

Pengembangan dan Validasi Isi Indikator Kesulitan Belajar Pemrograman Siswa SMP

Nusuki Syari'ati
Fathimah¹;
Enjun Junaeti²;
Andini Setia Aryanti³;
Erlangga⁴

^{1,2,3,4}Universitas
Pendidikan Indonesia

Alamat
Korespondensi
enjun@upi.edu

ABSTRAK

Tujuan dari penelitian ini adalah untuk mengidentifikasi dan memformulasikan indikator-indikator kesulitan belajar siswa Sekolah Menengah Pertama (SMP) dalam konteks pembelajaran pemrograman. Instrumen yang dikembangkan terdiri dari 23 indikator yang dikelompokkan ke dalam tiga elemen utama: pemahaman konsep dasar algoritma, keterampilan pemrograman, dan berpikir komputasional dalam tahapan problem solving. Validasi isi dilakukan oleh enam ahli yang memiliki latar belakang di bidang pendidikan dan informatika, dengan menggunakan skala penilaian 4 poin. Data dianalisis menggunakan pendekatan Item-Level Content Validity Index (I-CVI) dan koefisien kappa modifikasi (κ^*) untuk mengontrol kemungkinan kesepakatan yang terjadi secara kebetulan. Hasil menunjukkan bahwa seluruh item memiliki nilai I-CVI $\geq 0,83$ dan κ^* berkisar antara 0,82 hingga 1,00. Nilai Average-CVI sebesar 0,91 mengindikasikan bahwa instrumen memiliki validitas isi yang sangat baik. Berdasarkan saran dari para ahli, instrumen direvisi menjadi 43 butir indikator dengan nilai I-CVI dan κ^* sebesar 1,00. Instrumen ini dapat dikembangkan lebih lanjut menjadi survei persepsi guru terhadap kesulitan belajar siswa atau menjadi dasar penyusunan tes diagnostik yang ditujukan langsung kepada siswa.

Kata Kunci instrumen kesulitan belajar, pemrograman, validitas isi, I-CVI, kappa, siswa SMP.

1. Pendahuluan

Beberapa kurikulum internasional maupun nasional telah mengintegrasikan pemrograman ke dalam pendidikan dasar untuk mencapai kecakapan literasi digital abad ke-21 (Arodani et al., 2025; Huzooree & Soupramanien, 2024; Rodrigues et al., 2024; Woo & Falloon, 2024; Zeeshan et al., 2024). Di tingkat Sekolah Menengah Pertama (SMP), pemrograman tidak hanya diposisikan sebagai bagian dari kurikulum Informatika, tetapi juga sebagai sarana untuk mengembangkan keterampilan berpikir komputasional (Almdahem, 2024; Altrok & Üçgül, 2025; Liu et al., 2023; Mills et al., 2024; Sarro et al., 2024; Yermaganbetova et al., 2024). Namun demikian, berbagai studi menunjukkan bahwa baik siswa SD maupun mahasiswa masih menghadapi beragam kesulitan dalam proses pembelajaran pemrograman, baik dalam aspek pemahaman konsep dasar, keterampilan teknis, maupun strategi pemecahan masalah (Almdahem, 2024; Giannakoulas & Xinogalos, 2024; Islam et al., 2019; Kadar et al., 2021; Özmen & Altun, 2014; Waite et al., 2020), sehingga perlu dilakukan deteksi kesulitan belajar lebih dini.

Penelitian Waite (2020) mengidentifikasi bahwa guru menghadapi hambatan dalam mengimplementasikan desain dalam pembelajaran pemrograman untuk siswa SD kelas 1-5 seperti resistensi siswa, keterbatasan waktu, dan kurangnya pemahaman konseptual terhadap istilah seperti algoritma. Özmen (2014) juga melaporkan bahwa mahasiswa mengalami kesulitan dalam pengetahuan dan keterampilan pemrograman, serta dalam penggunaan struktur logika seperti parameter dan fungsi. Sedangkan Kadar (2021), melalui kajian sistematik literatur (SLR) dari tahun 2010 hingga 2019, merangkum berbagai kesulitan siswa dalam membangun algoritma, mentransfer algoritma ke dalam kode program, memahami struktur pemrograman, modularisasi, serta dalam proses pengujian dan debugging.

Studi-studi tersebut menunjukkan bahwa kesulitan belajar dalam pemrograman bersifat multidimensi dan dapat muncul dari berbagai faktor, mulai dari kurangnya pemahaman

dasar algoritma (Baist & Pamungkas, 2017; Islam et al., 2019; Kadar et al., 2021; Lahtinen et al., 2005), tantangan dalam menulis kode yang benar secara sintaksis maupun logis (Chan Mow, 2008; Kadar et al., 2021; Özmen & Altun, 2014), hingga kesulitan dalam mengevaluasi efektivitas solusi dan melakukan debugging secara sistematis (Baist & Pamungkas, 2017; Islam et al., 2019; Kadar et al., 2021; Özmen & Altun, 2014). Selain itu, keterampilan berpikir komputasional—seperti dekomposisi, pengenalan pola, dan abstraksi—yang menjadi dasar dalam menyelesaikan masalah secara sistematis, juga menjadi tantangan tersendiri bagi siswa (Altiok & Üçgül, 2025; Liu et al., 2023; Mills et al., 2024).

