

Masalah Value Trace untuk pembacaan koding dalam Bahasa Pemrograman C

¹Yeni Risyani, ²Susi Japit, ³Tanda Selamat
^{1, 2, 3} Universitas IBBI, Indonesia

¹ms_yenir@yahoo.com, ²susijapit@gmail.com, ³tandaselamat@gmail.com

ABSTRAK

Banyak universitas di seluruh dunia mengajarkan pemrograman C sebagai bahasa komputer pengantar. Oleh karena itu, sangat penting bagi siswa pemula untuk terlibat dalam pembacaan ekstensif kode sumber C sederhana untuk memahami prinsip-prinsip pemrograman. Namun, ada kekurangan alat yang dirancang dengan baik untuk mendukung pembelajaran membaca kode secara mandiri di rumah. Sebelumnya, kami telah memperkenalkan konsep value trace problem (VTP) untuk pemrograman Java, di mana setiap contoh VTP terdiri dari kode sumber, beberapa pertanyaan, dan jawaban yang benar. (Klabnik & Nichols, 2023) Pertanyaan-pertanyaan ini menanyakan tentang nilai variabel kritis atau pesan keluaran dalam kode sumber, dengan respons siswa yang langsung divalidasi melalui pencocokan string untuk belajar mandiri. Dalam makalah ini, kami memperkenalkan VTP untuk belajar mandiri membaca kode dalam pemrograman C. Kami membuat 42 contoh VTP berdasarkan kode sumber C langsung yang mencakup konsep tata bahasa dasar dan struktur data fundamental serta algoritma yang ditemukan dalam buku teks dan sumber daya online. Selain itu, untuk contoh kasus yang menantang yang melibatkan pointer dan algoritme yang rumit, kami menyertakan petunjuk, pertanyaan pilihan ganda, dan referensi untuk meningkatkan akurasi solusi. Untuk menilai pendekatan kami, kami meminta 49 mahasiswa S1 dari Jepang, Cina, dan Myanmar untuk menyelesaikan contoh-contoh ini secara mandiri di rumah. Tingkat akurasi rata-rata mereka adalah 94,29%, dengan fitur-fitur pendukung kami yang meningkatkan akurasi sebesar 33,26% pada kasus-kasus yang sulit. Dengan demikian, proposal kami secara efektif mendorong pembelajaran mandiri dalam pemrograman C di kalangan siswa pemula. (Aho et al., 2023)

Keywords : C programming, Algorithm, Value trace problem

PENDAHULUAN

Saat ini, pemrograman C banyak diajarkan di berbagai universitas di seluruh dunia sebagai bahasa pemrograman komputer awal. Selain jurusan TI, berbagai jurusan lain seperti sains, pertanian, dan teknik mesin/listrik juga menawarkan kursus pemrograman C. Faktanya, mempelajari pemrograman C dapat memberikan pengetahuan dasar untuk mempelajari bahasa yang lebih canggih dan praktis seperti Java, JavaScript, dan Python. Selain itu, siswa di departemen TI harus secara bersamaan mempelajari pemrograman C bersama dengan arsitektur komputer untuk mendapatkan wawasan tentang memori dan akses register. Selain itu, terlepas dari usianya, C tetap menjadi bahasa pemrograman terpopuler ketiga. (Nijkamp et al., 2022) Oleh karena itu, sangat penting bagi siswa pemula untuk membaca dan memahami banyak kode sumber sederhana untuk mengenal struktur pemrograman C. Sayangnya, sepengetahuan kami, alat yang efektif yang mendukung pembelajaran mandiri membaca kode di rumah belum dirancang dengan baik. (Xu et al., 2022)

Hingga saat ini, kami telah memperkenalkan dan mengembangkan Java Programming Learning Assistant System (JPLAS) untuk mempelajari pemrograman Java. Sebagai bahasa

pemrograman berorientasi objek, Java digunakan secara luas di komunitas IT. Untuk membantu pembelajaran pemrograman Java di berbagai tingkat kemahiran, JPLAS menawarkan berbagai jenis soal latihan dengan berbagai tingkat kesulitan. Untuk setiap jenis soal, respons siswa terhadap pertanyaan secara otomatis dinilai oleh sistem untuk memfasilitasi pembelajaran pemrograman mandiri. Salah satu sumber daya ini adalah Value Trace Problem (VTP), yang dirancang untuk membantu siswa pemula memahami konsep dan keterampilan pemrograman dasar melalui latihan membaca kode. Contoh VTP mencakup satu kode sumber, beberapa pertanyaan, dan string jawaban yang benar. Setiap pertanyaan berkaitan dengan nilai variabel penting atau output yang signifikan dalam kode yang disediakan. Respons siswa segera dievaluasi keakuratannya oleh antarmuka jawaban di browser web, yang menggunakan pencocokan string terhadap string jawaban yang benar. (Hauser et al., 2023)

