**LAPORAN PRAKTIKUM**

**ALGORITMA & STRUKTUR DATA**

**MODUL 1**

****

**STRUCT & POINTER**

**Oleh:**

**Indra Suryadilaga NIM. 2410817310014**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS LAMBUNG MANGKURAT**

**APRIL 2025**

# LEMBAR PENGESAHAN

**LAPORAN PRAKTIKUM ALGORITMA & STRUKTUR DATA**

**MODUL 1**

Laporan Praktikum Algoritma & Struktur Data Modul 1: Struct & Pointer ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Web I. Laporan Prakitkum ini dikerjakan oleh:

Nama Praktikan : Indra Suryadilaga

NIM : 2410817310014

|  |  |
| --- | --- |
| Menyetujui,  Asisten Praktikum  Zulfa Auliya Akbar  NIM. 2210817210010 | Mengetahui,  Dosen Penanggung Jawab Praktikum  Andreyan Rizky Baskara, S.Kom., M.Kom  NIP. 199307032019031011 |

# DAFTAR ISI

[LEMBAR PENGESAHAN 2](#_Toc194921710)

[DAFTAR ISI 3](#_Toc194921711)

[DAFTAR GAMBAR 4](#_Toc194921712)

[DAFTAR TABEL 5](#_Toc194921713)

[SOAL 1 6](#_Toc194921714)

[Pembahasan 7](#_Toc194921715)

[Algoritma 9](#_Toc194921716)

[SOAL 2 11](#_Toc194921717)

[Source code 11](#_Toc194921718)

[Pembahasan 12](#_Toc194921719)

[Algoritma 12](#_Toc194921720)

[SOAL 3 14](#_Toc194921721)

[Source code 14](#_Toc194921722)

[Pembahasan 14](#_Toc194921723)

[Algoritma 15](#_Toc194921724)

# DAFTAR GAMBAR

**No table of contents entries found.**

# DAFTAR TABEL

[Table 1 Pembahasan Source Code 1.1 7](#_Toc194921964)

[Table 2 Pembahasan Source Code 1.2 7](#_Toc194921965)

[Table 3 Pembahasan Source Code 1.3 7](#_Toc194921966)

[Table 4 Pembahasan Source Code 1.4 8](#_Toc194921967)

[Table 5 Pembahasan Source Code 1.5 8](#_Toc194921968)

[Table 6 Pembahasan Source Code 1.6 9](#_Toc194921969)

[Table 7 Source Code Soal 2 11](#_Toc194921970)

[Table 8 Pembahasan Source Code 2.1 12](#_Toc194921971)

[Table 9 Pembahasan Source Code 2.2 12](#_Toc194921972)

[Table 10 Pembahasan Source Code 2.3 12](#_Toc194921973)

[Table 11 Source Code Soal 3 14](#_Toc194921974)

[Table 12 Pembahasan Source Code 3.1 15](#_Toc194921975)

[Table 13 Pembahasan Source Code 3.2 15](#_Toc194921976)

[Table 14 Pembahasan Source Code 3.3 15](#_Toc194921977)

# 

# SOAL 1

Cobalah program berikut, running dan analisis hasilnya. Buatlah algoritma untuk program tersebut.

A picture containing text

Description automatically generated

Table

Description automatically generated

## Pembahasan

Program pada gambar tersebut digunakan untuk menghitung biaya SPP berdasarkan program studi D3 atau S1 dan jumlah SKS yang diambil.

|  |
| --- |
| struct mhs {  char nama[20], nim[10], jurusan[2];  int sks, program;  };  struct mhs bayar[2]; |

Table 1 Pembahasan Source Code 1.1

Struct pertama digunakan untuk menyimpan data-data dari mahasiswa seperti nama, nim, jurusan, sks, serta program. Sedangkan struct yang kedua unutk menyimpan array berisi 2 objek mhs. Artinya terdapat 2 mahasiswa dengan struktur data yang sama.