Meskipun sejumlah penelitian telah mengkaji hambatan dalam pembelajaran pemrograman, baik di tingkat sekolah maupun perguruan tinggi (Almdahem, 2024; Giannakoulas & Xinogalos, 2024; Islam et al., 2019; Kadar et al., 2021; Özmen & Altun, 2014; Waite et al., 2020), namun instrumen yang secara khusus dirancang untuk mengidentifikasi kesulitan belajar siswa di tingkat SMP secara sistematis dan terstruktur masih terbatas. Sebagian besar penelitian yang dilakukan lebih berfokus pada persepsi siswa atau performa dalam tes akademik, tetapi belum secara umum mengidentifikasi indikator-indikator kesulitan belajar yang dapat memetakan jenis dan indikator-indikator kesulitan siswa secara komprehensif (Kadar et al., 2021; Özmen & Altun, 2014), sehingga dapat digunakan sebagai acuan oleh peneliti lain maupun guru. Padahal, Instrumen diagnostik yang memiliki validitas dan keandalan tinggi memegang peran krusial dalam mendukung guru mengidentifikasi sumber kesulitan belajar siswa, sehingga memungkinkan perencanaan strategi pembelajaran yang lebih adaptif dan sesuai dengan kebutuhan masing-masing individu (Deviana et al., 2024).

Berdasarkan latar belakang tersebut, tujuan dari penelitian ini adalah untuk mengembangkan dan mengevaluasi validitas isi dari sebuah instrumen yang dirancang khusus untuk mengidentifikasi kesulitan belajar siswa SMP dalam pembelajaran pemrograman. Instrumen ini mencakup tiga domain utama, yaitu: (1) pemahaman konsep dasar algoritma, (2) keterampilan teknis dalam menulis dan menelaah kode pemrograman, serta (3) keterampilan berpikir komputasional dalam menyelesaikan masalah secara sistematis. Validitas isi dievaluasi melalui penilaian oleh para ahli untuk memastikan bahwa setiap item dalam instrumen relevan, representatif, dan sesuai dengan tujuan diagnostik.

2. Metode

Penelitian ini bertujuan mengembangkan instrumen untuk mengidentifikasi indikator kesulitan belajar pemrograman siswa SMP serta mengevaluasi validitas isinya. Penelitian menggunakan pendekatan kuantitatif deskriptif melalui teknik expert judgment (Marselina et al., 2021), yang bertujuan menggambarkan karakteristik instrumen berdasarkan penilaian numerik dari para ahli. Validitas isi penting untuk memastikan setiap item mewakili konstruk yang ingin diukur (Almanasreh et al., 2019; Masuwai et al., 2024).

Data penelitian berupa skor penilaian ahli menggunakan skala Likert 4 poin (Connelly, 2022), terhadap empat aspek yang diadaptasi dari literatur, yaitu: kelengkapan indikator terkait sejauh mana unsur-unsur penting yang relevan dengan elemen kesulitan belajar tertentu (Almanasreh et al., 2019; Masuwai et al., 2024), kejelasan redaksi indikator, relevansi indikator terhadap konteks pembelajaran pemrograman di SMP, serta konsistensi indikator dengan domain/subdomain kesulitan belajar (Hamsyah et al., 2020). Validasi dilakukan oleh enam ahli yang memiliki kompetensi di bidang pendidikan informatika, terdiri atas dosen yang juga merupakan instruktur nasional bimbingan teknis guru Informatika maupun penelaah buku Informatika. Pemilihan ahli dilakukan secara purposif dengan mempertimbangkan latar belakang akademik dan pengalaman mengajar (Masuwai et al., 2024).

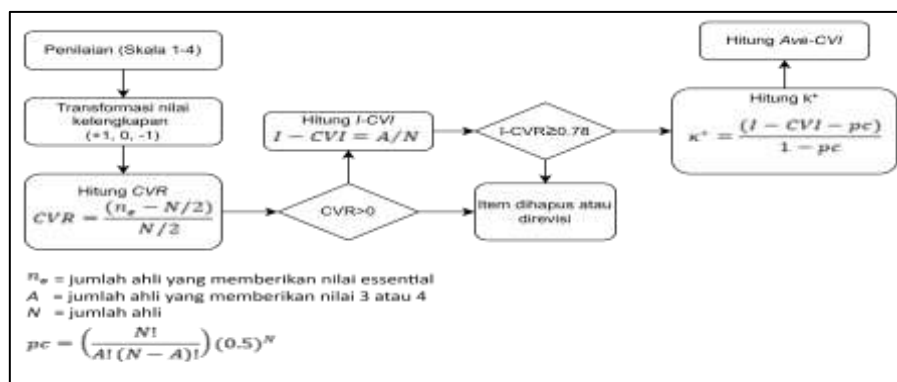
Analisis data dilakukan dalam empat tahap utama sesuai pendekatan validasi isi kuantitatif yang diadaptasi dari Almanasreh et al. (2019) berdasarkan teori yang dikembangkan dalam literatur sebelumnya (Polit et al., 2007; Polit & Beck, 2006). Secara umum proses validasi

isi disajikan pada Gambar 1. Tahap pertama dilakukan melalui perhitungan *Content Validity Ratio* (CVR) untuk menilai aspek kelengkapan. Skor ahli dikonversi ke dalam tiga kategori, yaitu: *essential* (+1) untuk skor 3–4, *useful but not essential* (0) untuk skor 2, dan *not necessary* (-1) untuk skor 1. Indikator dengan $CVR \leq 0$ dieliminasi dari analisis selanjutnya (Almanasreh et al., 2019).

Item yang memenuhi syarat dari CVR kemudian dievaluasi lebih lanjut menggunakan *Item-level Content Validity Index* (I-CVI) berdasarkan aspek kejelasan, relevansi, dan konsistensi. Perhitungan I-CVI dilakukan dengan menghitung proporsi ahli yang memberikan skor 3–4. Item dianggap valid jika memperoleh nilai $I-CVI \geq 0.78$ untuk jumlah ahli minimal 5 orang (Almanasreh et al., 2019).

Selanjutnya untuk mengontrol kesepakatan acak digunakan rumus kappa modifikasi (κ^*) (Polit et al., 2007). Item dengan nilai $\kappa^* \geq 0.74$ menunjukkan tingkat kesepakatan sangat baik. Nilai kappa modifikasi dihitung dengan terlebih dahulu menghitung probabilitas kesepakatan acak dari para ahli untuk menilai item indikator dengan nilai yang sama.