Kompilasi manual dari kode sumber C dan pemilihan variabel atau pesan keluaran dari setiap kode sumber untuk pertanyaan dapat menjadi kendala bagi VTP. Ketika seorang guru mengintegrasikan VTP ke dalam kursus pemrograman, mereka harus secara cermat mengumpulkan kode sumber dan memilih variabel atau pesan, bersama dengan waktunya, yang nilai atau konten aktualnya perlu ditelusuri dalam pertanyaan. (Mughnyanti & Ginting, 2023) Hal ini memastikan sinkronisasi dengan perkembangan kursus. Salah satu metode yang efisien bagi pengajar untuk membuat contoh VTP adalah dengan memanfaatkan contoh kode sumber yang disediakan bersama buku teks kursus, yang sering kali dapat diunduh dari situs web. Selanjutnya, mereka dapat memilih variabel atau pesan keluaran yang selaras dengan topik pengajaran yang tercakup dalam setiap sesi kelas. Keterbatasan lain dari VTP muncul dari ketidakmampuan siswa untuk berlatih menulis kode sumber, yang dapat diatasi dengan memasukkan jenis masalah lain, seperti latihan menulis kode. (Wayahdi et al., 2021)

TINJAUAN PUSTAKA

Pada bagian ini, kami meninjau karya-karya terkait dalam literatur. Beberapa makalah telah memperkenalkan alat bantu belajar pemrograman, sementara yang lain menyoroti pentingnya membaca kode dalam belajar pemrograman. (Ginting & others, 2024) Dalam sebuah penelitian, penulis mengulas berbagai alat yang dirancang untuk mendukung pendidikan pemrograman dan mengklasifikasikannya ke dalam empat kategori: 1) alat bantu dengan lingkungan pengembangan yang disederhanakan, 2) lingkungan berbasis contoh, 3) alat bantu yang menggunakan visualisasi dan animasi, dan 4) lingkungan simulasi. Proposal kami masuk ke dalam kategori 2. Dalam penelitian lain, penulis memperkenalkan QuizPACK, sebuah alat yang mirip dengan Value Trace Problem (VTP), dan menunjukkan bahwa alat ini secara signifikan meningkatkan pemahaman tentang semantik. (Ginting, 2023) Sebuah pertanyaan di QuizPACK meminta siswa untuk menentukan nilai ekspresi tertentu (biasanya variabel) dalam fragmen program. Guru harus secara manual menyiapkan satu set soal dengan menggunakan fragmen kode sumber dan ekspresi ini. Dibandingkan dengan VTP, keunggulan QuizPACK adalah kemampuannya untuk menghasilkan konstanta secara dinamis yang terkait dengan variabel yang ditanyakan dan jawaban yang benar, sehingga memungkinkan siswa yang berbeda untuk menemukan pertanyaan yang sama dengan jawaban yang benar yang berbeda-beda. Sebaliknya, VTP menggunakan konstanta tetap dan jawaban yang benar. Namun, keterbatasan QuizPACK adalah bahwa ia hanya dapat menanyakan nilai akhir variabel dalam fragmen kode dan tidak dapat menanyakan nilai pada tahap program yang berbeda. Hal ini membuat QuizPACK kurang cocok untuk mempelajari algoritma, di mana algoritma yang berbeda dapat menghasilkan nilai variabel akhir yang sama. (Gabbrielli & Martini, 2023)

