**LAPORAN PRAKTIKUM STRUKTUR DATA**

**MODUL I**

# PENGENALAN CODE BLOCKS



**Disusun Oleh :**

Andera Singgih Pratama 2211104007

**Dosen**

Diah Septiani S.Kom M.Cs

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK**

**FAKULTAS INFORMATIKA**

# TELKOM UNIVERSITY PURWOKERTO

**2025**

1. Dasar Teori

Pengajaran pemrograman untuk pemula merupakan area penelitian yang mendapat perhatian luas, mencakup isu-isu seperti pemilihan bahasa yang sesuai, strategi pengajaran, serta desain kurikulum yang mendukung pembelajaran bertahap. Kajian literatur menunjukkan bahwa pemahaman konteks riset ini sangat penting agar pendidik dapat menyusun pengalaman belajar yang efektif sekaligus berlandaskan bukti empiris [1].

Dalam penyusunan materi ajar, dua komponen kunci yang sering menjadi fokus adalah Abstract Data Type (ADT) dan fungsi. ADT, yang dapat direpresentasikan melalui struct dalam C++, memberikan kerangka untuk mengorganisasi data agar lebih terstruktur. Di sisi lain, fungsi berperan dalam membagi persoalan kompleks menjadi bagian-bagian yang lebih kecil, terukur, dan dapat digunakan kembali. Kedua konsep ini berfungsi sebagai instrumen penting dalam mengurangi kompleksitas pemrograman, namun sering kali justru menjadi tantangan awal bagi mahasiswa yang baru belajar [2].

Selain itu, penguasaan struktur kontrol seperti percabangan (conditional statement) merupakan fondasi utama logika pemrograman. Penggunaan if-else memungkinkan mahasiswa mengatur alur program berdasarkan kondisi tertentu. Sejumlah penelitian melaporkan bahwa kesulitan transisi dari sekadar memahami sintaks ke penerapan logika percabangan untuk menyelesaikan masalah nyata adalah salah satu hambatan terbesar dalam tahap awal pembelajaran pemrograman [3].

1. Guided

Guided 1

A screenshot of a computer program

AI-generated content may be incorrect.

Screenshots Output

A black screen with white text

AI-generated content may be incorrect.

Program ini menunjukkan cara kerja *pointer* dalam C++. Variabel x diisi dengan nilai 87, kemudian alamatnya disimpan dalam pointer px. Dengan menggunakan operator *dereference* (\*px), nilai dari x dapat diakses melalui *pointer* dan disalin ke variabel y. Program juga menampilkan alamat memori x, isi dari pointer px (alamat x), serta nilai dari x, \*px, dan y.

Guided 2

A screen shot of a computer program

AI-generated content may be incorrect.

Screenshots Output

A screenshot of a computer

AI-generated content may be incorrect.

Program ini mendemonstrasikan penggunaan array satu dimensi dan array dua dimensi dalam C++. Array satu dimensi *nilai* diisi melalui input pengguna, lalu ditampilkan kembali sebagai data nilai siswa. Sementara itu, array dua dimensi *nilai\_tahun* yang sudah berisi data bawaan dalam kode ditampilkan dalam bentuk matriks menggunakan *nested loop* (perulangan bersarang).

1. Unguided

Unguided 1

A screen shot of a computer code

AI-generated content may be incorrect.

Screenshots Output

A black background with white text

AI-generated content may be incorrect.

Program ini memperlihatkan penggunaan *struct* dalam C++ untuk mengelola data nilai mahasiswa. Sebuah *struct* bernama *NilaiSTD* didefinisikan dengan beberapa atribut, yaitu *clo1*, *clo2*, *clo3*, *clo4*, *nilaiAkhir*, dan *indeks*. Pada fungsi main, dibuat sebuah variabel *struct* bernama mahasiswa\_contoh yang digunakan untuk menyimpan nilai *CLO 1* melalui *input* pengguna. Nilai yang sudah dimasukkan kemudian ditampilkan kembali ke layar.

Unguided 2

A screen shot of a computer code

AI-generated content may be incorrect.

Screenshots Output

A black screen with white text

AI-generated content may be incorrect.

Program ini mendemonstrasikan penggunaan *struct* dalam C++ untuk menyimpan data nilai mahasiswa. Sebuah *struct* bernama *NilaiSTD* memiliki beberapa atribut, yaitu *clo1*, *clo2*, *clo3*, *clo4*, *nilaiAkhir*, dan *indeks*. Pada fungsi main, dibuat sebuah variabel *struct* bernama mahasiswa\_contoh. Program meminta pengguna memasukkan nilai untuk *CLO 1*, kemudian nilai tersebut disimpan ke dalam atribut clo1 dan ditampilkan kembali ke layar sebagai output.

Unguided 3

A screen shot of a computer code

AI-generated content may be incorrect.

Screenshots Output

A screen shot of a computer

AI-generated content may be incorrect.

