**LAPORAN PRAKTIKUM**

**STRUKTUR DATA**

**MODUL I**

**PENGENALAN CODE BLOCKS**

****

# Disusun Oleh :

NAMA : Jati Christanov Dite

NIM : 103122400032

# Dosen

**Diah Septiani S.KOM M.Cs**

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK  
FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO  
2025**

1. Dasar Teori

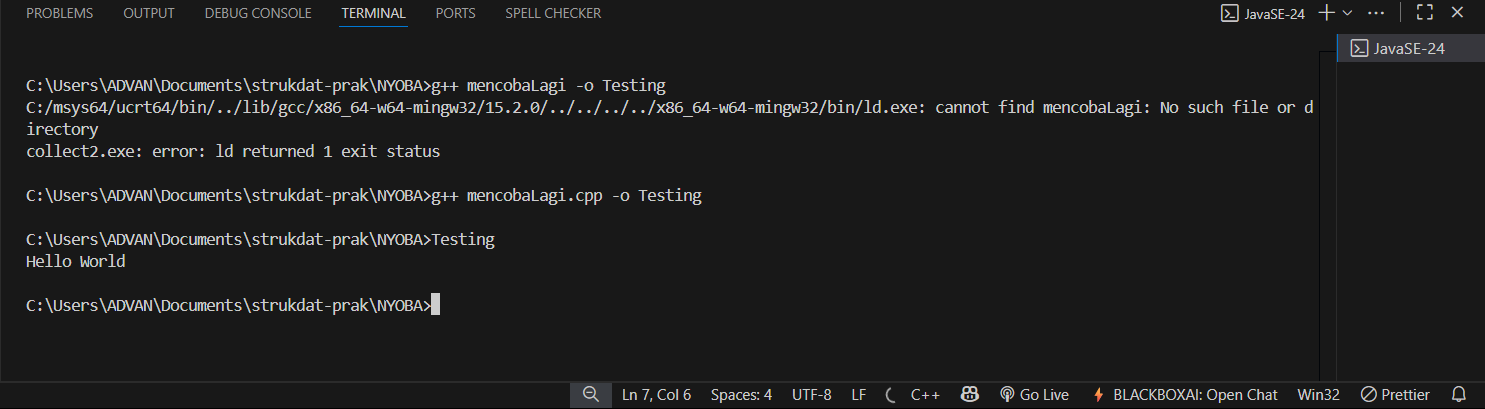
Bahasa pemrograman C++ dikembangkan untuk memfasilitasi pemrograman berorientasi objek dalam proyek simulasi. Kelebihan bahasa pemrograman C++ adalah dapat dijalankan dengan baik di berbagai mesin, dari komputer hingga server terpusat. Selain itu, bahasa pemrograman C++ dapat digunakan pada banyak framework kerja, misalnya DOS. UNIX, dan sebagainya [1].

1. Guided (berisi screenshot source code & output program disertai penjelasannya) [

Guided 1

|  |
| --- |
| #include <#include <iostream>      using namespace std;      int main(){          cout<<"Hello World"<<endl;          return 0;      } |

Screenshots Output



Deskripsi:

Program ini merupakan program dasar C++ yang digunakan untuk menampilkan teks “Hello World” ke layar. Program memanfaatkan library <iostream> untuk operasi input-output dan menggunakan perintah cout untuk mencetak teks. Struktur program ini memperkenalkan komponen utama dalam bahasa C++, yaitu fungsi main(), penggunaan namespace, perintah output, serta nilai balik return 0 yang menandakan eksekusi berhasil. Program ini menjadi latihan awal dalam pembelajaran pemrograman untuk melatih pemahaman dasar sintaks dan logika pemrograman [1].

