR KONSEP

Pembelajaran algoritma dan pemrograman pada mata pelajaran Informatika Fase D merupakan bagian integral dari pengembangan elemen berpikir komputasional (*computational thinking*) yang telah secara eksplisit diintegrasikan dalam struktur capaian pembelajaran Informatika. Dalam CP Informatika untuk fase D, elemen berpikir komputasional diarahkan agar peserta didik mampu:

1. memahami konsep himpunan data terstruktur dalam kehidupan sehari-hari
2. memahami konsep lembar kerja pengolah data
3. menerapkan berpikir komputasional dalam menyelesaikan persoalan yang mengandung himpunan data berstruktur sederhana dengan volume kecil

4

MENGIDENTIFIKASI KESULITAN BELAJAR

Pemrograman kini semakin diintegrasikan ke dalam kurikulum pendidikan dasar dan menengah, baik di tingkat internasional maupun nasional, sebagai elemen kunci dari literasi digital abad ke-21 (Huzooree & Soupramanien, 2024; Rodrigues *et al.*, 2024; Woo & Falloon, 2024; Zeeshan *et al.*, 2024; Arodani *et al.*, 2025). Secara spesifik, di jenjang Sekolah Menengah Pertama (SMP), pemrograman berfungsi tidak hanya sebagai bagian dari mata pelajaran Informatika, tetapi juga berperan sebagai alat esensial untuk mengasah keterampilan berpikir kritis, kreatif, dan komputasional siswa (Liu *et al.*, 2023; Almdahem, 2024; Altıok & Üçgül, 2025; Mills *et al.*, 2024; Sarro *et al.*, 2024; Yermaganbetova *et al.*, 2024).

Meskipun pentingnya diakui, berbagai penelitian secara konsisten menyoroti bahwa siswa, mulai dari tingkat sekolah dasar (SD) hingga mahasiswa, mengalami berbagai kesulitan dalam proses pembelajaran pemrograman (Almdahem, 2024; Giannakoulas & Xinogalos, 2024; Kadar, 2021; Waite, 2020; Islam, 2019; Özmen, 2014). Kesulitan ini bersifat multidimensi, mencakup:

1. **Pemahaman Konsep Dasar dan Algoritma:** Banyak siswa menghadapi tantangan dalam memahami dasar-dasar algoritma (Kadar, 2021; Islam, 2019; Andrzejewska, 2018; Baist, 2017; Lahtinen, 2005) dan istilah konseptual seperti yang diidentifikasi oleh Waite (2020) pada guru SD.
2. **Keterampilan Teknis dan Logika:** Kesulitan muncul dalam mentransfer algoritma menjadi kode program, menulis kode yang benar secara sintaksis maupun logis (Kadar, 2021; Özmen, 2014; Mow, 2008), serta dalam penggunaan struktur logika pemrograman seperti parameter dan fungsi (Özmen, 2014).
3. **Pemecahan Masalah dan *Debugging*:** Siswa kesulitan dalam mengevaluasi efektivitas solusi, melakukan pengujian, dan *debugging* secara sistematis (Kadar, 2021; Islam, 2019; Andrzejewska, 2018; Baist, 2017; Özmen, 2014).

Selain kesulitan siswa, Waite (2020) juga mengidentifikasi hambatan implementasi yang dihadapi guru SD, termasuk resistensi siswa, keterbatasan waktu, dan kurangnya pemahaman konseptual.

5

RANCANGAN SITUASI PEMBELAJARAN BERBASIS *DIDACTIC ENGINEERING*

Bab ini memaparkan rancangan awal situasi pembelajaran yang disusun berdasarkan kajian *a priori* terhadap materi yang akan diajarkan (*taught knowledge*) yang telah direkonstruksi sebelumnya. Rancangan ini dikembangkan dengan menggunakan pendekatan *Didactic Engineering* yang bertujuan mengintegrasikan unsur berpikir komputasional dalam pembelajaran algoritma dan pemrograman secara lebih kontekstual, bermakna, dan mendalam. Seluruh desain pembelajaran dirancang dengan mempertimbangkan kemungkinan respons siswa selama proses belajar berlangsung. Dengan demikian, guru dapat menyiapkan berbagai strategi dan intervensi pedagogis yang tepat untuk membantu siswa memahami konsep secara lebih efektif.