Tahap terakhir dalam analisis ini menghitung *Average Content Validity Index* (Ave-CVI), yaitu rata-rata nilai I-CVI dari seluruh item (Polit & Beck, 2006). Nilai Ave-CVI ≥ 0.90 menunjukkan bahwa instrumen memiliki validitas isi yang sangat baik. Jika nilai Ave-CVI < 0.90 , maka disarankan untuk melakukan revisi indikator dan dilanjutkan dengan validasi putaran kedua (Almanasreh et al., 2019).



Gambar 1: Prosedur analisis validitas isi (diadaptasi dari Almanasreh et al., 2019)

3. Hasil dan Pembahasan

Pada tahap awal konstruksi, penelitian ini menghasilkan sebuah instrumen yang terdiri atas 23 item, yang disusun berdasarkan adaptasi dari literatur sebelumnya dan digolongkan menjadi tiga domain utama: pemahaman konsep dasar algoritma, keterampilan pemrograman, dan integrasi berpikir komputasional dalam tahapan *problem solving*. Setiap item mencerminkan satu indikator kesulitan belajar siswa dalam pembelajaran pemrograman di tingkat Sekolah Menengah Pertama (SMP). Tabel 1 merupakan struktur indikator instrumen yang dihasilkan pada tahap awal konstruksi.

Tabel 1. Struktur Indikator Instrumen

Domain	Subdomain	Nama dan butir indikator
Pemahaman Konsep Dasar Algoritma	Definisi Algoritma (Kadar et al., 2021)	(k1) Kesulitan dalam mengurutkan instruksi suatu penyelesaian masalah dalam kehidupan sehari-hari
	Karakteristik suatu algoritma (Knuth, 2016)	(k2) Kesulitan dalam memahami karakteristik definiteness (setiap langkah didefinisikan secara tepat)
		(k3) Memahami karakteristik finiteness (sebuah algoritma harus selalu berhenti setelah sejumlah langkah yang terbatas)
		(k4) Kesulitan dalam memahami karakteristik effectiveness (setiap tahapan harus efektif)

Domain	Subdomain	Nama dan butir indikator	
	Konsep Input-Proses- Output (Islam et al., 2019)	(k5) Kesulitan dalam mengidentifikasi IPO dan memahami bagaimana suatu input diolah (proses) menjadi output dalam suatu algoritma	
	Variabel, Tipe data, dan Operator (Baist & Pamungkas, 2017; Islam et al., 2019)	(k6) Kesulitan dalam menetapkan atau menginisialisasi variabel untuk menyimpan dan mengolah data dalam algoritma sederhana	
		(k7) Kesulitan dalam memahami perbedaan antara berbagai tipe data (integer, float, string, boolean, array) suatu variabel	
		(k8) Kesulitan dalam penggunaan dan ekspresi berbagai operator (aritmetika, perbandingan, logika)	
	Struktur Kontrol Baist & Pamungkas, 2017; Islam et al., 2019)	(k9) Kesulitan dalam memahami pola paragraf kondisional	
		(k10) Kesulitan dalam memahami pola paragraf pengulangan	
		(k11) Kesulitan dalam memahami bentuk rekursi	
	Keterampilan Pemograman	Tracing (Xie et al., 2019)	(s1) Kesulitan dalam memprediksi perubahan status/nilai (variabel) dan output dari suatu algoritma
		Writing (Xie et al., 2019)	(s2) Kesulitan dalam menyusun tahap algoritma yang secara notasi baik sintaksis maupun semantik benar (dalam bentuk pseudocode atau flowchart)
			(s3) Kesulitan dalam penggunaan secondary notation (comments, spasi dan baris kosong, indentasi kode, and penyorotan kata kunci).
		Testing (Kadar et al., 2021)	(s4) Kesulitan dalam mengidentifikasi dan memverifikasi batasan (constraints) data input
(s5) Kesulitan dalam mengidentifikasi dan memverifikasi apakah algoritma menghasilkan output yang benar untuk berbagai skenario input.			
Debugging (Kadar et al., 2021)	(s6) Kesulitan dalam menemukan dan memperbaiki kesalahan dalam algoritma		
Integrasi Keterampilan Berpikir Komputasional dalam Problem Solving (Yusoff et al., 2020)	Memahami dan Mendefinisikan Masalah (abstraksi)	(bk1) Kesulitan dalam mengidentifikasi input dan output yang dibutuhkan dari suatu masalah.	
	Merencanakan solusi (dekomposisi)	(bk2) Kesulitan dalam mendekomposisi proses penyelesaian masalah	
	Membuat solusi (Pengenalan pola dan Berpikir Algoritmik)	(bk3) Kesulitan dalam memodifikasi algoritma dari masalah serupa (penggunaan template algoritma)	
		(bk4) Kesulitan dalam membangun/membuat algoritma	
	Mengevaluasi solusi	(bk5) Kesulitan dalam mengidentifikasi dan memverifikasi apakah algoritma berfungsi sesuai tujuan penyelesaian masalah	
Efektifitas solusi	(bk6) Kesulitan dalam mengidentifikasi dan memverifikasi apakah algoritma berfungsi secara efektif		

Validasi isi item instrumen dilakukan dengan pendekatan kuantitatif yang melibatkan penilaian oleh enam orang ahli. Para ahli tersebut mengevaluasi masing-masing item instrumen berdasarkan aspek kelengkapan, kejelasan, relevansi, dan konsistensi. Penilaian ahli kemudian dianalisis menggunakan metode yang direkomendasikan oleh Polit et al. (2007) dan Almasreh et al. (2019), yaitu melalui perhitungan *Content Validity Ratio (CVR)* setiap indikator, *Content Validity Index* pada level Item (*I-CVI*), dan koefisien Kappa modifikasi (κ^*).