Guided 2

|  |
| --- |
| #include<iostream>  #include<conio.h> //Untuk getch()  using namespace std;  int main(){      int x,y; //Variabel x dan y bertipe integer      int \*px; //Pointer px menunjuk ke variabel integer      x = 87; //Nilai x = 87      px = &x; //Pointer px menyimpan alamat dari x      y = \*px; // y menyimpan nilai yang ditunjuk oleh pointer px      cout<<"Alamat x = "<<&x<<endl; //Menampilkan alamat memori      cout<<"Isi px = "<<px<<endl; //isi pointer px = alamat x      cout<<"Isi x = "<<x<<endl; //Nilai variabel x      cout<<"Isi \*px = "<<\*px<<endl; //Nilai yang ditunjuk pointer      cout<<"Isi Y = "<<y<<endl; //Nilai y hasil copy dari \*px      getch(); //Agar program tidak langsung tertutup      return 0;  } |

Screenshots Output



Deskripsi:

Kode ini mendemonstrasikan konsep dasar pointer di C++. Variabel x diberi nilai 87, lalu sebuah pointer px digunakan untuk menyimpan alamat dari x. Dengan memanfaatkan dereferensi (\*px), nilai x bisa diakses kembali dan disalin ke variabel lain, yaitu y. Saat dijalankan, program akan menampilkan perbedaan antara alamat memori, isi pointer, nilai variabel, serta hasil dereferensinya. Contoh sederhana ini cocok untuk memperkenalkan hubungan antara variabel, alamat, dan pointer pada pemula.

Guided 3

|  |
| --- |
| #include<iostream>  #include<conio.h>  #define MAX 5  using namespace std;  int main(){      int i,j;      float nilai[MAX];      static int nilai\_tahun[MAX][MAX] = {          {0,2,2,0,0},          {0,1,1,1,0},          {0,3,3,3,0},          {4,4,0,0,4},          {5,0,0,0,5}      };      for(i = 0; i < MAX; i++){          cout<<"Masukan nilai ke-"<<i + 1<<": ";          cin>>nilai[i];      }      cout<<"\nData nilai siswa:\n";      for(i = 0; i < MAX;i++){          cout<<"Nilai ke-"<<i + i<<" = "<<nilai[i]<<endl;      }      cout<<"\nNilai tahunan:\n";      for(i = 0;i < MAX;i++){          for(j = 0;j < MAX;j++){              cout<<nilai\_tahun[i][j]<<" ";          }          cout<<endl;      }      getch();      return 0;  } |

Screenshots Output



Deskripsi:

Kode ini memperlihatkan bagaimana C++ menangani data dengan array satu dimensi dan dua dimensi. Pengguna diminta memasukkan 5 nilai yang disimpan di array nilai. Nilai-nilai tersebut kemudian ditampilkan ulang agar bisa dicek kembali. Selain itu, program juga memiliki array 2 dimensi bernama nilai\_tahun yang sudah diisi sebelumnya. Data ini dicetak dalam bentuk matriks menggunakan loop bersarang, sehingga tampak jelas susunan baris dan kolomnya. Dengan cara ini, program mengajarkan penggunaan array baik untuk input dinamis maupun data statis yang telah didefinisikan.

1. Unguided/Tugas (berisi screenshot source code & output program disertai penjelasannya)

Unguided 1

|  |
| --- |
| TYPE nilaiSTD = RECORD  CLO1 : REAL  CLO2 : REAL  CLO3 : REAL  CLO4 : REAL  nilaiAkhir : REAL  indeksNilai : STRING  END RECORD |

Deskripsi:

Tipe data nilaiSTD dibuat untuk menyimpan informasi nilai mahasiswa pada mata kuliah Struktur Data. Di dalamnya terdapat enam komponen, yaitu nilai CLO1 sampai CLO4, nilai akhir, serta indeks nilai. Nilai CLO dan nilai akhir menggunakan tipe desimal dengan kisaran 0 hingga 100. Sementara itu, indeks nilai berbentuk string yang berisi huruf mutu seperti A, AB, B, BC, C, D, atau E. Dengan adanya tipe bentukan ini, penyimpanan dan pengolahan data nilai mahasiswa menjadi lebih sistematis dan rapi.