A. RANCANGAN AKTIVITAS PEMBELAJARAN DAN PEMETAAN ELEMEN CT

Rancangan aktivitas pembelajaran dikembangkan dalam bentuk matriks integratif yang mengaitkan: (1) tujuan pembelajaran, (2) aktivitas yang dirancang, (3) elemen CT yang dikembangkan, serta (4) dimensi profil pelajar Pancasila yang ditargetkan. Pendekatan ini memungkinkan guru untuk melacak kontribusi setiap aktivitas terhadap pengembangan kemampuan berpikir komputasional dan karakter siswa. Lihat Tabel 5.1: Matriks Rancangan Aktivitas dan Elemen CT.

DAFTAR PUSTAKA

- Almanasreh, E., Moles, R., & Chen, T. F. (2019). Evaluation of methods used for estimating content validity. *Research in Social and Administrative Pharmacy*, 15(2), 214-221.
- Almdahem, A. A. (2024). Exploring the difficulties of learning programming among computer science students in Saudi higher education. *Education and Information Technologies*, 29(4), 4785-4809.
- Altıok, S., & Üçgül, M. (2025). Investigating the relationship between middle school students' *computational thinking* skills and their reflective *thinking* skills for problem solving. *Education and Information Technologies*, 30(1), 743-772.
- Andrzejewska, M. (2018). Difficulties in learning programming based on students' opinions. *Informatics in Education*, 17(1), 1-16.
- Arodani, D. A., Setyosari, P., Praherdhiono, H., & Kuswandi, D. (2025). Integrating *computational thinking* in the primary school curriculum: A systematic literature review. *Heliyon*, 11(1), e34164.
- Artigue, M. (2009). *Didactical design in mathematics education*. In ICME-11 Survey Team Reports.
- Artigue, M. (2014). *Didactic engineering in mathematics education*. Dalam S. Lerman (Ed.), *Encyclopedia of Mathematics Education* (hlm. 159–162). Springer.
- Asunda, P., Faezipour, M., Tolemy, J., & Engel, M. T. D. (2023). Embracing *computational thinking* as an impetus for artificial intelligence in integrated STEM disciplines through *engineering* and technology education. *Journal of Technology Education*, 34(2), 43–63.
- Bai, H., Wang, X., & Zhao, L. (2021). Effects of the problem-oriented learning model on middle school students' *computational thinking* skills in a Python course. *Frontiers in Psychology*, 12, 771221.
- Baist, A. (2017). Learning difficulties in programming for beginners. *Journal of Physics: Conference Series*, 895, 012128.
- Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of *computational thinking*. Proceedings of the AERA.
- Brousseau, G. (1997). *Theory of didactical situations in mathematics*. Springer.

- Cansu, F. K., & Cansu, S. K. (2019). An overview of *computational thinking*. International Journal of Computer Science Education in Schools, 3(1), 17-30.
- Chevallard, Y. (1985). La transposition didactique. Grenoble: La Pensée Sauvage.
- Chevallard, Y. (1991). La transposition didactique: Du savoir savant au savoir enseigné (2nd ed.). La Pensée Sauvage.
- Chevallard, Y. (2014). [Catatan: Sitasi ini kemungkinan merujuk pada karya bersama. Karya yang paling sering dikutip dari periode ini adalah: Chevallard, Y., & Bosch, M. (2014). *Didactic* transposition in mathematics education. Dalam S. Lerman (Ed.), Encyclopedia of Mathematics Education. Springer.]
- Chevallard, Y., & Bosch, M. (2020). Anthropological theory of the *didactic* (ATD). Encyclopedia of mathematics education, 53–61.
- Chevallard, Y., & Bosch, M. (2020). *Didactic* Transposition in Mathematics Education. Dalam S. Lerman (Ed.), Encyclopedia of Mathematics Education (hlm. 264–269). Springer International Publishing.
- Cifredo-Chacón, M. A., Figueiredo, M., & Gonçalves, V. (2016). Learning *Computational Thinking* with Robotics in Middle School Assisted with Augmented Reality and Mobile Devices. INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN MULTIDISCIPLINARY EDUCATION, 3(9), 1569–1581.
- Deviana, T., Mn, F. R., Sulistyani, N., & Maulyda, M. A. (2024). DI SEKOLAH DASAR Pendidikan Guru Sekolah Dasar, Universitas Negeri Mataram, NTB, Indonesia E-mail: Abstrak PENDAHULUAN Implementasi kurikulum Merdeka pada jenjang Sekolah Dasar (SD) dilakukan secara bertahap, sesuai kesiapan sekolah. Kurikulum mer. 13(4), 1269–1280.
- El Fadel, H. (2024). The *Didactic* Transposition: Practices and Pedagogical Issues. International Journal of Innovative Research in Multidisciplinary Education, 3(09).
- Fadel, Hanane El. (2024). The *Didactic* Transposition: Practices and Pedagogical Issues. In Proceedings of the 18th European Conference on e-Learning: ECEL (pp. 260–7).
- Forsey, M. (2010). *Taught knowledge*: A sociological perspective. Teaching in Higher Education, 15(5), 501–512.
- Giannakoulas, A., & Xinogalos, S. (2024). A systematic literature review on the difficulties and the common errors of novice programmers. *Education and Information Technologies*, 29(1), 1-52.