* nama[20]:Menyimpan nama mahasiswa (maksimal 20 karakter).
* nim[10]: Menyimpan NIM mahasiswa (maksimal 10 karakter).
* jurusan[2]: Menyimpan jurusan (maksimal 2 karakter).
* sks: Menyimpan jumlah SKS yang diambil.
* program: Menyimpan jenis program (1 untuk D3, 2 untuk S1).

|  |
| --- |
| for(int i=0; i<2; i++)  {  cout<<"\nNama mhs = "; cin>>bayar[i].nama;  cout<<"\nNIM = "; cin>>bayar[i].nim;  cout<<"\nJurusan [TI, PTK] = "; cin>>bayar[i].jurusan;  input:  cout<<"\nProgram [1=D3, 2=S1] = "; cin>>bayar[i].program; |

Table 2 Pembahasan Source Code 1.2

Loop ini berjalan 2 iterasi yang digunakan untuk menginput data 2 mahasiswa.

|  |
| --- |
| if (bayar[i].program != 1 && bayar[i].program != 2) {  cout << "Program tidak sesuai\n";  goto input; // Lompat kembali ke label 'input'  }  } |

Table 3 Pembahasan Source Code 1.3

Pengkondisian yang digunakan untuk mengecek jika input dari Program bukan 1 atau 2, maka program tidak akan melanjutkan ke langkah berikutnya dan justru mengulang input Program.

|  |
| --- |
| cout<<"\nJumlah sks = "; cin>>bayar[i].sks;  if (bayar[i].program==1)  {  tetap=500000;  var=bayar[i].sks\*25000;  }  else if (bayar[i].program==2)  {  tetap=750000;  var=bayar[i].sks\*50000;  } |

Table 4 Pembahasan Source Code 1.4

Pengkondisian untuk menghitung jumlah SPP dan biaya tetap sesuai dengan Program dan jumlah sks.

* Jika program D3 (1): Biaya tetap 500.000, biaya per SKS 25.000.
* Jika program S1 (2): Biaya tetap 750.000, biaya per SKS 50.000.
* Total SPP = Biaya tetap + (Jumlah SKS \* biaya per SKS).

|  |
| --- |
| cout<<"\nOutput ";  cout<<"\nNama mhs = "<<bayar[i].nama;  cout<<"\nNIM = "<<bayar[i].nim;  cout<<"\nJurusan = "<<bayar[i].jurusan;  cout<<"\nProgram = "<<bayar[i].program;  cout<<"\nJumlah sks = "<<bayar[i].sks;  cout<<"\nSPP tetap = "<<tetap;  cout<<"\nSPP variable = "<<var;  cout<<endl; |

Table 5 Pembahasan Source Code 1.5

Output untuk menampilkan nama, nim, program, jumlah sks, biaya teteap, dan biaya variable.

Adapun kesalahan yang diperbaika adalah:

* char jurusan[2]: jika pengguna memasukan “TI” atau “PTK”, harusnya alokasi karekter lebih dari 2. Solusinya mengubah batasnya menjadi 3 karakter.
* goto input: membuat source code sulit dibaca dan dimaintain. Solusinya bisa menggunkan while loop.

|  |
| --- |
| do {  cout<<"\nProgram [1=D3, 2=S1] = ";  cin>>bayar[i].program;  if (bayar[i].program != 1 && bayar[i].program != 2)  cout << "Program tidak sesuai, silakan ulangi.\n";  } while (bayar[i].program != 1 && bayar[i].program != 2); |

Table 6 Pembahasan Source Code 1.6

## Algoritma

1. Mulai Program

2. Deklarasi Variabel

1. Buat struktur data mhs untuk menyimpan informasi mahasiswa:
   1. nama (string)
   2. nim (string)
   3. jurusan (string)
   4. sks (integer)
   5. program (integer)
2. Buat array bayar[2] untuk menyimpan data 2 mahasiswa.
3. Deklarasikan variabel bts, var, tetap untuk perhitungan SPP.

3. Input Data Mahasiswa (Loop untuk 2 mahasiswa)

Untuk setiap mahasiswa i (0 hingga 1):

1. Minta input nama mahasiswa.
2. Minta input NIM mahasiswa.
3. Minta input jurusan mahasiswa (TI atau PTK).

4. Validasi Program (D3 atau S1)

1. Minta input program studi (1 = D3, 2 = S1).
2. Jika input bukan 1 atau 2, tampilkan pesan "Program tidak sesuai".
3. Kembalikan ke langkah input program.