Unguided 2

|  |
| --- |
| DECLARE mhs1, mhs2 : nilaiSTD |

Deskripsi:

Deklarasi di atas membuat dua variabel baru, yaitu mhs1 dan mhs2, yang keduanya menggunakan tipe bentukan nilaiSTD. Dengan deklarasi ini, setiap variabel dapat menyimpan enam komponen nilai, yaitu CLO1 hingga CLO4, nilai akhir, serta indeks nilai. Penggunaan tipe bentukan mempermudah pengelolaan data karena seluruh informasi terkait nilai mahasiswa bisa disimpan dalam satu variabel terstruktur.

Unguided 3

|  |
| --- |
| DECLARE CLO1, CLO2, CLO3, CLO4 : REAL  OUTPUT "Masukkan nilai CLO 1 : "  INPUT CLO1  OUTPUT "Masukkan nilai CLO 2 : "  INPUT CLO2  OUTPUT "Masukkan nilai CLO 3 : "  INPUT CLO3  OUTPUT "Masukkan nilai CLO 4 : "  INPUT CLO4 |

Deskripsi:

Pseudocode di atas membuat sebuah program sederhana untuk menerima input dari pengguna. Program akan meminta empat data berupa nilai CLO1 hingga CLO4. Keempat nilai tersebut dideklarasikan sebagai variabel bertipe real (desimal) dan bukan bagian dari tipe bentukan nilaiSTD. Dengan cara ini, pengguna dapat memasukkan nilai secara langsung untuk setiap CLO, yang kemudian bisa dipakai untuk proses perhitungan atau analisis lebih lanjut.

Unguided 4

|  |
| --- |
| Function ubahNilaiCLO (CLO1, CLO2, CLO3, CLO4 : REAL) → nilaiSTD |

Deskripsi:

Function ubahNilaiCLO digunakan untuk mengubah data input berupa nilai CLO1 sampai CLO4 menjadi sebuah data terstruktur dengan tipe bentukan nilaiSTD. Parameter yang dibutuhkan adalah empat nilai desimal (REAL) yang mewakili skor masing-masing CLO.

Unguided 5

|  |
| --- |
| FUNCTION simpanNilai (CLO1, CLO2, CLO3, CLO4 : real) → nilaiSTD  VAR data : nilaiSTD  BEGIN  data.CLO1 ← CLO1  data.CLO2 ← CLO2  data.CLO3 ← CLO3  data.CLO4 ← CLO4  data.nilaiAkhir ← 0  data.indeksNilai ← ""  RETURN data  END |

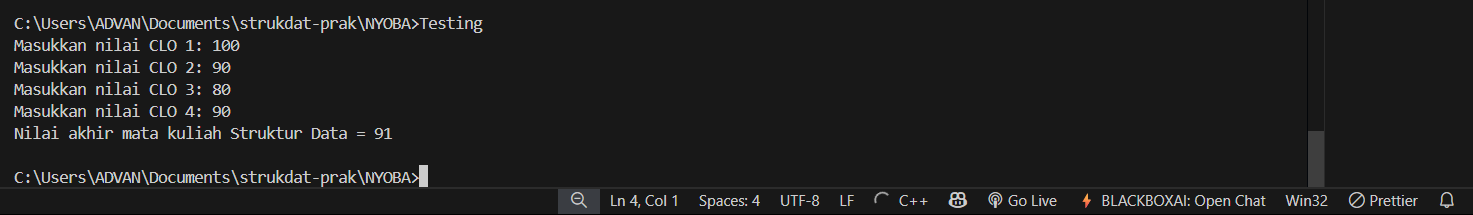
Deskripsi:

Function simpanNilai berfungsi untuk menyimpan data nilai CLO1 sampai CLO4 ke dalam sebuah variabel bertipe nilaiSTD. Empat nilai yang diberikan sebagai input akan langsung dimasukkan ke dalam struktur data sesuai dengan field masing-masing. Pada tahap awal, nilai akhir belum dihitung sehingga diisi dengan 0, sedangkan indeks nilai juga belum ditentukan sehingga masih kosong. Setelah proses penyimpanan selesai, function ini akan mengembalikan variabel data yang berisi informasi dasar nilai CLO. Dengan begitu, data sudah tersimpan secara terstruktur dan siap digunakan untuk perhitungan atau pengolahan lebih lanjut.