- Grover, S., & Pea, R. (2013). *Computational thinking* in K–12. *Educational Researcher*, 42(1), 38–43.
- Hamsyah, H. (2021). Proses Validasi Instrumen dengan Pendekatan *Expert Judgment*. *Jurnal Sinestesia*, 11(2), 127-133.
- Huzooree, G., & Soupramanien, G. (2024). A Systematic Literature Review of *Computational Thinking* in Primary Education: The Transition from Unplugged to Plugged Activities. *Education Sciences*, 14(4), 365.
- Islam, M. S. (2019). Common Difficulties in Learning Programming: A Study on Students of Computer Science. *International Journal of Computer Applications*, 178(41), 1-7.
- Jonasen, T. S., & Gram-Hansen, S. B. (2019). Problem based learning: A facilitator of *computational thinking*. In *Proceedings of the 18th European Conference on e-Learning: ECEL* (pp. 260–7).
- Kadar, I. A. (2021). A systematic literature review of students' difficulties in learning computer programming. *International Journal of Information and Education Technology*, 11(11), 536-543.
- Kakavas, P., & Ugolini, F. C. (2019). *Computational thinking* in primary education: A systematic literature review. *Research on Education and Media*, 11(2), 64–94.
- Kang, J. (2024). AI-based adaptive learning: A systematic review of the literature. *Computers & Education*, 216, 105021.
- Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi (Kemdikbudristek). (2022). *Panduan Implementasi Profil Pelajar Pancasila*.
- Lahtinen, E., Ala-Mutka, K., & Järvinen, H. M. (2005). A study of the difficulties of novice programmers. *ACM SIGCSE Bulletin*, 37(3), 14-18.
- Lampert, M. (1990). When the problem is not the question and the solution is not the answer. *American Educational Research Journal*, 27(1), 29–63.
- Liu, Z., Wang, S., Wang, Z., & Wang, Q. (2023). The effect of programming education on the *computational thinking* skills of middle school students: A meta-analysis. *Computers & Education*, 205, 104886.
- Lockwood, J., & Mooney, A. (2017). A pilot study investigating the introduction of a computer-science course at second level focusing on *computational thinking*. *The Irish Journal of Education/Iris Eireannach an Oideachais*, 42, 108–127.
- Lockwood, J., & Mooney, A. (2017). *Computational Thinking* in Education: Where does it Fit? A systematic literary review. (arXiv:1703.07659 [cs.CY]).

