

**ANALISIS KETERAMPILAN BERPIKIR KOMPUTASIONAL SISWA DALAM
MENYELESAIKAN PROYEK PEMROGRAMAN
(STUDI KASUS DI KELAS X SMA KRISTEN CITRA BANGSA)**

Lukas Feri Natapuling¹, Maria Magdalena Beatrice Sogen²,
Diana Yanni Ariswati Fallo³,

^{1,2,3} PENDIDIKAN INFORMATIKA FKIP Universitas Citra Bangsa Kupang

¹lukasferinatapuling8@gmail.com, ²beatricesogen11@gmail.com,
³dianayani25@gmail.com

ABSTRACT

Lukas Ferry Natapuling, 2026. Analysis of Students' Computational Thinking Skills in Completing Programming Projects (A Case Study of Grade X Students at Citra Bangsa Christian Senior High School). Informatics Education Study Program, Faculty of Teacher Training and Education, Citra Bangsa University. Supervisor I: Dr. Maria M. B. Sogen, S.Kom., M.Pd., Supervisor II: Diana Yanni Ariswati Fallo, S.Kom., M.T. This study aims to describe the computational thinking processes of Grade X students at Citra Bangsa Christian Senior High School in completing programming projects and to identify the challenges and strategies used by students to overcome difficulties during the process. Computational thinking is an essential 21st-century skill that includes decomposition, pattern recognition, abstraction, and algorithmic thinking. These skills are necessary to help students solve problems logically, systematically, and effectively in informatics learning. This research employed a qualitative approach using a case study design. The research subjects were Grade X students of Citra Bangsa Christian Senior High School selected through purposive sampling based on their active involvement in programming projects. Data were collected through observation, semi-structured interviews, and documentation. Data analysis was conducted using the Miles and Huberman model, which consists of data reduction, data display, and conclusion drawing. Data validity was ensured through source and technique triangulation. The findings indicate that students applied various aspects of computational thinking in completing programming projects, although their level of mastery varied across different aspects. Decomposition and pattern recognition were evident in students' ability to break down problems and identify similarities in solution procedures. However, abstraction and algorithmic thinking remained challenging for some students, particularly in filtering relevant information and developing systematic solution steps. The challenges encountered included difficulties in understanding programming logic, determining appropriate algorithms, and resolving program errors. To overcome these challenges, students employed several strategies, such as discussing problems with peers, consulting teachers, searching for references on the internet, and using trial-and-error methods. This study provides a deeper understanding of students' computational thinking processes and may serve as a reference for teachers in designing more effective and student-centered informatics learning experiences.

Keywords: Computational Thinking, Decomposition, Abstraction, Algorithmic Thinking, Programming Projects, Case Study.

ABSTRAK

Lukas Ferry Natapuling, 2026 Analisis Keterampilan Berpikir Komputasional Siswa dalam Menyelesaikan Proyek Pemrograman (Studi Kasus di Kelas X SMA Kristen Citra Bangsa). Program Studi Pendidikan Informatika, Fakultas Keguruan dan Ilmu Pendidikan, Universitas Citra Bangsa. Pembimbing I: Dr. Maria M. B. Sogen, S.Kom., M.Pd., Pembimbing II: Diana Yanni Ariswati Fallo, S.Kom., M.T. Penelitian ini bertujuan untuk mendeskripsikan proses berpikir komputasional siswa kelas X SMA Kristen Citra Bangsa dalam menyelesaikan proyek pemrograman serta mengidentifikasi hambatan dan strategi yang digunakan siswa dalam mengatasi kesulitan selama proses tersebut. Berpikir komputasional merupakan keterampilan penting abad ke-21 yang meliputi dekomposisi, pengenalan pola, abstraksi, dan algoritma. Kemampuan ini diperlukan untuk membantu siswa menyelesaikan masalah secara logis, sistematis, dan efektif dalam pembelajaran informatika. Penelitian ini menggunakan pendekatan kualitatif dengan jenis penelitian studi kasus. Subjek penelitian adalah siswa kelas X SMA Kristen Citra Bangsa yang dipilih menggunakan teknik purposive sampling berdasarkan keterlibatan aktif dalam proyek pemrograman. Teknik pengumpulan data dilakukan melalui observasi, wawancara semi-terstruktur, dan dokumentasi. Analisis data menggunakan model Miles dan Huberman yang meliputi reduksi data, penyajian data, dan penarikan kesimpulan. Keabsahan data diperoleh melalui triangulasi sumber dan teknik. Hasil penelitian menunjukkan bahwa siswa telah menerapkan aspek berpikir komputasional dalam menyelesaikan proyek pemrograman, namun tingkat penguasaan setiap aspek berbeda-beda. Aspek dekomposisi dan pengenalan pola telah terlihat dalam kemampuan siswa memecah masalah dan mengenali kesamaan langkah penyelesaian. Sementara itu, aspek abstraksi dan algoritma masih menjadi tantangan bagi sebagian siswa, terutama dalam menyaring informasi penting dan menyusun langkah penyelesaian yang sistematis. Hambatan yang ditemukan meliputi kesulitan memahami logika program, menentukan algoritma yang tepat, serta mengatasi kesalahan (error) dalam program. Untuk mengatasi hambatan tersebut, siswa menggunakan berbagai strategi seperti berdiskusi dengan teman, bertanya kepada guru, mencari referensi melalui internet, dan melakukan metode coba-coba (trial and error). Penelitian ini memberikan gambaran yang lebih mendalam mengenai proses berpikir komputasional siswa sehingga dapat menjadi dasar bagi guru dalam merancang pembelajaran informatika yang lebih efektif dan sesuai dengan kebutuhan siswa.

Kata Kunci: Berpikir Komputasional, Dekomposisi, Abstraksi, Algoritma, Proyek Pemrograman, Studi Kasus.