Unguided 6

|  |
| --- |
| #include <iostream>  using namespace std;  float hitungNilaiAkhir(float CLO1, float CLO2, float CLO3, float CLO4) {      float nilaiAkhir;      nilaiAkhir = (0.3 \* CLO1) + (0.3 \* CLO2) + (0.2 \* CLO3) + (0.2 \* CLO4);      return nilaiAkhir;  }  int main() {      float c1, c2, c3, c4, hasil;      cout << "Masukkan nilai CLO 1: ";      cin >> c1;      cout << "Masukkan nilai CLO 2: ";      cin >> c2;      cout << "Masukkan nilai CLO 3: ";      cin >> c3;      cout << "Masukkan nilai CLO 4: ";      cin >> c4;      hasil = hitungNilaiAkhir(c1, c2, c3, c4);      cout << "Nilai akhir mata kuliah Struktur Data = " << hasil << endl;      return 0;  } |

Screenshots Output



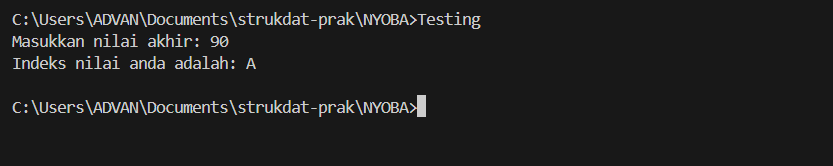
Deskripsi:

Program di atas menggunakan sebuah function bernama hitungNilaiAkhir yang menerima empat parameter, yaitu nilai CLO1 hingga CLO4. Di dalam function, masing-masing nilai dikalikan dengan bobotnya: 30% untuk CLO1, 30% untuk CLO2, 20% untuk CLO3, dan 20% untuk CLO4. Hasil perkalian tersebut dijumlahkan untuk menghasilkan nilai akhir, yang kemudian dikembalikan oleh function.

Unguided 7

|  |
| --- |
| #include <iostream>  #include <string>  using namespace std;  // Function untuk menentukan indeks nilai  string tentukanIndeks(float nilaiAkhir) {      if (nilaiAkhir > 80) {          return "A";      } else if (nilaiAkhir > 70 && nilaiAkhir <= 80) {          return "AB";      } else if (nilaiAkhir > 65 && nilaiAkhir <= 70) {          return "B";      } else if (nilaiAkhir > 60 && nilaiAkhir <= 65) {          return "BC";      } else if (nilaiAkhir > 50 && nilaiAkhir <= 60) {          return "C";      } else if (nilaiAkhir > 40 && nilaiAkhir <= 50) {          return "D";      } else {          return "E";      }  }  int main() {      float nilai;      cout << "Masukkan nilai akhir: ";      cin >> nilai;      string indeks = tentukanIndeks(nilai);      cout << "Indeks nilai anda adalah: " << indeks << endl;      return 0;  } |

Screenshots Output



Deskripsi:

Program ini memiliki sebuah function bernama tentukanIndeks yang digunakan untuk menentukan indeks mutu berdasarkan nilai akhir seorang mahasiswa. Function menerima satu parameter berupa angka desimal (nilai akhir), lalu menggunakan percabangan if-else untuk mengecek interval nilai yang sesuai.

1. Kesimpulan

Pada praktikum minggu pertama ini, kita mempelajari dasar bahasa pemrograman C++. Dengan membuat sebuah program sederhana seperti Hello World, kemudian mempelajari apa itu pointer, mempelajari Array dalam bahasa pemrograman C++

1. Referensi[Bob

[1] Aulia, F. & Yahfizham. (2024). Mengenal Bahasa Pemrograman Pada Algoritma Pemrograman. Journal of Informatics and Business, 1(4), 223–228. ITTC Indonesia.