Dalam penelitian lain, penulis memperkenalkan LECGO (Lingkungan Belajar untuk pemrograman menggunakan C dengan Objek Geometris), sebuah lingkungan pemecahan masalah yang dirancang untuk pemula yang belajar pemrograman C. LECGO menekankan beberapa aspek utama: (a) memanfaatkan berbagai representasi eksternal untuk meningkatkan pembelajaran siswa, (b) memotivasi siswa melalui kegiatan pemecahan masalah yang didasarkan pada konteks yang

akrab dan bermakna, (c) mendorong partisipasi aktif siswa melalui pengalaman langsung, (d) memberikan umpan balik yang tepat untuk memfasilitasi koreksi diri, dan (e) menawarkan konten yang holistik, berbasis aktivitas, multi-media, multi-representasi, dan berlapis-lapis untuk mempelajari konsep-konsep dasar. (Hagino, 2020)

Dalam penelitian lain, penulis memperkenalkan GradeIT, sebuah sistem yang dirancang untuk melayani tujuan ganda yaitu penilaian otomatis dan perbaikan program untuk kursus pemrograman pengantar. GradeIT mengevaluasi kode sumber yang dikirimkan berdasarkan jumlah kasus uji yang lulus, kebalikan dari waktu yang dibutuhkan untuk menyelesaikan masalah, dan sebagian kecil dari kompilasi yang berhasil. Skor kompilasi dihitung sebagai $1 - (\text{jumlah kesalahan kompilasi dalam kode}) / (\text{jumlah maksimum kesalahan kompilasi di antara siswa})$. Untuk perbaikan program, GradeIT menerapkan aturan penulisan ulang yang sederhana untuk memperbaiki kesalahan waktu kompilasi yang umum dan langsung. Dalam penelitian yang berbeda, penulis mengusulkan LAPLE (Learning Analytics in Programming Language Education), sebuah sistem plug-in untuk Moodle yang dirancang untuk menyediakan dasbor pembelajaran yang melacak perilaku siswa di dalam kelas dan mengidentifikasi kesulitan yang dihadapi oleh siswa yang berbeda di berbagai bidang pengetahuan. LAPLE mengharuskan siswa untuk menulis kode sumber lengkap dan mengumpulkan serta menganalisis log kompilasi mereka setiap lima menit, memberikan umpan balik visual secara real-time selama kelas berlangsung. Para penulis memeriksa distribusi dari 36 jenis kesalahan yang diklasifikasikan dan mendorong siswa untuk fokus pada kesalahan yang paling sering terjadi. Sementara LAPLE ditujukan untuk digunakan di kelas pemrograman, VTP dirancang untuk belajar mandiri di rumah.

METODE PENELITIAN

Definisi Soal Penelusuran Nilai, Misi dari VTP adalah agar siswa menelusuri nilai dari variabel kritis atau output yang signifikan dalam kode sumber yang diberikan dengan membaca dan memahami perilakunya dengan cermat. Untuk membuat contoh VTP, seseorang harus menyiapkan kode sumber, satu set pertanyaan dengan kolom jawaban, dan string jawaban yang benar. Setiap pertanyaan menanyakan nilai saat ini dari variabel atau pesan penting dalam kode sumber seperti yang ditentukan oleh guru. Oleh karena itu, kode sumber harus menyertakan pernyataan keluaran standar yang sesuai dengan variabel-variabel ini, sehingga siswa dapat dengan mudah mengidentifikasi nilai yang ditanyakan. Tujuan desain untuk VTP dalam pemrograman C adalah sebagai berikut: 1) Berbagai kode sumber C disediakan untuk pemula untuk latihan membaca kode. 2) Siswa dapat menyelesaikan pertanyaan dalam contoh VTP dengan membaca kode sumber yang diberikan dengan cermat dan memahami perilakunya. 3) Jawaban siswa secara otomatis diperiksa melalui pencocokan string pada antarmuka, dan hasilnya langsung dikembalikan kepada siswa. 4) Guru dapat memilih kode sumber yang sesuai dalam hal konten dan tingkat kesulitan untuk setiap contoh VTP yang ditugaskan dalam kursus pemrograman.