A. Pendahuluan

Kemampuan berpikir komputasional sangat diperlukan bagi manusia dalam menghadapi era Revolusi Industri 4.0. Eive (2021),	menjelaskan bahwa kemampuan abad ke-21 seperti kepemimpinan, kreativitas, literasi digital, komunikasi yang efektif, kecerdasan emosional, kewirausahaan, pemecahan
--	---

masalah, dan kerja sama merupakan kunci untuk dapat bertahan dan berkembang pada era tersebut. Sejumlah ahli ilmu komputer dan otoritas pendidikan juga menganggap berpikir komputasional sebagai salah satu bentuk literasi penting abad ke-21. Hal ini ditegaskan oleh Amri (2019), yang menyatakan bahwa berpikir komputasional merupakan keterampilan literasi baru yang wajib dimiliki oleh peserta didik di era digital. Menurut Voskoglou dan Buckley (2012), berpikir komputasional adalah cara berpikir baru dalam memecahkan masalah dengan menggunakan konsep-konsep ilmu komputer. Selanjutnya, Wing (2006), menegaskan bahwa kemampuan berpikir komputasional merupakan kemampuan fundamental yang seharusnya dimiliki oleh setiap orang, bukan hanya oleh ilmuwan computer. García-Peñalvo dkk (2018), menjelaskan bahwa kemampuan ini tidak hanya berfokus pada pemrograman, tetapi juga pada pengembangan pola pikir komputasional yang sistematis dan efisien. García-Peñalvo dkk (2018), menjelaskan bahwa kemampuan ini tidak hanya berfokus pada pemrograman, tetapi juga pada

pengembangan pola pikir komputasional yang sistematis dan efisien. Pendiri Apple, Steven Paul Jobs, bahkan menyatakan bahwa setiap orang seharusnya belajar memprogram komputer karena hal tersebut mengajarkan cara berpikir yang logis dan terstruktur. SMA Kristen Citra Bangsa sebagai institusi pendidikan yang berupaya menjawab tantangan abad ke-21 telah mengimplementasikan kurikulum yang menekankan pembelajaran berbasis praktik dan proyek pada mata pelajaran Informatika. Namun, hingga saat ini belum terdapat analisis mendalam mengenai bagaimana serta mengapa peserta didik kelas X mengaplikasikan keterampilan berpikir komputasional mereka dalam penyelesaian proyek pemrograman. Pemahaman kualitatif terhadap proses berpikir siswa mulai dari tahap perancangan ide, menghadapi kesulitan, hingga menemukan solusi merupakan kunci penting untuk mengevaluasi efektivitas proses pembelajaran secara menyeluruh dan holistik.

Berdasarkan data hasil belajar siswa, kemampuan berpikir komputasional siswa secara umum berada pada kategori baik dengan

rata-rata nilai 85–90. Namun, masih terdapat beberapa siswa yang memperoleh nilai 75, yang menunjukkan bahwa pemahaman terhadap konsep berpikir komputasional belum sepenuhnya optimal. Sebagian siswa masih cenderung mengikuti contoh atau menggunakan metode *trial and error* tanpa memahami logika penyelesaian masalah. Kondisi ini menunjukkan perlunya evaluasi dan pembelajaran yang lebih terstruktur untuk meningkatkan kemampuan dekomposisi, abstraksi, dan algoritma.

Data wawancara yang dilakukan peneliti selama Praktek Pegalaman Lapangan (PPL) bertujuan untuk menggali lebih dalam proses berpikir komputasional tidak selalu tercermin secara sepenuhnya melalui hasil tugas atau nilai proyek pemrograman yang mereka peroleh, karena penilaian tersebut belum tentu menggambarkan proses berpikir dan strategi pemecahan masalah yang digunakan selama penyelesaian tugas. Penelitian ini berupaya menggali lebih jauh proses, pengalaman, dan persepsi siswa dalam pembelajaran informatika dan proyek pemrograman di SMA Citra

Bangsa. Dengan pendekatan kualitatif, peneliti berusaha memahami secara menyeluruh bagaimana siswa menghadapi kesulitan, strategi apa yang mereka kembangkan, serta bagaimana faktor internal maupun eksternal memengaruhi proses berpikir mereka.

B. Metode penelitian

Metode Penelitian yang digunakan dalam penelitian ini adalah metode kualitatif dengan pendekatan studi kasus. Sementara itu, jenis penelitian yang digunakan adalah studi kasus (*case study*). Jenis penelitian ini digunakan karena peneliti ingin memusatkan perhatian pada satu kasus tertentu yaitu keterampilan berpikir komputasional siswa di satu kelas atau kelompok tertentu. Prosedur penelitian dilaksanakan dengan tiga teknik pengumpulan data, yaitu observasi terstruktur, wawancara, dan dokumentasi. Observasi terstruktur dilakukan untuk mengamati secara langsung aktivitas siswa ketika mengerjakan proyek pemrograman, Wawancara dilakukan secara semi-terstruktur kepada siswa dan guru TIK. Wawancara dengan siswa bertujuan menggali proses berpikir

yang tidak tampak dalam observasi, seperti alasan mereka memilih langkah tertentu. Sementara itu, wawancara dengan guru digunakan untuk mendapatkan perspektif tambahan tentang kemampuan, kelemahan, dan kebutuhan siswa dalam pembelajaran pemrograman. Dokumentasi digunakan sebagai data pelengkap berupa hasil proyek pemrograman siswa, catatan guru, serta foto atau rekaman kegiatan belajar yang dapat memverifikasi dan memperkuat hasil observasi serta wawancara.

C. Hasil Penelitian dan Pembahasan

Hasil Penelitian

Penelitian dilaksanakan pada 22 siswa kelas X SMA Kristen Citra Bangsa Mandiri Kupang melalui proyek pembuatan aplikasi kalkulator sederhana menggunakan bahasa pemrograman C++. Hasil penelitian menunjukkan bahwa sebagian besar siswa mampu menyelesaikan proyek dengan baik, mulai dari menyusun logika program, menuliskan sintaks C++, hingga menghasilkan program yang dapat berjalan tanpa kesalahan (error). Selain itu, beberapa siswa juga menunjukkan kreativitas dalam

merancang tampilan program berbasis teks.