- Lynn, M. R. (1986). Determination and quantification of content validity. *Nursing Research*, 35(6), 382-385.
- Lyon, J. A., & Magana, A. J. (2020). *Computational thinking* in higher education: A review of the literature. *Computer Applications in Engineering Education*, 28(5), 1174–1189.
- Marselina, W. (2021). Expert Judgment in Educational Research: A Systematic Review. *Jurnal Pendidikan: Teori, Penelitian, dan Pengembangan*, 6(10), 1475-1481.
- Marton, F., & Säljö, R. (2005). Approaches to learning. In *The Experience of Learning*.
- Masuwai, A., Tajjudin, M., & dkk. (2024). Content validity of an instrument measuring digital leadership practices. *International Journal of Learning, Teaching and Educational Research*.
- Maton, K. (2013). Making semantic waves: A key to cumulative knowledge-building. *Linguistics and Education*, 24(1), 8–22.
- Maton, K. (2014). *Knowledge and knowers: Towards a realist sociology of education*. Routledge.
- Mills, K. A., Gynther, K., & dkk. (2024). *Computational thinking* in middle school: A systematic review of the literature. *Education and Information Technologies*, 29(2), 1-32.
- Mow, C. (2008). Common errors and difficulties in learning to program. *Proceedings of the 9th LTSN-ICS Annual Conference*.
- Özmen, E. (2014). Difficulties in learning programming: A case study of computer engineering students. *International Journal of Education in Mathematics, Science and Technology*, 2(1), 1-13.
- Perrin-Glorian, M. J. (2011). Didactique des mathématiques et ingénierie didactique. *Recherches en didactique des mathématiques*, 31(3), 297–320.
- Plomp, T., & Nieveen, N. (2007). An introduction to educational design research. SLO-Netherlands Institute for Curriculum Development.
- Polit, D. F., & Beck, C. T. (2006). The content validity index: Are you sure you know what's being reported? Critique and recommendations. *Research in Nursing & Health*, 29(5), 489-497.
- Rodrigues, M., Hubwieser, H., & dkk. (2024). *Computational thinking* in primary and secondary education: A systematic mapping study. *ACM Transactions on Computing Education*.
- Sarro, C., Foggia, P., & dkk. (2024). Teaching programming in middle schools: A systematic literature review. *Computers & Education*, 210, 104958.

- Selamat, M., Sobihatun, N., & Mustapa, N. (2024). Investigation of *Computational Thinking* Skills through Instructional Techniques, Games and Programming Tools. *International Journal of Learning, Teaching and Educational Research*, 23(4), 185-202.
- Sentence, S., Waite, J., & Kallia, M. (2019). Teaching computer programming with PRIMM: A sociocultural perspective. *Computer Science Education*, 29(2-3), 136-176.
- Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying *computational thinking*. *Educational Research Review*, 22, 142-158.
- Singh, B., & Kaunert, C. (2024). *Computational Thinking* for Innovative Solutions and Problem-Solving Techniques: Transforming Conventional Education to Futuristic Interdisciplinary Higher Education. Dalam *Revolutionizing Curricula Through Computational Thinking, Logic, and Problem Solving* (hlm. 60-82). IGI Global.
- Srinivasa, K. G., Shobha, G., & Shwetha, R. (2022). *Computational Thinking*. Dalam *Computational Thinking for Problem Solving and Managerial Decision Making* (hlm. 1-20). CRC Press.
- Tykhonova, T., & Koshkina, V. (2018). *A Framework for Decomposition in Computational Thinking*. Makalah dipresentasikan pada 1st International Workshop on *Computational Thinking* Education, Hong Kong.
- Voogt, J., Fisser, P., Good, J., Mishra, P., & Yadav, A. (2015). *Computational thinking* in compulsory education: A new conceptual framework. *International Conference on Exploring the Material Conditions of Learning (EXS)*.
- Waite, J. (2020). *Analysing the curriculum, pedagogy and practice of primary programming*. [Disertasi Doktor, King's College London].
- Waite, J., Maton, K., Curzon, P., & Tuttiett, L. (2019). *Unplugged Computing and Semantic Waves: Analysing Crazy Characters*. Makalah dipresentasikan pada UKICER2019 Conference, Canterbury, Kent, UK.
- Wiggins, G. (1998). *Educative assessment: Designing assessments to inform and improve student performance*. Jossey-Bass.
- Wing, J. M. (2006). *Computational thinking*. *Communications of the ACM*, 49(3), 33-35.
- Woo, H., & Falloon, G. (2024). A systematic review of K-12 *computational thinking* education in Australasia. *Education and Information Technologies*, 29(3), 1-34.