Instance VTP baru dapat dibuat dengan menggunakan prosedur berikut: 1. Pilih kode sumber yang sesuai untuk mempelajari konsep tata bahasa dasar atau struktur/algorithm data dasar. 2. Variabel-variabel penting atau pesan-pesan keluaran yang akan dilacak dalam kode sumber diidentifikasi. 3. Pernyataan keluaran standar yang relevan dimasukkan ke dalam kode. 4. Pertanyaan yang menanyakan nilai/pesan pada output standar dan jawaban yang benar dirumuskan. 5. Kode sumber, bersama dengan pertanyaan dan jawaban yang benar, dikonsolidasikan ke dalam satu file teks input. 6. Program yang diuraikan dalam dieksekusi menggunakan file teks masukan untuk menghasilkan file CSS/HTML/JavaScript untuk antarmuka jawaban pada browser web. 7. Contoh VTP yang dihasilkan diberikan kepada siswa sebagai tugas.

Antarmuka jawaban berbasis web browser digunakan bagi siswa untuk menyelesaikan contoh VTP yang dihasilkan dalam makalah ini. Fungsi-fungsi penting seperti pencocokan string untuk evaluasi jawaban dan menyimpan jawaban yang dikirimkan diimplementasikan dalam JavaScript, memungkinkan penggunaan antarmuka ini secara offline. Untuk menggunakan antarmuka jawaban ini secara efektif, jawaban yang benar untuk pertanyaan harus diberikan kepada

siswa sebelumnya, sehingga jawaban mereka dapat dievaluasi secara offline. Untuk mencegah siswa mengakses jawaban yang benar sebelum mencoba pertanyaan, nilai hash mereka dibuat menggunakan SHA256 sebelum didistribusikan. Secara khusus, setiap jawaban yang benar digabungkan dengan tugas dan ID soal sebelum di-hash, memastikan bahwa jawaban yang benar yang sama untuk pertanyaan yang berbeda menghasilkan nilai hash yang berbeda. Pendekatan ini mencegah pengungkapan jawaban yang benar melalui nilai hash saja. Selama evaluasi, setiap jawaban siswa digabungkan dan di-hash menggunakan SHA256 untuk dibandingkan dengan jawaban benar yang telah di-hash.

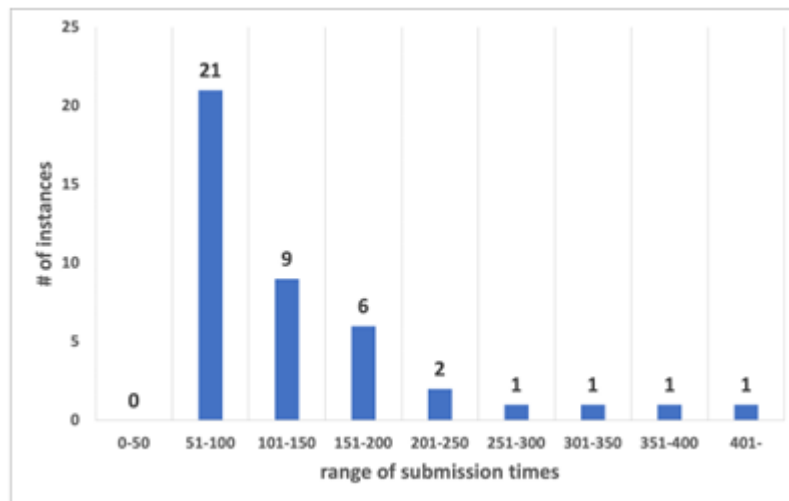
HASIL DAN PEMBAHASAN

Pada bagian ini, kami menilai 42 contoh VTP yang dihasilkan, yang terdiri dari 586 formulir jawaban, dengan memberikannya kepada 49 mahasiswa sarjana di Myanmar (21 mahasiswa), Jepang (18 mahasiswa), dan Cina (10 mahasiswa), yang semuanya saat ini sedang mengikuti atau telah menyelesaikan mata kuliah pemrograman C dan struktur & algoritma data dasar. Pada awalnya, hasil solusi untuk 42 contoh VTP individual yang dicoba oleh 49 siswa diperiksa. Gambar tersebut mengilustrasikan jumlah keseluruhan pengiriman jawaban dan persentase rata-rata jawaban yang benar untuk setiap contoh VTP yang dikirimkan oleh para siswa. Seperti yang dirangkum dalam Tabel 2, nilai maksimum, minimum, rata-rata, dan standar deviasi (SD) dari jumlah total pengajuan jawaban dan rata-rata tingkat jawaban benar di 42 contoh VTP disajikan. Tingkat rata-rata jawaban benar sebesar 94,29% dan rata-rata jumlah pengiriman 139,17 oleh semua siswa (2,84 pengiriman per siswa) untuk setiap contoh mengindikasikan bahwa contoh VTP yang dibuat memiliki tingkat kesulitan yang moderat untuk pelajar pemula dalam pemrograman C.