Proyek yang diberikan dirancang untuk melatih empat pilar berpikir komputasional, yaitu dekomposisi, pengenalan pola, abstraksi, dan algoritma. Tahapan pengerjaan proyek meliputi identifikasi masalah, perancangan algoritma, penulisan kode, pengujian dan debugging, hingga penyelesaian akhir program. Bahasa pemrograman C++ dipilih karena mampu melatih ketelitian, logika berpikir, serta kemampuan pemecahan masalah siswa.

Berdasarkan hasil observasi, dokumentasi, wawancara, dan penilaian menggunakan rubrik berpikir komputasional, kemampuan siswa secara umum berada pada kategori berkembang hingga sangat baik. Aspek yang paling menonjol adalah pengenalan pola, sedangkan abstraksi dan penyusunan algoritma masih memerlukan penguatan. Hasil wawancara juga menunjukkan bahwa siswa memanfaatkan penjelasan guru, pengalaman belajar sebelumnya, diskusi, dan pencarian informasi melalui internet untuk menyelesaikan kendala yang dihadapi.

Analisis terhadap empat pilar berpikir komputasional menunjukkan bahwa:

- Dekomposisi berada pada kategori tinggi, karena sebagian besar siswa mampu memecah permasalahan menjadi bagian-bagian yang lebih sederhana.
- Pengenalan pola berada pada kategori baik, ditunjukkan dengan kemampuan menghubungkan proyek dengan latihan yang pernah dipelajari sebelumnya.
- Abstraksi berada pada kategori baik, karena siswa mampu memfokuskan perhatian pada informasi penting dalam penyelesaian proyek, meskipun sebagian masih terdistraksi oleh tampilan program.
- Algoritma juga berada pada kategori baik, ditunjukkan melalui kemampuan menyusun langkah-langkah penyelesaian secara sistematis serta melakukan debugging ketika terjadi kesalahan.

Pembahasan

1. Analisis Berpikir Komputasioal Siswa dalam Menyelesaikan Proyek Pemrograman.

Berdasarkan serangkaian aktivitas pengambilan data di laboratorium komputer, hasil penelitian ini secara nyata menunjukkan adanya variasi performa dan respon kognitif dari 22 siswa ketika diinstruksikan untuk mentransformasikan logika hitung matematika ke dalam baris kode program kaku.

- a. Proses Praktik dan Bimbingan: Melalui pengamatan langsung, aktivitas pembelajaran berbasis proyek
- b. Artefak Produk Digital: Program ini terbukti mampu beroperasi secara akurat dalam memproses data input numerik serta mencetak luaran hasil kalkulasi matematika (penjumlahan, pengurangan, perkalian, pembagian) secara presisi.
- c. Karakteristik Respon Wawancara: Dari sampel terstruktur, ditemukan empat pola strategi siswa dalam menyelesaikan proyek. Amelia dan Chlara menunjukkan kemampuan analisis mandiri yang matang dalam memecah masalah dan melakukan pelacakan kode secara runut saat *error*. Juan memperlihatkan strategi adaptif

dengan memodifikasi latihan lama dan mengeksplorasi internet. Sementara Keyzha mencerminkan pola belajar repetitif yang membutuhkan penjelasan verbal berulang dari guru akibat kebingungan konseptual di awal proyek.

B. Keterangan Hasil dengan Teori Computational Thinking

Apabila temuan hasil penelitian di atas dikorelasikan dengan postulat dasar teori *computational thinking*, tampak jelas bahwa aktivitas siswa di laboratorium komputer bukanlah sekadar kegiatan mengetik teks (*coding* mekanis), melainkan sebuah manifestasi konkret dari proses aktivasi mental atau pemecahan masalah (*problem-solving process*

C. Analisis Kemampuan Siswa Berdasarkan Indikator Computational Thinking

Empat pilar indikator utama, yaitu dekomposisi, pengenalan pola, abstraksi, dan algoritma:

- a. Dekomposisi (*Decomposition*): Siswa berhasil memenuhi indikator ini dengan kemampuan memecah komponen proyek kalkulator yang rumit ke dalam elemen-elemen Input (deklarasi variabel angka dan operator), Proses (logika rumus

hitung percabangan), dan Output (perintah cetak cout). Mayoritas siswa berada di kategori tinggi, walau beberapa mengalami hambatan kecil dalam mengurai kondisi batas seperti pembagian dengan angka nol (0).

- b. Pengenalan Pola (*Pattern Recognition*): Indikator ini terpenuhi saat siswa mampu mendeteksi kesamaan logika antar-operasi aritmatika dan mengaitkan proyek ini sebagai modifikasi atau penggabungan dari materi latihan yang pernah mereka terima sebelumnya. Siswa memanfaatkan pola struktur yang berulang ini untuk melakukan efisiensi pengetikan kode melalui teknik *copy-paste* di laboratorium.

- c. Abstraksi (*Abstraction*): Siswa terbukti mampu melakukan abstraksi dengan menyaring data esensial, seperti memilih tipe data pecahan (float/double) agar hasil pembagian akurat, serta membuang detail fungsi kalkulator ilmiah yang tidak diperlukan. Hambatan pada pilar ini hanya terjadi pada sebagian kecil siswa yang terlalu terdistraksi menghias visual tampilan menu ketimbang

menuntaskan logika utama program.

- d. Berpikir Algoritma (*Algorithmic Thinking*): Indikator ini tercapai dengan keberhasilan siswa menyusun urutan langkah pengerjaan program secara hierarkis dari atas ke bawah. Kematangan pilar ini diuji saat terjadi *error*, di mana siswa secara sistematis mengecek kembali kode, melacak letak kesalahan simbol (seperti tanda titik koma atau kurung kurawal), dan memperbaikinya secara bertahap sampai program berjalan benar.