- Yadav, A., Mayfield, C., Zhou, N., Hambruch, S., & Korb, J. T. (2014). *Computational thinking* in elementary and secondary teacher education. *ACM Transactions on Computing Education (TOCE)*, 14(1), 1–16.
- Yadav, A., Stephenson, C., & Hong, H. (2017). *Computational thinking* for all. *ACM Inroads*, 8(4), 35–37.
- Yelland, N., Gilbert, R., Kilderry, A., & Ainley, J. (2020). *Semantic waves in curriculum and pedagogy*. Routledge.
- Yermaganbetova, M., Tursunbayeva, A., & dkk. (2024). Developing *computational thinking* in middle school students through programming: A systematic review. *Heliyon*, 10(7), e28641.
- Zeeshan, M., Jamil, A., & dkk. (2024). Integrating *computational thinking* into primary education: A systematic literature review. *Journal of Educational Computing Research*.

PROFIL PENULIS

Nusuki Syari'ati Fathimah, M.Pd.



Penulis adalah dosen pada Program Studi Pendidikan Ilmu Komputer, Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam (FPMIPA) Universitas Pendidikan Indonesia. Penulis memperoleh gelar Sarjana dari Universitas Pendidikan Indonesia pada Program Studi Pendidikan Ilmu Komputer (2012) dan gelar magister dari Universitas Negeri Yogyakarta pada Program Studi Teknologi Pembelajaran (2017). Saat ini penulis mengampu mata kuliah Strategi Pembelajaran, Perencanaan Pembelajaran, Kapita Selekta, dan Etika Profesi.

Andini Setya Arianti, M.Ds.



Penulis adalah staf pengajar pada Program Studi Pendidikan Ilmu Komputer, Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam (FPMIPA) Universitas Pendidikan Indonesia. Penulis memperoleh gelar Sarjana dari Telkom University pada tahun 2016, dan gelar magister dari Universitas Komputer Indonesia pada tahun 2019. Saat ini penulis mengampu mata kuliah Animasi, Interaksi Manusia dan Komputer, Teknologi *Motion Graphic* dan Multimedia.

Enjun Junaeti, M.Si.



Penulis adalah staf pengajar pada Program Studi Pendidikan Ilmu Komputer, Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam (FPMIPA) Universitas Pendidikan Indonesia. Penulis memperoleh gelar Sarjana dari Universitas Pendidikan Indonesia pada tahun 2008, dan gelar magister dari Institut Teknologi Bandung pada tahun 2011. Penulis mengampu berbagai mata kuliah, seperti Kalkulus, Matematika Informatika, Statistika, Teori Bilangan, Aljabar Linier dan Matriks, Matematika Diskrit, Metode Numerik, Matematika Terapan, Matematika Bisnis, Teknik Riset Operasi, serta Teknik Simulasi dan Pemodelan.

DESAIN PEMBELAJARAN ELEMEN BERPIKIR KOMPUTASIONAL TERINTEGRASI ALGORITMA DAN PEMROGRAMAN

UNTUK GURU TINGKAT SMP

Buku ini mengajak pembaca memahami bagaimana pembelajaran algoritma dan pemrograman dapat disajikan secara lebih bermakna bagi siswa SMP. Melalui pendekatan yang menekankan berpikir komputasional (*computational thinking*), buku ini menjelaskan bagaimana siswa dapat belajar memecahkan masalah secara sistematis melalui langkah-langkah seperti menguraikan masalah, mengenali pola, membuat abstraksi, dan menyusun algoritma. Konsep-konsep tersebut diperkenalkan secara bertahap dengan berbagai contoh dan representasi yang dekat dengan pengalaman belajar siswa, sehingga membantu mereka memahami logika pemrograman secara lebih mendalam dan tidak sekadar menghafal langkah-langkah teknis.

Selain membahas konsep, buku ini juga memberikan gambaran praktis tentang bagaimana merancang pembelajaran algoritma dan pemrograman yang menarik dan mudah dipahami. Pembahasan mencakup cara memperkenalkan konsep dasar algoritma, hingga memahami struktur kontrol percabangan seperti *if-else* dalam pemrograman. Dengan pendekatan yang sistematis namun tetap sederhana, buku ini diharapkan dapat menjadi referensi bagi guru, mahasiswa, maupun siapa saja yang ingin mengembangkan pembelajaran informatika yang lebih kreatif, kontekstual, dan berorientasi pada pemahaman konsep.



SCANME

www.penerbitwidina.com
@penerbitwidina
penerbit widina
penerbitwidina@gmail.com
widina store
widina bookstore

Layanan Print-on-Demand & Pengiriman Buku
0815-7000-699

Pendidikan

ISBN 978-634-246-892-0



9

786342

468920