5. Input Jumlah SKS

1. Minta input jumlah SKS yang diambil mahasiswa.

6. Hitung SPP Mahasiswa

1. Jika program = 1 (D3)
   1. Biaya tetap = 500.000
   2. Biaya SKS = 25.000 per SKS.
2. Jika program = 2 (S1)
   1. Biaya tetap = 750.000
   2. Biaya SKS = 50.000 per SKS.

* Hitung SPP total = biaya tetap + (jumlah SKS \* biaya per SKS).

7. Output Data Mahasiswa

* Tampilkan Nama, NIM, Jurusan, Program, Jumlah SKS, Biaya Tetap, dan Total SPP.

8. Selesai

# SOAL 2

Buatlah program dengan menggunakan struct dengan hasil eksekusi program sebagai berikut:

1. Plat Nomor Kendaraan : DA1234MK
2. Jenis Kendaraan : RUSH
3. Nama Pemilik : Andika Hartanto
4. Alamat : Jl. Kayu Tangi 1
5. Kota : Banjarmasin.

## Source code

|  |
| --- |
| #include <iostream>  using *namespace* std;  *// Deklarasi struct Kendaraan*  *struct* Kendaraan {      string platNomor;      string jenisKendaraan;      string namaPemilik;      string alamat;      string kota;  };  *int* main() {  *// Inisialisasi data kendaraan menggunakan struct*      Kendaraan mobil = {"DA1234MK", "RUSH", "Andika Hartanto", "Jl. Kayu Tangi 1", "Banjarmasin"};    *// Menampilkan hasil eksekusi*      cout << "Plat Nomor Kendaraan : " << mobil.platNomor << endl;      cout << "Jenis Kendaraan : " << mobil.jenisKendaraan << endl;      cout << "Nama Pemili : " << mobil.namaPemilik << endl;      cout << "Alamat : " << mobil.alamat << endl;      cout << "Kota : " << mobil.kota << endl;        return 0;  } |

Table 7 Source Code Soal 2

## Pembahasan

Program ini menggunakan struct untuk mengelompokkan beberapa atribut yang berkaitan dalam satu entitas yaitu kendaraan.

|  |
| --- |
| struct Kendaraan {  string platNomor;  string jenisKendaraan;  string namaPemilik;  string alamat;  string kota;  }; |

Table 8 Pembahasan Source Code 2.1

Struct ini digunkan unutk menyimpan informasi kendaraan seperti plat nomor, jenis kendaraan, nama pemilik, alamat, dan kota. Yang mana semua tipe data variabelnya adalah string.

|  |
| --- |
| Kendaraan mobil = {"DA1234MK", "RUSH", "Andika Hartanto", "Jl. Kayu Tangi 1", "Banjarmasin"}; |

Table 9 Pembahasan Source Code 2.2

Variabel mobil dideklarasikan dengan tipe data kendaraan dengan menginisialisasikan dengan data kendaraan yang ditampilkan.

|  |
| --- |
| cout << "Plat Nomor Kendaraan : " << mobil.platNomor << endl;  cout << "Jenis Kendaraan : " << mobil.jenisKendaraan << endl;  cout << "Nama Pemilik : " << mobil.namaPemilik << endl;  cout << "Alamat : " << mobil.alamat << endl;  cout << "Kota : " << mobil.kota << endl; |

Table 10 Pembahasan Source Code 2.3

Output untuk menampilkan data dari mobil yang sudah diinisialisasikan.

## Algoritma

Algoritma Program Struct Kendaraan:

1. Mulai.
2. Deklarasi struct Kendaraan dengan atribut:
   1. platNomor (string).
   2. jenisKendaraan (string).
   3. namaPemilik (string).
   4. alamat (string).
   5. kota (string).
3. Inisialisasi variabel mobil dari struct Kendaraan dengan data:
   1. Plat Nomor Kendaraan = "DA1234MK".
   2. Jenis Kendaraan = "RUSH".
   3. Nama Pemilik = "Andika Hartanto".
   4. Alamat = "Jl. Kayu Tangi 1".
   5. Kota = "Banjarmasin".
4. Cetak hasil ke layar:
   1. "Plat Nomor Kendaraan :" diikuti dengan mobil.platNomor.
   2. "Jenis Kendaraan :" diikuti dengan mobil.jenisKendaraan.
   3. "Nama Pemilik :" diikuti dengan mobil.namaPemilik.
   4. "Alamat :" diikuti dengan mobil.alamat.
   5. "Kota :" diikuti dengan mobil.kota.
5. Selesai.