Berdasarkan pemenuhan indikator tersebut, siswa dapat diklasifikasikan ke dalam tiga tingkatan kemampuan: kelompok Tinggi (mandiri dan solutif saat *error*), kelompok Sedang (memahami logika namun bergantung pada referensi internet/luar), dan kelompok Rendah (mengalami kebingungan struktural dan memerlukan bimbingan intensif/*scaffolding* dari guru).

D. Teori yang Digunakan (Computational Thinking)

Kerangka teoritis utama yang melandasi dan memayungi seluruh analisis dalam penelitian ini adalah Teori Berpikir Komputasional

(*Computational Thinking*). Dengan menerapkan landasan teori ini, peneliti dapat melihat secara jernih bahwa penguasaan empat pilar konseptual (dekomposisi, pengenalan pola, abstraksi, dan rancangan algoritma)..

2. Analisis Keterampilan Berpikir Komputasional Siswa pada Pilar Dekomposisi (*Decomposition*)

Pilar pertama dalam *computational thinking* yang dibedah dalam proyek pemrograman kalkulator sederhana menggunakan bahasa C++ ini adalah dekomposisi. Dekomposisi merupakan kemampuan kognitif siswa kelas X SMA Kristen Citra Bangsa Mandiri Kupang dalam mengurai satu kesatuan masalah besar yang kompleks dan abstrak menjadi sub-sub komponen masalah yang lebih kecil, sederhana, dan spesifik. Dengan memecah masalah menjadi bagian-bagian kecil, siswa dapat mengisolasi kerumitan sehingga program dapat dirancang secara

A. Konfirmasi Validitas Melalui Respons Wawancara Siswa

Hasil wawancara menunjukkan adanya variasi strategi penalaran yang sangat kaya di lapangan:

- a. Keyzha mencerminkan proses pemahaman bertahap yang

- sangat bergantung pada kehadiran guru. Ia menyatakan bahwa ia memahami masalah proyek “Dengan baik, walaupun kadang bingung, namun Guru dapat menjelaskan dengan baik”.
- b. Amelia menunjukkan pendekatan analisis mandiri pasca-pembelajaran. Ia mengungkapkan: “Saya menerima edukasi dari Guru lalu mencoba analisis permasalahannya.
- c. Juan memilih strategi dekomposisi berbasis eksplorasi multi-sumber guna memperjelas bagian-bagian kode yang rumit. Ia memahami masalah “Dengan cara belajar dari berbagai sumber baik Guru dan melalui internet agar mendapatkan jawaban”. Pencarian referensi luar ini membantu Juan mengurai komponen logika yang belum ia kuasai sepenuhnya.
- d. Chlara menerapkan teknik penajaman fokus yang sangat selaras dengan pilar dekomposisi, yaitu dengan mengisolasi variabel penting. Ia menjelaskan: “Cara saya memahaminya ialah dengan melihat kata kunci dari data yang dicarikan. Setelah itu saya

mencoba untuk mempraktekkannya...”. Chlara membuang narasi panjang pada soal tugas dan langsung membidik “kata kunci data” sebagai bagian kecil yang harus diselesaikan terlebih dahulu.

D. Evaluasi Tingkat Kemampuan dan Hambatan Dekomposisi Siswa

Melalui triangulasi data antara keberhasilan fungsi pada produk kalkulator C++, dinamika aktivitas di laboratorium, serta kedalaman jawaban wawancara, peneliti menyimpulkan bahwa tingkat kemampuan dekomposisi siswa kelas X secara umum berada pada kategori Tinggi. Mayoritas dari 22 siswa telah memiliki kemandirian kognitif yang matang untuk memisahkan struktur program secara mandiri tanpa mencampuradukkan antara data masukan (*input*) dan data luaran (*output*).

Namun demikian, peneliti masih menemukan adanya hambatan spesifik terkait pilar dekomposisi ini di lapangan. Hambatan tersebut muncul ketika siswa melakukan dekomposisi pada logika proses operasi pembagian. Sebagian siswa langsung memasukkan rumus

pembagian umum tanpa memecah masalah potensial yang akan terjadi jika pengguna memasukkan angka pembagi bernilai nol (0). Karena mereka tidak mengurai logika kondisi batas (*boundary condition*) tersebut, program mengalami kemacetan (*crash/runtime error*) saat diuji coba dengan angka nol.

Temuan hambatan ini menjadi catatan penting dalam pembahasan ini, yang menandakan bahwa meskipun kemampuan dekomposisi siswa sudah tinggi untuk alur normal, mereka masih memerlukan latihan intensif untuk melatih kepekaan dekomposisi hingga ke detail skenario pengecualian dalam pemrograman.

3. Analisis Keterampilan Berpikir Komputasional Siswa pada Pilar Pengenalan Pola (*Pattern Recognition*)

Pilar kedua dalam *computational thinking* yang dibahas dalam pelaksanaan proyek pemrograman kalkulator sederhana menggunakan C++ ini adalah pengenalan pola.

A. Kemampuan Mengenali Pola Masalah dan Struktur Logika Berulang

Berdasarkan temuan di lapangan, penalaran pengenalan

pola siswa mula-mula teridentifikasi ketika mereka menyadari sifat berulang dari operasi aritmatika. Meskipun kalkulator yang dirancang memiliki empat menu operasi hitung yang berbeda—yaitu penjumlahan, pengurangan, perkalian, dan pembagian—siswa berhasil menangkap bahwa cara kerja fundamental dari keempat operasi tersebut memiliki karakteristik struktur yang serupa.