Program ini digunakan untuk menerima *input* nilai dari empat *Course Learning Outcome (CLO)* mahasiswa. Empat variabel bertipe *float* dideklarasikan untuk menyimpan nilai masing-masing *CLO*. Program meminta pengguna memasukkan nilai satu per satu melalui perintah cin, kemudian menampilkannya kembali di layar menggunakan cout. Dengan demikian, program memperlihatkan proses dasar *input* dan *output* data pada C++ menggunakan tipe data numerik.

Unguided 4,5

A screen shot of a computer program

AI-generated content may be incorrect.

Screenshots Output

A black screen with white text

AI-generated content may be incorrect.

Program ini menunjukkan penggunaan *struct* dan fungsi dalam C++. Sebuah *struct* bernama *NilaiSTD* didefinisikan untuk menyimpan nilai empat *Course Learning Outcome (CLO)*, nilai akhir, serta indeks huruf. Fungsi gabungNilaiStd dibuat untuk menerima empat nilai *CLO* sebagai parameter, lalu mengembalikan sebuah objek *NilaiSTD* dengan nilai-nilai tersebut. Pada fungsi main, pengguna diminta memasukkan empat nilai *CLO*. Data tersebut kemudian diproses oleh fungsi gabungNilaiStd dan hasilnya disimpan dalam variabel mahasiswa. Program menampilkan pesan bahwa nilai berhasil disimpan, sekaligus menampilkan salah satu nilai *CLO* yang tersimpan dalam *struct*.

Unguided 6

A screen shot of a computer program

AI-generated content may be incorrect.

Screenshots Output

A screen shot of a computer

AI-generated content may be incorrect.

Program ini memperlihatkan penggunaan *struct* dan fungsi untuk menghitung nilai akhir mahasiswa berdasarkan bobot masing-masing *Course Learning Outcome (CLO)*. Sebuah *struct* bernama *NilaiSTD* didefinisikan dengan empat atribut *CLO*, nilai akhir, serta indeks huruf. Fungsi hitungNilaiAkhir menerima parameter berupa objek *NilaiSTD*, kemudian menghitung nilai akhir dengan rumus: 30% untuk *CLO 1*, 30% untuk *CLO 2*, 20% untuk *CLO 3*, dan 20% untuk *CLO 4*. Pada fungsi main, pengguna diminta untuk memasukkan empat nilai *CLO* melalui *input*. Data tersebut diproses oleh fungsi hitungNilaiAkhir dan hasilnya ditampilkan ke layar dengan format dua angka di belakang koma. Program ini sekaligus menunjukkan penerapan *struct*, fungsi, operasi aritmatika berbobot, serta format keluaran numerik dalam C++.

Unguided 7

A computer screen shot of a program code

AI-generated content may be incorrect.

Screenshots Output

A black background with white text

AI-generated content may be incorrect.

Program ini menunjukkan penggunaan *struct* dan fungsi dalam C++. Sebuah *struct* bernama *NilaiSTD* dibuat untuk menyimpan empat nilai *Course Learning Outcome (CLO)*, nilai akhir, serta indeks huruf. Fungsi gabungNilaiStd berfungsi untuk menerima empat nilai *CLO* sebagai parameter, kemudian mengembalikan sebuah objek *NilaiSTD* dengan data tersebut, sementara nilai akhir diinisialisasi dengan 0 dan indeks dengan “N/A”. Pada fungsi main, pengguna diminta untuk memasukkan nilai empat *CLO* melalui *input*. Data yang diperoleh diproses oleh fungsi gabungNilaiStd dan hasilnya disimpan dalam variabel mahasiswa. Program menampilkan pesan bahwa nilai berhasil disimpan, serta menampilkan kembali salah satu nilai *CLO* yang tersimpan di dalam *struct*.

1. Kesimpulan

Praktikum ini berhasil memperlihatkan penerapan konsep dasar pemrograman dalam C++ melalui penggunaan struct, fungsi, array, pointer, serta struktur kontrol. Dengan memanfaatkan struct sebagai representasi Abstract Data Type (ADT), data dapat dikelola secara lebih sistematis dan mudah diakses. Penerapan fungsi seperti gabungNilaiStd dan hitungNilaiAkhir membuktikan efektivitas pendekatan modular dalam memecah persoalan kompleks menjadi bagian yang lebih sederhana dan dapat digunakan kembali. Selain itu, penguasaan input-output, operasi aritmatika berbobot, hingga penggunaan looping untuk array satu dan dua dimensi menunjukkan pentingnya keterampilan teknis dalam membangun program yang terstruktur. Secara keseluruhan, praktikum ini mendukung pemahaman mahasiswa terhadap konsep fundamental pemrograman sekaligus melatih logika penyelesaian masalah secara bertahap

1. Referensi

[1] Robins, A., Rountree, J., & Rountree, N. (2003). Learning and Teaching Programming: A Review and Discussion. Computer Science Education, 13(2), 137–172.  
[2] Lister, R. (2011). Concrete and Other Neo-Piagetian Forms of Reasoning in the Novice Programmer. Proceedings of the Thirteenth Australasian Computing Education Conference, 9–18.  
[3] Bennedsen, J., & Caspersen, M. E. (2007). Failure Rates in Introductory Programming. ACM SIGCSE Bulletin, 39(2), 32–36.