Tabel 2 menyajikan hasil analisis validasi isi terhadap item-item dalam instrumen. Seluruh item memperoleh nilai $CVR > 0$, yang menunjukkan bahwa mayoritas ahli menilai item sebagai essential, sehingga tidak ada item yang dieliminasi pada tahap awal. Analisis *I-CVI* menunjukkan bahwa semua item memiliki nilai antara 0.83 hingga 1.00, dan nilai kappa modifikasi (κ^*) berada dalam rentang 0.82 hingga 1.00, yang mengindikasikan tingkat kesepakatan antar ahli berada pada kategori baik hingga sangat baik. Sebagai pengukuran agregat, nilai *Average-CVI (Ave-CVI)* instrumen secara keseluruhan adalah 0.91, yang mengindikasikan bahwa instrumen memiliki validitas isi yang sangat baik. Nilai ini melampaui

ambang batas yang disarankan, yaitu 0.90 (Almanasreh et al., 2019), dan mendukung kualitas keseluruhan dari instrumen yang dikembangkan.

Tabel 2. Hasil Validasi Isi Item Instrumen

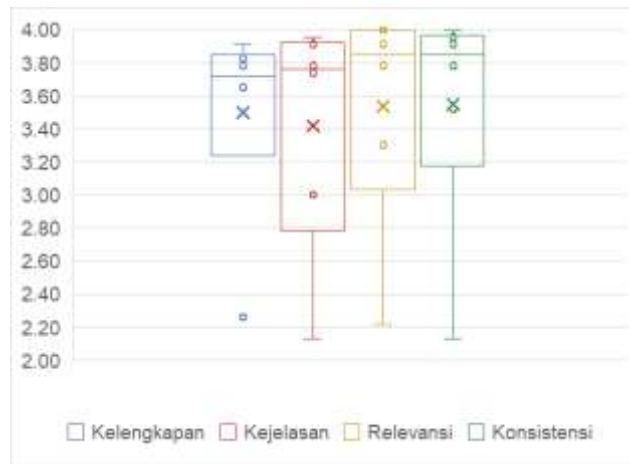
No	Nama Item	CVR	I-CVI	κ^*	Keterangan
1	k1	0.67	0.94	0.94	Sangat baik
2	k2	1.00	1.00	1.00	Sangat baik
3	k3	1.00	1.00	1.00	Sangat baik
4	k4	1.00	1.00	1.00	Sangat baik
5	k5	0.67	0.83	0.82	Sangat baik
6	k6	0.67	0.83	0.82	Sangat baik
7	k7	1.00	1.00	1.00	Sangat baik
8	k8	0.67	0.83	0.82	Sangat baik
9	k9	1.00	1.00	1.00	Sangat baik
10	k10	1.00	1.00	1.00	Sangat baik
11	k11	1.00	1.00	1.00	Sangat baik
12	S1	0.67	0.83	0.82	Sangat baik
13	S2	0.67	0.83	0.82	Sangat baik
14	S3	0.67	0.83	0.82	Sangat baik
15	S4	0.67	0.83	0.82	Sangat baik
16	S5	0.67	0.83	0.82	Sangat baik
17	S6	0.67	0.83	0.82	Sangat baik
18	bk1	0.67	0.83	0.82	Sangat baik
19	bk2	1.00	1.00	1.00	Sangat baik
20	bk3	1.00	1.00	1.00	Sangat baik
21	bk4	1.00	1.00	1.00	Sangat baik
22	bk5	0.67	0.83	0.82	Sangat baik
23	bk6	0.67	0.83	0.82	Sangat baik
Average CVI (Ave-CVI)			0.91		

Selanjutnya, dilakukan analisis rata-rata per domain untuk mengetahui kekuatan validitas isi dalam kelompok indikator tertentu (tingkat domain) yang dirangkum dalam Tabel 2. Berdasarkan kolom Rata-rata I-CVI dan Rata-rata κ^* , maka dapat disimpulkan bahwa setiap domain memiliki nilai Ave-CVI yang tinggi dan relatif seimbang. Hal ini menandakan konsistensi kualitas item di seluruh bagian instrumen.

Tabel 2. Hasil Validasi Isi Item Instrumen

Domain	Rata-rata I-CVI	Rata-rata κ^*	Jumlah item valid
Pemahaman Konsep Dasar Algoritma	0.95	0.94	11
Keterampilan Pemograman	0.83	0.82	6
Integrasi Keterampilan Berpikir Komputasional dalam <i>Problem Solving</i>	0.92	0.91	6

Untuk memperkuat interpretasi data, hasil validasi divisualisasikan dalam bentuk boxplot (Gambar 2) untuk menggambarkan persebaran skor penilaian ahli pada setiap aspek. Secara keseluruhan, aspek relevansi dan konsistensi secara dominan mendapat penilaian positif meskipun masih terdapat beberapa item yang perlu perbaikan. Namun, aspek kejelasan menunjukkan variasi terbesar dalam penilaian ahli, dengan beberapa ahli secara eksplisit menilai bahwa sejumlah indikator perlu direvisi agar lebih jelas. Aspek kelengkapan juga memerlukan perhatian pada item tertentu yang dinilai kurang mencakup unsur-unsur penting oleh sebagian kecil ahli.



Gambar 2: Boxplot Penilaian Aspek Kejelasan, Kelengkapan, Relevansi, dan Konsistensi dari ahli

Hasil validasi ini menjadi dasar bagi peneliti dalam melakukan revisi instrumen dengan mengacu pada rekomendasi yang disampaikan oleh para ahli. Secara umum, tidak ada perubahan signifikan pada tingkat dimensi, namun dilakukan revisi pada tingkat subdimensi dengan penambahan subdimensi modular. Selain itu, beberapa indikator mengalami perbaikan secara redaksional agar lebih spesifik dan jelas dalam menggambarkan kesulitan siswa, khususnya pada indikator yang berkaitan dengan pemahaman definisi algoritma, penggunaan variabel, operator, serta struktur kontrol. Penyempurnaan redaksional ini bertujuan meningkatkan keterbacaan dan relevansi item dengan konteks pembelajaran pemrograman di tingkat SMP, sehingga instrumen dapat secara akurat mengidentifikasi area spesifik yang memerlukan intervensi.