Selama proses pengamatan langsung di laboratorium komputer, indikator fisik dari aktifnya pilar pengenalan pola ini terlihat sangat jelas pada perilaku penulisan kode (*coding execution*) siswa. Saat menyusun blok kode percabangan untuk menu pilihan menggunakan perintah kontrol *if-else* atau *switch-case*, sebagian besar siswa memanfaatkan fitur menyalin dan menempel (*copy-paste*). Setelah sukses menyusun susunan sintaksis yang valid untuk menu operasi penjumlahan, siswa dengan cekatan menyalin blok kode tersebut, menempelkannya di baris bawah, dan tinggal mengubah operator aritmatikanya untuk menu pengurangan, perkalian, dan pembagian.

B. Konfirmasi Validitas Melalui Respons Wawancara Siswa

Data hasil wawancara terstruktur bersama siswa kelas X memperkuat temuan observasi mengenai bagaimana siswa mengaitkan ingatan masa lalu mereka untuk mengenali pola pada proyek kalkulator ini:

- a. Keyzha mampu mengenali keterkaitan materi harian dengan proyek akhir secara akumulatif. Ketika merefleksikan kesamaan tugas ini dengan tugas sebelumnya, ia mengutarakan: *"Like proyek yang lama di gabung mejadi satu"*. Respons ini menunjukkan bahwa Keyzha mengenali proyek kalkulator sebagai sebuah muara yang mengintegrasikan pola-pola program sederhana yang pernah ia pelajari secara terpisah.
- b. Amelia menunjukkan kemampuan abstraksi pola yang sangat mendasar dan filosofis. Ia berhasil menangkap satu karakteristik inti yang menyatukan seluruh penugasan informatika, yaitu: *"Iya pakai Logika semua"*. Bagi Amelia, pola penyelesaian tugas baru tetap sama dengan tugas lama, yaitu bersandar pada

ketatnya alur penalaran logika komputer.

- c. Juan secara sadar mengenali bahwa proyek ini merupakan kontinuitas dari proses latihan harian di kelas. Ia menjelaskan: *"Iya, karena proyek yang sedang di kerjakan pasti dari latihan sebelumnya dan tinggal di modifikasi"*. Juan berhasil melihat pola hubungan langsung, di mana ia hanya perlu memanggil memori kognitifnya tentang kode latihan terdahulu lalu melakukan penyesuaian (modifikasi) kecil agar sesuai dengan fungsi kalkulator.
- d. Chlara mengonfirmasi adanya kebiasaan atau pola konsisten yang dibangun oleh guru dalam memberikan penugasan di laboratorium komputer. Ia menyatakan: *"Ada karena proyek yang kami dapat selalu tugas yang pernah diberikan"*. Chlara mengenali tren pembelajaran tersebut, sehingga ia merasa familier dan tidak merasa asing saat instruksi proyek kalkulator ini diberikan pertama kali.

D. Evaluasi Tingkat Kemampuan dan Hambatan Pengenalan Pola Siswa

Melalui triangulasi data dari kesesuaian produk program kalkulator C++, jalannya aktivitas praktik, serta hasil wawancara, peneliti menyimpulkan bahwa tingkat kemampuan pengenalan pola siswa kelas X berada pada kategori Baik. Sebanyak 22 siswa rata-ulang terbukti cakap dalam memanfaatkan kemiripan fungsi kontrol menu, sehingga proses pengerjaan logika pemrograman dapat diselesaikan secara terarah dan efisien.

Kendati demikian, peneliti tetap mencatat adanya hambatan spesifik terkait pilar pengenalan pola ini di lapangan. Hambatan terbesar yang dialami siswa muncul pada pola penulisan simbol operator aritmatika di dalam teks editor C++. Beberapa siswa mengalami kesalahan kompilasi (*compiler error*) karena reflex menerapkan pola kebiasaan menulis simbol matematika di buku tulis ke dalam komputer. Sebagai contoh, untuk proses perkalian mereka mengetik huruf x (bukan simbol bintang *), dan untuk operasi pembagian mereka mengetik tanda titik dua : (bukan simbol garis miring /).

4. Analisis Keterampilan Berpikir Komputasional Siswa pada Pilar Abstraksi (*Abstraction*)

Pilar ketiga dalam *computational thinking* yang dibedah dalam pelaksanaan proyek pemrograman kalkulator sederhana menggunakan bahasa C++ ini adalah abstraksi. Abstraksi merupakan kemampuan kognitif peserta didik kelas X SMA Kristen Citra Bangsa Mandiri Kupang dalam menyaring informasi, memfokuskan perhatian pada gagasan atau komponen yang paling penting dan esensial, serta mengabaikan detail-detail lain yang tidak relevan agar suatu permasalahan yang rumit menjadi lebih sederhana untuk diselesaikan. Kemampuan ini sangat krusial agar siswa tidak terjebak dalam kompleksitas data yang tidak mendesak selama proses penulisan kode program.

A. Kemampuan Menyederhanakan Masalah dan Memilih Data Esensial

Berdasarkan temuan di lapangan, penalaran abstraksi siswa mula-mula teridentifikasi ketika mereka mampu menyaring batasan masalah dari spesifikasi proyek kalkulator. Siswa menyadari bahwa mereka tidak perlu memasukkan seluruh fungsi kompleks yang ada pada kalkulator ilmiah (*scientific calculator*), melainkan cukup

mengambil esensi utama yang diminta dalam tugas, yaitu empat operasi aritmatika dasar (penjumlahan, pengurangan, perkalian, dan pembagian).

Selama pengamatan, siswa yang memiliki kemampuan abstraksi baik akan memilih tipe data pecahan seperti float atau double untuk variabel angka mereka, bukan tipe data bulat int. Aktivitas penalaran ini menunjukkan bahwa siswa berhasil menarik kesimpulan penting: jika kalkulator nantinya diuji dengan operasi pembagian bilangan ganjil, penggunaan tipe data pecahan sangat esensial agar komputer dapat mengeluarkan hasil desimal yang presisi dan menghindari *error* pembulatan ke bawah.