# SOAL 3

Buatlah program dengan tampilan sebagai berikut:

1. Masukkan sebuah huruf = .............
2. Masukan sebuah kata = .................
3. Masukkan Angka = ..................
4. Huruf yang Anda masukkan adalah .............
5. Kata yang Anda masukkan adalah ..............
6. Angka yang Anda masukkan adalah ........

## Source code

|  |
| --- |
| #include <iostream>  using namespace std;  int main() {  char huruf;  string kata;  int angka;  // Input data  cout << "Masukkan sebuah huruf = ";  cin >> huruf;  cin.ignore(); // Mengatasi masalah input buffer  cout << "Masukkan sebuah kata = ";  getline(cin, kata);  cout << "Masukkan sebuah angka = ";  cin >> angka;  // Output hasil  cout << "\nHuruf yang Anda masukkan adalah " << huruf << endl;  cout << "Kata yang Anda masukkan adalah " << kata << endl;  cout << "Angka yang Anda masukkan adalah " << angka << endl;  return 0;  } |

Table 11 Source Code Soal 3

## Pembahasan

|  |
| --- |
| char huruf;  string kata;  int angka; |

Table 12 Pembahasan Source Code 3.1

* huruf : Variabel bertipe char, digunakan untuk menyimpan satu karakter.
* Kata: Variabel bertipe string, digunakan untuk menyimpan satu kata atau kalimat.
* Angka: Variabel bertipe int, digunakan untuk menyimpan bilangan bulat yang dimasukkan pengguna.

|  |
| --- |
| cout << "Masukkan sebuah huruf = ";  cin >> huruf;  cin.ignore(); // Mengatasi masalah input buffer  cout << "Masukkan sebuah kata = ";  getline(cin, kata);  cout << "Masukkan sebuah angka = ";  cin >> angka; |

Table 13 Pembahasan Source Code 3.2

Input untuk pengguna memasukkan sebuah huruf, sebuah kata, dan sebuah angka. cin.ignore() digunakan untuk membersihkan buffer input agar getline(cin, kata) bisa berfungsi dengan baik setelah cin >> huruf. Sedangkan getline(cin, kata) digunakan untuk input bisa berupa teks yang mengandung spasi.

|  |
| --- |
| cout << "\nHuruf yang Anda masukkan adalah " << huruf << endl;  cout << "Kata yang Anda masukkan adalah " << kata << endl;  cout << "Angka yang Anda masukkan adalah " << angka << endl; |

Table 14 Pembahasan Source Code 3.3

Output untuk menampilkan kembali huruf, kata, dan angka yang telah dimasukan.

## Algoritma

1. Mulai.
2. Deklarasi struct DataInput dengan atribut:
   1. huruf (char).
   2. kata (string).
   3. angka (int).
3. Inisialisasi variabel data dari struct DataInput.
4. Membaca input dari pengguna:
5. Tampilkan "Masukkan sebuah huruf = ".
6. Simpan input ke dalam data.huruf.
7. Kosongkan buffer dengan cin.ignore().
8. Tampilkan "Masukkan sebuah kata = ".
9. Simpan input ke dalam data.kata menggunakan getline().
10. Tampilkan "Masukkan sebuah angka = ".
11. Simpan input ke dalam data.angka.
12. Cetak hasil ke layar:
13. Tampilkan "Huruf yang Anda masukkan adalah " diikuti dengan data.huruf.
14. Tampilkan "Kata yang Anda masukkan adalah " diikuti dengan data.kata.
15. Tampilkan "Angka yang Anda masukkan adalah " diikuti dengan data.angka.
16. Selesai.