Agar indikator lebih operasional, item-item yang sebelumnya mengandung lebih dari satu level kognitif (kata kerja) atau lebih dari satu konsep (kata benda) dipecah menjadi dua atau tiga indikator yang terpisah. Sebagai contoh, indikator awal "*Kesulitan dalam penggunaan dan ekspresi berbagai operator (aritmetika, perbandingan, logika)*" direvisi menjadi dua indikator terpisah, yakni "*Kesulitan dalam menentukan tipe data dalam membuat algoritma sederhana*" dan "*Kesulitan dalam menggunakan operator dalam membuat algoritma sederhana.*"

Instrumen hasil revisi terdiri atas 43 item yang tetap terbagi ke dalam tiga dimensi utama, yaitu pemahaman konsep algoritma (terdiri dari 6 subdimensi dengan 17 butir), keterampilan pemrograman (terdiri dari 4 subdimensi dengan 14 butir), dan keterampilan berpikir komputasional dalam pemecahan masalah (terdiri dari 5 subdimensi dengan 12 butir). Setiap item pada instrumen revisi memperoleh skor 3 atau 4 dari seluruh validator, sehingga diperoleh nilai $CVR > 0$, serta nilai $I-CVI$ dan koefisien Kappa modifikasi (κ^*) sebesar 1.00. Hal ini menunjukkan bahwa instrumen memiliki validitas isi yang sangat tinggi serta tingkat kesepakatan antar ahli yang sempurna.

Pembahasan

Secara teoretis, struktur indikator dalam instrumen ini dirancang untuk mendukung pendekatan konstruktivis dalam pembelajaran informatika. Pendekatan ini menekankan bahwa proses belajar merupakan hasil konstruksi pengetahuan melalui pengalaman aktif dan keterlibatan langsung siswa dalam memecahkan masalah. Dalam konteks pembelajaran pemrograman di jenjang SMP, kesulitan belajar tidak hanya disebabkan oleh kurangnya pemahaman terhadap konsep dasar, tetapi juga berkaitan erat dengan lemahnya kemampuan penyelesaian masalah serta berpikir komputasional siswa (Yusoff et al., 2020). Kondisi ini menjadi tantangan tersendiri, mengingat implementasi kurikulum saat ini telah mengintegrasikan pemrograman untuk mencapai kecakapan literasi digital abad ke-21 (Huzooree & Soupramanien, 2024; Rodrigues et al., 2024; Zeeshan et al., 2024) yang menuntut penguasaan keterampilan berpikir tingkat tinggi. Oleh karena itu, pengembangan instrumen ini

memiliki relevansi yang strategis dalam mendukung implementasi kurikulum, dengan menyediakan alat yang dapat mengidentifikasi secara spesifik area kesulitan siswa dan menjadi dasar perancangan intervensi pembelajaran yang mengembangkan keterampilan berpikir komputasional seperti hasil penelitian dari beberapa literatur (Altiok & Üçgül, 2025; Liu et al., 2023; Mills et al., 2024).

Domain “*Pemahaman Konsep Dasar Algoritma*” mencakup elemen-elemen seperti penyusunan instruksi yang tepat sebagai definisi algoritma (Kadar et al., 2021), struktur kontrol, variabel dan tipe data, serta input-proses-output (IPO) yang telah dilaporkan dalam berbagai studi sebagai sumber kesulitan utama bagi siswa pemula (Baist & Pamungkas, 2017; Islam et al., 2019; Kadar et al., 2021). Peneliti juga menambahkan subdomain modularitas sebagai elemen kesulitan yang sering ditemui pada kalangan mahasiswa (Kadar et al., 2021; Özmen & Altun, 2014) sebagai deteksi dini. Peneliti juga menambahkan konsep dasar operator sebagai bagian dari indikator, karena kebutuhan pendefinisian ekspresi dalam struktur kontrol maupun modularitas. Untuk lebih memudahkan dalam penyusunan instrumen tes sebagai pengembangan dari indikator yang telah diteliti, penulis menyusun kemunculan indikator berdasarkan pada *raison d'être* (Berglund & Lister, 2010) dari konsep dasar algoritma ini.

Domain “*Keterampilan Pemrograman*” mencerminkan kompleksitas kognitif siswa dalam proses *tracing* dan *writing* yang ditekankan oleh Xie et al. (2019), serta *testing* dan *debugging* (Baist & Pamungkas, 2017; Islam et al., 2019; Kadar et al., 2021; Özmen & Altun, 2014). Keterampilan *tracing*, merupakan keterampilan yang jarang ditemui dalam literatur sebelumnya terkait dengan kesulitan belajar. Akan tetapi menurut beberapa literatur dalam strategi pembelajaran pemrograman terstruktur (Bradley, 2024; Peserico et al., 2023; Sentance & Waite, 2021) keterampilan ini merupakan keterampilan yang harus dikuasai sebelum siswa mampu menuliskan algoritma. Seperti halnya siswa harus mampu membaca sebelum belajar menulis. Lebih lanjut, pengelompokan pada domain ini berbeda dengan penelitian sebelumnya (Özmen & Altun, 2014) yang mengelompokkan kesulitan berdasarkan keterampilan dengan penekanan pada *problem solving*. Peneliti membuat domain tersendiri untuk kesulitan terkait dengan *problem solving* yang diintegrasikan dengan kemampuan berpikir komputasional (Yusoff et al., 2020).