B. Konfirmasi Validitas Melalui Respons Wawancara Siswa

Data hasil wawancara terstruktur bersama siswa kelas X memperkuat temuan observasi mengenai bagaimana struktur kognitif mereka bekerja untuk menyaring kerumitan program dari sudut pandang pilar abstraksi:

a. Keyzha mengatasi kerumitan dengan cara memusatkan perhatian pada fondasi utama yang telah dipelajari. Ketika

dihadapkan pada masalah yang rumit, tindakan abstraksi yang ia ambil adalah: *"Lebih mengingat dasar yang sudah ada"*. Keyzha memilih untuk mengabaikan kebingungan dari detail kode yang panjang dan memfokuskan pikirannya kembali pada aturan-aturan dasar sintaksis C++ yang paling krusial.

b. Amelia menunjukkan strategi penyaringan masalah yang sangat sistematis. Untuk menyederhanakan kerumitan proyek, ia mengungkapkan: *"Saya membaginya menjadi beberapa langkah kecil supaya lebih mudah di pahami dan di kerjakan"*. Amelia membuang tumpukan kebingungan besar yang abstrak dan memilih fokus mengisolasi bagian-bagian esensial langkah demi langkah agar lebih mudah dicerna oleh logikanya.

c. Juan secara langsung mengincar inti dari struktur program demi membuang detail yang tidak mendesak. Ia menjelaskan caranya menyederhanakan masalah dengan: *"Mulai dari dasar bagian utama"*. Juan mengabaikan komponen-komponen tambahan yang

bersifat sekunder dan mendahulukan pengetikan blok kode paling krusial, seperti fungsi utama (*main function*) program kalkulatornya terlebih dahulu.

- d. Chlara menerapkan esensi murni dari pilar *abstraction* melalui teknik penajaman fokus yang sangat kuat. Ia menjelaskan bahwa cara dirinya menyederhanakan kerumitan proyek adalah: "*Dengan melihat inti dari masalah itu*". Chlara secara langsung menyaring seluruh informasi yang ia terima di laboratorium, mengabaikan detail-detail pendukung yang membingungkan, dan hanya mengambil "inti masalah" sebagai panduan utama dalam mengetik kode program.

C. Evaluasi Tingkat Kemampuan dan Hambatan Abstraksi Siswa

Melalui triangulasi data dari kesesuaian pemilihan tipe data pada produk program, jalannya aktivitas praktik, serta hasil wawancara, peneliti menyimpulkan bahwa tingkat kemampuan abstraksi siswa kelas X berada pada kategori Baik. Dari total 22 siswa yang diteliti, mayoritas besar terbukti sukses mengabaikan distraksi atau variabel visual yang

kompleks dan memilih mengarahkan fokus penalaran mereka sepenuhnya pada validitas alur data serta ketepatan fungsi matematika utama kalkulator

5. Analisis Keterampilan Berpikir Komputasional Siswa pada Pilar Berpikir Algoritma (*Algorithmic Thinking*)

Pilar keempat sekaligus muara dari implementasi *computational thinking* dalam proyek pembuatan kalkulator sederhana berbasis C++ ini adalah berpikir algoritma. Berpikir algoritma didefinisikan sebagai kemampuan kognitif siswa kelas X SMA Kristen Citra Bangsa Mandiri Kupang dalam menyusun, merancang, dan merumuskan langkah-langkah penyelesaian masalah secara logis, runtut, sistematis, dan bertahap ke dalam baris-baris instruksi yang dapat dipahami serta dieksekusi dengan benar oleh mesin komputer.

A. Kemampuan Menyusun Urutan Kode Program secara Runtut

Hasil pengamatan menunjukkan bahwa mayoritas siswa kelas X berhasil menyusun urutan pengerjaan kalkulator secara hierarkis dan logis dari atas ke

bawah, yang meliputi langkah-langkah berikut:

- a. Menginisialisasi *library* standar C++ (`#include <iostream>`).
- b. Mendeklarasikan variabel input dan output di dalam fungsi utama.
- c. Menampilkan menu pilihan operasi aritmatika ke layar teks.
- d. Menggunakan struktur kontrol percabangan (*if-else* atau *switch-case*) untuk memproses kalkulasi matematika berdasarkan pilihan pengguna.
- e. Mencetak hasil akhir pemrosesan secara dinamis di jendela terminal.

Ketepatan penyusunan prosedur bertahap ini menjadi bukti nyata di lapangan bahwa siswa tidak sekadar menyalin teks secara acak, melainkan memahami runtutan proses bagaimana sebuah data dialirkan dan diproses oleh komputer.

B. Aktivitas Pelacakan Kutu (*Debugging*) Selama Praktik

Selama proses pengamatan langsung di laboratorium komputer, indikator aktifnya pilar algoritma ini juga terlihat sangat jelas ketika siswa menghadapi kendala teknis berupa pesan kesalahan (*error*) saat mengompilasi program. Peneliti mencatat adanya aktivitas

penelusuran ulang kode yang sangat intensif.

Siswa tidak langsung menyerah atau membuat ulang program dari awal, melainkan melakukan tindakan jeda untuk membaca baris demi baris instruksi yang telah mereka ketik secara urut dari atas ke bawah. Aktivitas menguji coba jalannya program secara berulang dengan variasi input angka yang berbeda dilakukan siswa secara sadar demi melacak di mana letak kebocoran logika atau kesalahan sintaksisnya. Proses pelacakan mandiri (*debugging*) ini mencerminkan kematangan penalaran algoritmik siswa dalam mengevaluasi efektivitas langkah solutif yang mereka buat.