Tabel 1: Ringkasan hasil solusi untuk masing-masing contoh VTP.

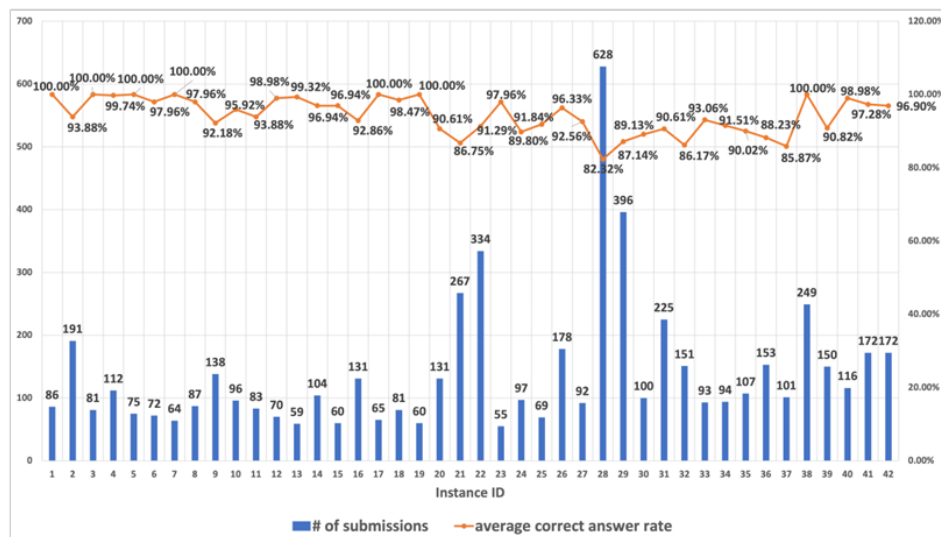
	total# of submissions	average correct rate
maximum	628	100 %
minimum	55	82.32 %
average	139.17	94.29%
SD	106.14	4.84%

Gambar ini menggambarkan distribusi frekuensi jumlah pengiriman jawaban di seluruh contoh VTP. Dua puluh satu contoh kasus diselesaikan dalam 100 kali pengiriman, sementara 30 contoh kasus diselesaikan dalam 150 kali pengiriman atau rata-rata 3,57 kali pengiriman per siswa. Namun, untuk tiga contoh dengan ID 22, 28, dan 29, jawaban dikirimkan lebih dari 300 kali, yang mengindikasikan adanya kebutuhan untuk perbaikan.



Gambar 1: Distribusi waktu pengajuan di antara beberapa contoh.

Hasil Solusi untuk Setiap Siswa, Setelah itu, hasil solusi untuk setiap siswa mengenai 42 contoh VTP diperiksa. Gambar 15 mengilustrasikan jumlah total pengajuan jawaban dan tingkat jawaban benar rata-rata di semua contoh untuk setiap siswa. Tabel 1 memberikan ringkasan hasil untuk setiap siswa. Temuan ini menunjukkan bahwa mayoritas siswa berhasil menyelesaikan contoh kasus VTP.



Gambar 2: Hasil solusi untuk masing-masing contoh VTP

KESIMPULAN

Dalam penelitian ini, Value Trace Problem (VTP) diperkenalkan untuk pemahaman kode mandiri dalam pemrograman C. Empat puluh dua contoh VTP dibuat dengan menggunakan kode sumber C langsung yang mencakup konsep tata bahasa dasar dan struktur data & algoritma penting yang ditemukan di buku teks atau sumber daya online. Selain itu, strategi seperti tips, petunjuk, pertanyaan pilihan ganda, dan referensi ke situs web eksternal disediakan untuk meningkatkan kinerja saat menangani contoh-contoh yang menantang yang melibatkan pointer dan algoritma. Untuk tujuan penilaian, 42 contoh kasus VTP yang dihasilkan didistribusikan kepada 49 mahasiswa di Jepang, Tiongkok, dan Myanmar, yang ditugaskan untuk menyelesaikannya secara mandiri di