Domain “*Integrasi Berpikir Komputasional dalam Tahapan Problem Solving*” ini merupakan upaya deteksi kemampuan siswa untuk memenuhi capaian pembelajaran elemen Berpikir Komputasional terintegrasi Algoritma dan pemrograman dalam mata pelajaran Informatika Fase D di Indonesia. Yusuff et al. (2020) mengkonstruksi kemampuan berpikir komputasional dalam 6 elemen, yaitu dekomposisi, abstraksi, pengenalan pola, algoritma, penalaran logis, serta asesmen dan evaluasi. Keenam elemen ini menjadi acuan peneliti untuk diintegrasikan dalam proses *problem solving*. Temuan Zeeshan et al. (2024) memperkuat relevansi domain ini, di mana pembelajaran coding secara sistematis terbukti dapat mengembangkan kemampuan berpikir komputasional siswa, terutama dalam hal menyusun strategi penyelesaian masalah yang logis dan efisien. Oleh karena itu, domain ini tidak hanya berperan sebagai indikator kesulitan, tetapi juga sebagai jembatan untuk mengembangkan keterampilan berpikir tingkat tinggi yang menjadi inti dalam pembelajaran informatika.

Faktor-faktor kesulitan lain yang telah ditemukan oleh peneliti terdahulu, seperti penggunaan lingkungan pengembangan program (Baist & Pamungkas, 2017; Islam et al., 2019) dan penggunaan bahasa pemrograman tertentu (Baist & Pamungkas, 2017; Kadar et al., 2021; Özmen & Altun, 2014) tidak menjadi fokus dalam penelitian, karena belum menjadi capaian pembelajaran untuk siswa SMP. Meskipun demikian kesulitan siswa dalam membuat program sederhana dengan menggunakan pemrograman blok dapat menjadi pertimbangan untuk diteliti lebih lanjut, karena pembelajaran dengan pemrograman blok ini telah diimplementasikan sebagai salah satu strategi dalam berbagai literatur sebelumnya (Dorodchi et al., 2021; Grover, 2021; Weng et al., 2023).

Hasil penelitian menunjukkan bahwa instrumen yang dikembangkan untuk mengidentifikasi kesulitan belajar siswa SMP dalam pembelajaran pemrograman memiliki validitas isi yang sangat baik, sebagaimana ditunjukkan oleh nilai-nilai *CVR*, *I-CVI*, dan kappa modifikasi (κ^*) yang tinggi dan konsisten di seluruh item. Seluruh item memperoleh nilai *CVR* > 0 , yang mengindikasikan bahwa tidak ada item yang dinilai “tidak esensial” oleh mayoritas ahli. Hal ini mengonfirmasi bahwa konstruksi indikator telah sesuai dengan prinsip validitas isi sebagaimana dikemukakan oleh Polit dan Beck (2006) dan diperkuat oleh Almanasreh et al. (2019).

Lebih lanjut, nilai *I-CVI* yang berada dalam rentang 0.83 hingga 1.00 menunjukkan tingkat kesepakatan yang tinggi terhadap kejelasan, relevansi, dan konsistensi setiap item. Hasil ini merujuk pada temuan Almanasreh et al. (2019) yang menyatakan bahwa *I-CVI* ≥ 0.78 dapat diterima sebagai valid dalam kelompok panelis kecil (5–10 orang). Nilai kappa modifikasi (κ^*) yang juga tinggi (> 0.80) menandakan bahwa kesepakatan yang dicapai para ahli tidak terjadi secara kebetulan, melainkan berdasarkan pertimbangan substansial terhadap isi setiap indikator.

Kekuatan instrumen ini terletak pada keterpaduan antara kajian teoretis dan validasi empiris oleh para ahli, yang menghasilkan indikator yang tidak hanya sah secara konten, tetapi juga kontekstual dengan kebutuhan dan karakteristik pembelajaran pemrograman di SMP. Hal ini juga memperkuat gagasan bahwa pengukuran kesulitan belajar harus dilakukan secara multidimensi, karena kesulitan siswa tidak hanya terletak pada satu aspek tertentu, melainkan kombinasi dari pengetahuan deklaratif, prosedural, dan strategis (Khoirotunnisa et al., 2022). Lebih khusus lagi kajian mengenai kesulitan belajar dapat juga dianalisis dari hambatan belajar yang diakibatkan oleh desain didaktis guru (Brousseau et al., 2002), namun hal ini tidak menjadi prioritas dalam penelitian ini.

Meski begitu, terdapat sejumlah keterbatasan dalam penelitian ini. Pertama, validitas isi hanya melibatkan enam orang ahli, sehingga meskipun sudah mencukupi menurut literatur, hasil ini masih memerlukan penguatan lebih lanjut melalui validasi eksternal di populasi siswa. Kedua, penelitian ini belum melakukan uji reliabilitas empiris maupun analisis faktor eksploratori (EFA) yang diperlukan untuk menguji konsistensi internal dan struktur konstruk secara statistik sebagaimana dilakukan oleh Masuwai et al. (2024).

Dari sisi implikasi, instrumen ini memiliki potensi yang besar untuk digunakan sebagai pengembangan alat diagnostik formatif dalam konteks pembelajaran maupun pengembangan instrumen survey persepsi kesulitan belajar kepada guru. Guru dapat menggunakan hasil pengukuran untuk mengidentifikasi letak kesulitan siswa secara spesifik, sehingga dapat merancang pembelajaran diferensiatif yang lebih tepat sasaran. Di samping itu, instrumen ini juga dapat digunakan dalam penelitian lebih lanjut untuk mengevaluasi efektivitas intervensi pembelajaran berbasis kesulitan belajar.

4. Kesimpulan

Penelitian ini telah berhasil mengembangkan instrumen yang valid secara isi untuk mengidentifikasi kesulitan belajar pemrograman siswa SMP. Instrumen terdiri dari 23 item yang mencakup domain pemahaman konsep algoritma, keterampilan pemrograman, dan berpikir komputasional dalam pemecahan masalah. Seluruh item menunjukkan nilai *CVR* > 0 , nilai *I-CVI* antara 0.83–1.00, dan koefisien Kappa modifikasi (κ^*) > 0.80 . Setelah revisi, instrumen terdiri dari 43 indikator dengan nilai *I-CVI* dan koefisien Kappa modifikasi adalah 1. Hasil ini menunjukkan validitas isi yang sangat baik dan tingkat kesepakatan ahli yang tinggi.