C. Konfirmasi Validitas Melalui Respons Wawancara Siswa

Data hasil wawancara terstruktur memperkuat temuan observasi mengenai bagaimana respons psikologis dan tindakan algoritmik siswa ketika mendapati urutan program yang mereka rancang mengalami kesalahan:

- a. Keyzha menunjukkan sikap ketelitian dan kehati-hatian yang tinggi dalam merunut ulang algoritmanya. Ketika programnya mendapati kendala, tindakan

- yang ia ambil adalah: “Mengevaluasi ulang dan kerjakan secara rinci”. Keyzha memilih untuk mundur satu langkah, meneliti kembali detail baris instruksinya, dan mengerjakannya ulang secara lebih saksama.
- b. Amelia mencerminkan strategi penanganan *error* yang sangat sistematis dan berulang (*looping action*). Ia menjelaskan: “Saya mengecek kembali kode yang saya tulis dan mencoba memperbaiki bagian yang salah sampai programnya berjalan dengan benar”. Amelia mengisolasi area kode yang bermasalah, melakukan perbaikan bertahap, dan langsung mengujinya kembali hingga komputer merespon sukses.
- c. Juan memilih memperluas ruang validasi logikanya dengan memanfaatkan sumber daya di sekitarnya saat algoritmanya macet. Ia mengungkapkan tindakannya: “Langsung mencari kesalahan dengan cara mencari jawaban dari berbagai sumber”. Juan menggunakan bantuan luar, seperti bertukar pikiran dengan guru maupun melakukan penelusuran mandiri via internet, untuk meluruskan struktur logika kodenya yang keliru.
- d. Chlara mengonfirmasi kemandirian kognitif yang senada dengan Amelia dalam menghadapi kendala teknis. Ia mengutarakan: “Saya cek kembali kode yang saya tulis dan mencoba memperbaiki bagian yang salah sampai programnya berjalan dengan benar”. Respons ini menunjukkan bahwa Chlara memiliki ketahanan berpikir algoritmik untuk melakukan pelacakan secara mandiri tanpa langsung bergantung pada intervensi guru.

D. Evaluasi Tingkat Kemampuan dan Hambatan Berpikir Algoritma Siswa

Melalui triangulasi data dari keberhasilan berjalannya fungsi pada produk aplikasi kalkulator, kedisiplinan alur ketikan selama praktik, serta ketajaman strategi penyelesaian masalah dari hasil wawancara, peneliti menyimpulkan bahwa tingkat kemampuan berpikir algoritma siswa kelas X berada pada kategori Baik. Ke-22 siswa rata-rata telah menunjukkan struktur berpikir yang runut dan logis dalam

menerjemahkan prosedur matematika ke dalam bahasa mesin.

D. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan pada bab sebelumnya mengenai analisis keterampilan berpikir komputasional (*computational thinking*) siswa kelas X SMA Kristen Citra Bangsa Mandiri Kupang dalam menyelesaikan proyek pemrograman kalkulator sederhana menggunakan bahasa C++, maka dapat ditarik beberapa kesimpulan utama sebagai berikut:

1. Keterampilan Dekomposisi Siswa

Siswa kelas X menunjukkan tingkat keterampilan dekomposisi yang berada pada kategori Tinggi. Hal ini dibuktikan dengan kemampuan mereka yang sangat baik dalam mengurai masalah proyek kalkulator yang kompleks ke dalam sub-komponen yang lebih kecil dan spesifik. Siswa sukses memetakan elemen logika program secara terstruktur mulai dari bagian *library*, area deklarasi variabel masukan data angka dan operator (Input), perumusan fungsi matematika percabangan (Proses), hingga perintah cetak teks hasil akhir

(Output). Hambatan minor hanya ditemukan pada ketidakpekaan mengurai logika kondisi batas (*boundary condition*), seperti skenario pengecualian ketika terjadi operasi pembagian dengan angka nol (0).

2. Keterampilan Pengenalan Pola Siswa

Siswa kelas X memiliki keterampilan pengenalan pola yang berada pada kategori Baik. Siswa terbukti cakap dalam mengidentifikasi tren, kesamaan karakteristik alur kerja operasi aritmatika harian, serta mendeteksi kemiripan struktur logika program kalkulator ini dengan latihan-latihan dasar pemrograman yang pernah mereka kerjakan sebelumnya. Kemampuan mengenali pola ini dimanfaatkan siswa secara efektif di laboratorium melalui aktivitas menyalin dan menempel (*copy-paste*) blok kode percabangan secara berulang untuk menghemat waktu pengetikan. Adapun hambatan yang muncul hanyalah kekeliruan reflek menuliskan simbol operator akibat masih terbawa pola tulisan matematika konvensional di buku tulis (seperti menulis $\times$ untuk perkalian atau $:$ untuk pembagian).

3. Keterampilan Abstraksi Siswa

Siswa kelas X menunjukkan keterampilan abstraksi yang berada pada kategori Baik. Kemampuan ini ditunjukkan melalui ketepatan siswa dalam menyaring informasi penting dan fokus hanya pada spesifikasi esensial proyek, yaitu kalkulator empat operasi aritmatika dasar, serta mengabaikan detail kalkulator ilmiah yang rumit. Siswa juga sukses melakukan penalaran abstrak saat memilih tipe data pecahan (float atau double) demi akurasi hasil pembagian bilangan ganjil. Hambatan spesifik pada pilar ini terjadi pada sebagian kecil siswa yang terlalu terdistraksi menghabiskan waktu praktik untuk menghias visual antarmuka teks (*User Interface*) daripada memprioritaskan logika utama program (*core logic*).

4. Keterampilan Berpikir Algoritma Siswa

Siswa kelas X memiliki keterampilan berpikir algoritma yang berada pada kategori Baik. Karakteristik kelogisan siswa terlihat dari kedisiplinan dan ketepatan mereka dalam menyusun baris-baris instruksi bahasa C++ secara hierarkis, runtut, dan sistematis dari atas ke bawah agar program dapat

dikompilasi tanpa kegagalan sistem. Kematangan pilar algoritma ini juga tecermin dari resiliensi dan ketahanan mental siswa dalam melakukan pelacakan kesalahan secara mandiri (*debugging*) ketika mendapati kode mereka *error*. Hambatan teknis yang konsisten dialami siswa pada pilar ini berpusat pada faktor ketidaktelitian penulisan detail simbolik bahasa pemrograman, seperti kelupaan meletakkan tanda titik koma (;) atau tertukarnya penempatan tanda kurung kurawa.