rumah dengan menggunakan antarmuka jawaban yang dapat diakses secara offline. Tingkat jawaban benar rata-rata di seluruh contoh VTP adalah 94,29%, dan dengan penerapan peningkatan yang kami sarankan, tingkat rata-rata untuk empat contoh soal yang menantang meningkat 33,26% dari 57,80% pada awalnya. Hasilnya, proposal kami berhasil mendorong pembelajaran pemrograman C secara mandiri di kalangan siswa pemula di rumah. Salah satu kendala VTP adalah pemilihan variabel atau pesan keluaran dari kode sumber untuk pertanyaan secara manual, serta pengumpulan kode sumber C secara manual. Akan sangat bermanfaat untuk mengeksplorasi algoritme yang dapat secara otomatis menghasilkan contoh VTP baru dari kode sumber yang disediakan. Keterbatasan lain muncul dari ketidakmampuan siswa untuk berlatih menulis kode sumber sendiri.

Dalam upaya kami di masa depan, kami bertujuan untuk mengeksplorasi algoritme untuk menghasilkan instance VTP secara otomatis, membuat instance baru menggunakan kode sumber C untuk berbagai konsep tata bahasa dan algoritme seperti file I/O, penanganan kesalahan, rekursi, dan algoritme grafik, dan mengintegrasikannya ke dalam kursus pemrograman C untuk digunakan oleh siswa. Selain itu, kami berencana untuk melakukan studi yang komprehensif dan formal tentang pengorganisasian eksperimen pedagogis untuk menilai efektivitas proposal dalam meningkatkan motivasi siswa pemula untuk belajar pemrograman C. Selain itu, kami akan mengembangkan metodologi untuk menganalisis hasil dari eksperimen pedagogis ini.

REFERENSI

- Aho, A. V, Kernighan, B. W., & Weinberger, P. J. (2023). *The AWK programming language*. Addison-Wesley Professional.
- Gabrielli, M., & Martini, S. (2023). *Programming languages: principles and paradigms*. Springer Nature.
- Ginting, S. H. N. (2023). The Utilization Of The Simple Multi Attribute Rating Exploiting Ranks Can Enhance The Performance Of The Aco Algorithm. *Jurnal Minfo Polgan*, 12, 1325. <https://doi.org/doi.org/10.33395/jmp.v12i1.12743>
- Ginting, S. H. N., & others. (2024). Penerapan Algoritma k-means dalam Data Mining untuk Mengidentifikasi Strategi Promosi di Politeknik Ganesha Medan. *Jurnal Minfo Polgan*, 13(1), 189–196.
- Hagino, T. (2020). A categorical programming language. *ArXiv Preprint ArXiv:2010.05167*.
- Hauser, F., Häberle, M., Merling, D., Lindner, S., Gurevich, V., Zeiger, F., Frank, R., & Menth, M. (2023). A survey on data plane programming with p4: Fundamentals, advances, and applied research. *Journal of Network and Computer Applications*, 212, 103561.
- Klabnik, S., & Nichols, C. (2023). *The Rust programming language*. No Starch Press.
- Mughnyanti, M., & Ginting, S. H. N. (2023). Data Mining Manhattan Distance dan Euclidean Distance Pada Algoritma X-Means Dalam Klasifikasi Minat dan Bakat Siswa. *REMIK: Riset Dan E-Jurnal Manajemen Informatika Komputer*, 7(1), 835–842.
- Nijkamp, E., Pang, B., Hayashi, H., Tu, L., Wang, H., Zhou, Y., Savarese, S., & Xiong, C. (2022). Codegen: An open large language model for code with multi-turn program synthesis. *ArXiv Preprint ArXiv:2203.13474*.
- Wayahdi, M. R., Ginting, S. H. N., & Syahputra, D. (2021). Greedy, A-Star, and Dijkstra's algorithms in finding shortest path. *International Journal of Advances in Data and Information Systems*, 2(1), 45–52.
- Xu, F. F., Alon, U., Neubig, G., & Hellendoorn, V. J. (2022). A systematic evaluation of large language models of code. *Proceedings of the 6th ACM SIGPLAN International Symposium on Machine Programming*, 1–10.