Instrumen ini berpotensi besar digunakan sebagai alat diagnostik oleh guru untuk merancang intervensi pembelajaran yang tepat sasaran. Namun demikian, penelitian lanjutan tetap diperlukan untuk menguji validitas konstruk dan reliabilitas empiris melalui uji coba dengan sampel yang lebih besar. Selain itu, indikator instrumen perlu ditinjau secara berkala

agar relevan dengan perkembangan kurikulum dan teknologi, serta mendukung keterampilan abad 21.

5. Referensi

- 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. <https://doi.org/10.1016/j.sapharm.2018.03.066>
- Almdahem, A. (2024). The perceptions of secondary school students regarding the learning of computer programming at Key Stage 4 in the revised curriculum in England: problems and solutions. *International Journal of Computer Science Education in Schools*, 7(1). <https://doi.org/10.21585/ijcses.v7i1.147>
- Altıok, S., & Üçgül, M. (2025). Effects of Robotic Coding on Computational Thinking Skills of Secondary School Students. *Canadian Journal of Learning and Technology*, 50(2), 1–24. <https://doi.org/10.21432/cjlt28607>
- Andrzejewska, M. (2020). *Difficulties in learning introductory programming - students' perspective*. February, 1–11.
- Arodani, M. P., Misriyani, Firdausy, F., & odi, M. (2025). Pendidikan sekolah dasar 2024; menyiapkan generasi emas dengan keterampilan abad 21. *Jurnal Ilmiah Research Student*, 2(1), 145–154. <https://doi.org/10.61722/jirs.v2i1.3633>
- Baist, A., & Pamungkas, A. S. (2017). Analysis of Student Difficulties in Computer Programming. *VOLT: Jurnal Ilmiah Pendidikan Teknik Elektro*, 2(2), 81. <https://doi.org/10.30870/volt.v2i2.2211>
- Berglund, A., & Lister, R. (2010). Introductory Programming and the Didactic Triangle. *Conferences in Research and Practice in Information Technology Series*, 103(Ace), 35–44.
- Bradley, S. (2024). *PRIMM and Proper: Authentic Investigation in HE Introductory Programming with PeerWise and GitHub*. 33–36. <https://doi.org/10.1145/3633053.3633062>
- Brousseau, G., Balacheff, N., Cooper, M., Sutherland, R., & Warfield, V. (Eds.). (2002). *Epistemological Obstacles, Problems, and Didactical Engineering BT - Theory of Didactical Situations in Mathematics: Didactique des Mathématiques, 1970–1990* (pp. 79–117). Springer Netherlands. https://doi.org/10.1007/0-306-47211-2_6
- Chan Mow, I. T. (2008). Issues and difficulties in teaching novice computer programming. *Innovative Techniques in Instruction Technology, E-Learning, E-Assessment, and Education*, 199–204. <https://doi.org/10.1007/978-1-4020-8739-4-36>
- Connelly, L. M. (2022). Measurement Instrument Validity. *MEDSURG Nursing*, 31(1), 2022. <https://www.proquest.com/openview/da54df381f59305b07cc6d6522823639/1?pq-origsite=gscholar&cbl=30764>
- Deviana, T., Mn, F. R., Sulistyani, N., & Mulyda, 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.*
- Dorodchi, M. M., Benedict, A., Rorrer, A., Pugalee, D. K., & Cao, L. (2021). Broadening the Middle School Computational Thinking Interventions Beyond Block Programming. *ASEE Annual Conference and Exposition, Conference Proceedings*. <https://doi.org/10.18260/1-2--36757>
- Giannakoulas, A., & Xinogalos, S. (2024). A Critical Review of Primary School Students' Difficulties in Learning Programming Through Educational Games. *Journal of Educational Computing Research*, 0(0), 1–35. <https://doi.org/10.1177/07356331241309074>
- Grover, S. (2021). *Teaching and Assessing for Transfer from Block-to-Text Programming in Middle School Computer Science BT - Transfer of Learning: Progressive Perspectives for Mathematics Education and Related Fields* (C. Hohensee & J. Lobato (Eds.); pp. 251–276). Springer