Saran

Berdasarkan kesimpulan dari hasil penelitian mengenai keterampilan berpikir komputasional (*computational thinking*) siswa kelas X, maka peneliti dapat mengajukan beberapa saran yang diharapkan dapat menjadi bahan masukan dan rekomendasi demi peningkatan mutu pembelajaran ke depan:

1. Bagi Guru Informatika

- a. Memberikan Latihan Kondisi Batas (*Boundary Condition*): Guru disarankan untuk memperkaya variasi studi kasus pemrograman yang melatih pilar dekomposisi siswa, tidak hanya untuk alur logika normal melainkan juga

untuk skenario pengecualian (seperti penanganan *error* ketika pengguna menginput angka nol pada operasi pembagian).

- b. Menekankan Aturan Sintaksis Komputer: Guru perlu memberikan penekanan dan pengingat yang intensif sejak awal praktikum mengenai perbedaan pola penulisan simbol matematika konvensional di buku tulis dengan pola penulisan simbol kaku pada bahasa pemrograman (seperti penggunaan operator *** dan */*) demi meminimalkan *compiler error*.

- c. Menerapkan Pembagian Waktu Praktik yang Proporsional: Guru diharapkan dapat membimbing siswa dalam hal manajemen prioritas pengerjaan proyek (pilar abstraksi), agar siswa dapat menuntaskan logika utama (*core logic*) program terlebih dahulu sebelum menghabiskan waktu untuk menghias estetika tampilan visual (*User Interface*).

2. Bagi Peserta Didik (Siswa)

- a. Meningkatkan Ketelitian dan Ketahanan Mental (*Resilience*): Siswa disarankan untuk terus melatih ketelitian dalam menuliskan detail simbolik bahasa

pemrograman (seperti ketepatan tanda titik koma ; dan kurung kurawal { }), serta tidak mudah panik ketika mendapati kodenya mengalami *error*.

- b. Membudayakan Praktik Mandiri (*Debugging*): Siswa diharapkan terbiasa melakukan pelacakan baris kode secara mandiri dan runtut (pilar algoritma) sebelum langsung bergantung pada bantuan guru, serta memanfaatkan ruang digital secara bijak untuk mengeksplorasi referensi tambahan.

3. Bagi Sekolah (SMA Kristen Citra Bangsa Mandiri Kupang)

Menjaga Kualitas dan Aksesibilitas Fasilitas Laboratorium: Pihak sekolah disarankan untuk terus mempertahankan dan meningkatkan kualitas infrastruktur teknologi, terutama stabilitas jaringan internet dan kondisi perangkat komputer di laboratorium, karena terbukti menjadi faktor eksternal yang sangat mendukung kenyamanan siswa dalam mengeksplorasi kemampuan berpikir komputasional mereka.

4. Bagi Peneliti Selanjutnya

Memperluas Konteks Proyek dan Metodologi: Peneliti selanjutnya

yang tertarik pada bidang *computational thinking* disarankan untuk memperluas ruang lingkup penelitian dengan menggunakan studi kasus proyek pemrograman yang berbeda, jumlah sampel yang lebih besar, atau menerapkan metode kombinasi (*mixed methods*) agar dapat menyajikan alat ukur kuantitatif yang lebih terstandarisasi.

DAFTAR PUSTAKA

- Maharani A. Computational Thinking dalam Pembelajaran Matematika Menghadapi Era Society 5.0. *Euclid*. 2020;7(2):86. doi:10.33603/e.v7i2.3364
- Herlina Budiarti, Teguh Wibowo, Puji Nugraheni. Analisis Berpikir Komputasional Siswa dalam Menyelesaikan Masalah Matematika. *J Pendidik Mipa*. 2022;12(4):1102-1107. doi:10.37630/jpm.v12i4.752
- Ansori M. Pemikiran Komputasi (Computational Thinking) dalam Pemecahan Masalah. *Dirasah J Stud Ilmu dan Manaj Pendidik Islam*. 2020;3(1):111-126. doi:10.29062/dirasah.3i1.83
- Pratama HY, Tobia MI, Saniyati SL, Yuginanda AS. Integrasi Computational Thinking Pada Mata Pelajaran Bahasa Indonesia Materi Pantun Kelas IV Sekolah Dasar. *J Penelitian, Pendidik dan Pengajaran JPPP*. 2023;4(1):68- 74. doi:10.30596/jppp.v4i1.14564
- Pramanik D, Desdimona •, Kumar R. Computational Thinking: A 21st Century Skill for Learners co01putational Thinking: A 21st Century Skill for Learners. *Comput Think A 21st Century Ski Learn* co01putational Think. 2025;(December 2023). <https://www.researchgate.net/publication/391189112>.
- García-Peñalvo F, Mendes A. Exploring the computational thinking effects in pre-university education. *Comput Human Behav*. 2017;80. doi:10.1016/j.chb.2017.12.005
- Andrian R, Hikmawan R. The Importance of Computational Thinking to Train Structured Thinking in Problem Solving. *J Online Inform*. 2021;6(1):113-117. doi:10.15575/join.v6i1.677
- Yana F, Wijaya A. Enhancing Computational Thinking Skills in Solving Two-Variable Linear Equations through Problem-Based Learning in Indonesian Middle Schools. *AL-ISHLAH J Pendidik*. 2025;17(1):614-631. doi:10.35445/alishlah.v17i1.5869
- Nur Ahzan Z, Wua Laja YP, Hijriani L. Efektivitas Pengembangan Media Bahasa Pemrograman Open-Source dalam Penerapan Computational Thinking Mahasiswa. *J Cendekia J Pendidik Mat*. 2023;7(2):1519-1532. doi:10.31004/cendekia.v7i2.2282