- International Publishing. https://doi.org/10.1007/978-3-030-65632-4_11
- Hamsyah, D., Luzyawati, L., & Yuliana, E. (2020). Validitas Instrumen Penalaran Ilmiah pada Materi Sistem Pertahanan Tubuh Kelas XI. *Quagga: Jurnal Pendidikan Dan Biologi*, 13(1), 26. <https://doi.org/10.25134/quagga.v13i1.3474>
- Huzooree, G., & Soupramanien, L. D. B. (2024). Empowering Students for the 21st Century Through Digital Literacy. *Advances in Higher Education and Professional Development Book Series*, 161–200. <https://doi.org/10.4018/979-8-3693-3856-8.ch005>
- Islam, N., Shafi Sheikh, G., Fatima, R., & Alvi, F. (2019). A Study of Difficulties of Students in Learning Programming. *Journal of Education & Social Sciences*, 7(2), 38–46. <https://doi.org/10.20547/jess0721907203>
- Kadar, R., Abdul Wahab, N., Othman, J., Shamsuddin, M., & Mahlan, S. B. (2021). A Study of Difficulties in Teaching and Learning Programming: A Systematic Literature Review. *International Journal of Academic Research in Progressive Education and Development*, 10(3), 591–605. <https://doi.org/10.6007/ijarped/v10-i3/11100>
- Khoirotunnisa, A. U., Amin, A. K., & Taufiqurrohman, M. (2022). Analisis Kesulitan Belajar Siswa dengan Taksonomi Solo. *Jurnal Pendidikan Edutama*, 9(2), 249–258.
- Knuth, D. E. (2016). The Art of Computer Programming. In *Nuclear Science and Engineering: Vol. 1/Fundamen* (3rd ed.). Mathematical Sciences Publishers (MSP). <https://doi.org/10.13182/nse70-a19705>
- Lahtinen, E., Ala-Mutka, K., & Järvinen, H. M. (2005). A study of the difficulties of novice programmers. *Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, 14–18. <https://doi.org/10.1145/1067445.1067453>
- Liu, H., Li, Z., Duan, J. T., & Zeng, Y. (2023). The Impact of Pair Programming Based on Project-Based Learning on Computational Thinking of Junior High School Students. *Proceedings of the 2022 5th International Conference on Education Technology Management*, 287–291. <https://doi.org/10.1145/3582580.3582630>
- Marselina, K. T., Lasmawan, I. W., & Dantes, N. (2021). Pengembangan Instrumen Kemampuan Berpikir Kritis Dan Hasil Belajar Ips Pada Siswa Kelas V Sd. *Jurnal Penelitian Dan Evaluasi Pendidikan Indonesia*, 11(2), 105–114. <https://doi.org/10.23887/jpepi.v11i2.620>
- Masuwai, A., Zulkifli, H., & Hamzah, M. I. (2024). Evaluation of content validity and face validity of secondary school Islamic education teacher self-assessment instrument. *Cogent Education*, 11(1). <https://doi.org/10.1080/2331186X.2024.2308410>
- Mills, K. A., Cope, J., Scholes, L. L., & Rowe, L. (2024). Coding and Computational Thinking Across the Curriculum: A Review of Educational Outcomes. *Review of Educational Research*. <https://doi.org/10.3102/00346543241241327>
- Özmen, B., & Altun, A. (2014). Undergraduate Students' Experiences in Programming: Difficulties and Obstacles. *Turkish Online Journal of Qualitative Inquiry*, 5(3), 9–27. <https://doi.org/10.17569/tojqi.20328>
- Peserico, G., Serafini, M., Voltolini, F., Picasso, F., Agostini, D., Fiore, F., Serbati, A. S., & Montresor, A. (2023). the Primm Method for Teaching Programming: Experimentation and Validation. *ICERI2023 Proceedings*, 1(November), 7278–7282. <https://doi.org/10.21125/iceri.2023.1810>
- Polit, D. F., & Beck, C. T. (2006). Focus on Research Methods Handling Missing Data in Self-Report Measures. *Research in Nursing & Health*, 29, 488–495. <https://doi.org/10.1002/nur>
- Polit, D. F., Beck, C. T., & Owen, S. V. (2007). Focus on Research Methods Handling Missing Data in Self-Report Measures. *Research in Nursing & Health*, 30, 459–467. <https://doi.org/10.1002/nur>
- Rodrigues, C. A. D., Mendes, A. D., Wanderley, A. A., Cardoso, G. P., Fernandes, M. D. F.,

- Barbosa, T. M., de Muniz, T. C., & Demuner, J. A. (2024). *Skills development for the 21st century: the influence of ai and technologies on the educational curriculum*. 6(3). <https://doi.org/10.56238/arev6n3-090>
- Sarro, K. A., Chacón-Castro, M., Madrigal, E. G. M., & Jadán-Guerrero, J. (2024). *Didactic Strategy for the Teaching of Educational Robotics and Programming at Secondary Level* (pp. 448–457). Springer International Publishing. https://doi.org/10.1007/978-3-031-54256-5_42
- Sentance, S., & Waite, J. (2021). Teachers' Perspectives on Talk in the Programming Classroom: Language as a Mediator. In *ICER 2021 - Proceedings of the 17th ACM Conference on International Computing Education Research* (Vol. 1, Issue 1). Association for Computing Machinery. <https://doi.org/10.1145/3446871.3469751>
- Waite, J., Curzon, P., Marsh, W., & Sentance, S. (2020). Difficulties with design: The challenges of teaching design in K-5 programming. *Computers and Education*, 150, 1–44. <https://doi.org/10.1016/j.compedu.2020.103838>
- Weng, X., Ng, O. L., Cui, Z., & Leung, S. (2023). Creativity Development With Problem-Based Digital Making and Block-Based Programming for Science, Technology, Engineering, Arts, and Mathematics Learning in Middle School Contexts. *Journal of Educational Computing Research*, 61(2), 304–328. <https://doi.org/10.1177/07356331221115661>
- Woo, K., & Falloon, G. (2024). Moving beyond the rhetoric: integrating coding into the English curriculum in Australian primary schools. *Australian Educational Researcher*. <https://doi.org/10.1007/s13384-024-00742-z>
- Xie, B., Loksa, D., Nelson, G. L., Davidson, M. J., Dong, D., Kwik, H., Tan, A. H., Hwa, L., Li, M., & Ko, A. J. (2019). A theory of instruction for introductory programming skills. *Computer Science Education*, 29(2–3), 205–253. <https://doi.org/10.1080/08993408.2019.1565235>
- Yermaganbetova, M. A., Meirmanov, S., & Tokzhigitova, A. N. (2024). Overview of methodological approaches and technologies of teaching programming in secondary schools. *Л.Н. Гумилев Атындағы Еуразия Ұлттық Университетінің Хабаршысы*, 149(4), 96–111. <https://doi.org/10.32523/2616-6895-2024-149-4-96-111>
- Yusoff, K. M., Ashaari, N. S., Wook, T. S. M. T., & Ali, N. M. (2020). Analysis on the requirements of computational thinking skills to overcome the difficulties in learning programming. *International Journal of Advanced Computer Science and Applications*, 11(3), 244–253. <https://doi.org/10.14569/ijacsa.2020.0110329>
- Zeeshan, K., Hämäläinen, T., & Neittaanmäki, P. (2024). Computational Thinking and AI Coding for Kids to Develop Digital Literacy. *International Journal of Education*, 12(3), 55–74. <https://doi.org/10.5121/ije2024.12305>