

Laporan Tugas Pemrograman Dasar - Implementasi Union dan Struct

A. Exercise 6 (Valid or Invalid)

```

struct nameType
{
    string first;
    string last;
} ;

struct courseType
{
    string name;
    int callNum;
    int credits;
    char grade;
} ;

struct studentType
{
    nameType name;
    double gpa;
    courseType course;
} ;

```

studentType student;
 studentType classList[100];
 courseType course;
 nameType name;

- a. student.course.callNum = "CSC230";

Solusi : **Invalid**

student.course.callNum didefinisikan sebagai int, tetapi diberi input string ("CSC230").

- b. cin >> student.name;

Solusi : **Invalid**

cin tidak bisa membaca seluruh struct secara langsung. Oleh sebab itu, untuk menginput nama harus dilakukan dengan membaca setiap anggota struct satu per satu,

cin >> student.name.first; // Contoh

cin >> student.name.last; // Contoh

- c. classList[0] = name;

Solusi : **Invalid**

Tipe data tidak cocok. classList[0] adalah variabel dengan tipe studentType, sedangkan name adalah variabel dengan tipe nameType. Nilai nameType tidak bisa diberikan ke studentType.

- d. classList[1].gpa = 3.45;

Solusi : **Valid**

classList[1] adalah studentType yang valid dan gpa adalah anggota bertipe double. Memberinya nilai 3.45 (sebuah double) adalah operasi yang valid.

- e. name = classList[15].name;

Solusi : **Valid**

Ini adalah penetapan nilai antar struct yang tipenya sama. name adalah tipe nameType, dan classList[15].name juga tipe nameType.

- f. student.name = name;

Solusi : **Valid**

Ini adalah penetapan nilai yang valid antara dua variabel yang sama-sama bertipe nameType.

g. cout << classList[10] << endl;

Solusi : **Invalid**

Cout tidak bisa mencetak seluruh struct secara langsung. Setiap anggota struct harus dipanggil dan dicetak satu per satu.

h. for (int j = 0; j < 100; j++)

 classList[j].name = name;

Solusi : **Valid**

Loop ini mengiterasi array classList. Di dalam loop, setiap classList[j].name (yang bertipe nameType) diberi nilai dari name (juga nameType). Ini adalah operasi yang valid.

i. classList.course.credits = 3;

Solusi : **Invalid**

classList adalah sebuah array, bukan struct. Anda tidak bisa menggunakan operator titik (.) langsung pada nama array. Anda harus menentukan elemen mana yang ingin Anda akses menggunakan indeks.

 classList[0].course.credits = 3; // Contoh

j. course = studentType.course;

Solusi : **Invalid**

studentType adalah sebuah nama tipe, bukan variabel. Operator titik (.) tidak dapat digunakan pada nama tipe, operator tersebut harus harus digunakan pada variabel,

 course = student.course; // Contoh

B. Exercise 2 (Students Grade)

a. Permasalahan

Buatlah program C++ yang membaca data 20 siswa (nama depan, nama belakang, skor tes) dari sebuah file. Program harus menyimpan data ini dalam sebuah array dari struct studentType.

Tugas program adalah sebagai berikut.

1. Memberikan nilai huruf (A, B, C, D, F) berdasarkan skor tes.
2. Mencari skor tes tertinggi di kelas.
3. Membuat file laporan studentResults.txt yang berisi:
 - o Daftar nama siswa (format: NamaBelakang, NamaDepan, skor, dan nilai huruf).
 - o Laporan skor tertinggi dan nama-nama siswa yang mendapatkannya.
4. Fungsi main hanya boleh berisi deklarasi, pembukaan file, dan panggilan fungsi.

b. Penjelasan Kode

Program ini dibangun secara modular mengikuti aturan yang diberikan:

1. Struktur Data (studentType)

Sebuah struct dibuat untuk menyimpan data studentFName, studentLName, testScore, dan grade sebagai satu unit. Sebuah array global studentType classList[20] dibuat untuk menampung data seluruh kelas.

2. Fungsi main

Fungsi ini hanya mendeklarasikan array classList, membuka file input (studentsScore.txt) dan output (studentResults.txt), lalu memanggil fungsi-fungsi pekerja secara berurutan.

3. Fungsi Pekerja:

- o readData: Membaca studentsScore.txt baris per baris dan mengisi array classList.
- o assignGrades: Mengisi komponen grade untuk setiap siswa di array berdasarkan testScore.
- o printResults: Mencetak laporan utama (daftar nama, skor, nilai) ke studentResults.txt dengan format yang diminta.
- o findHighestScore: Memindai array untuk menemukan dan mengembalikan skor tes tertinggi.
- o printHighestScorers: Mencetak nama siswa yang memiliki skor sama dengan skor tertinggi ke studentResults.txt.

Source Code : [GitHub Muhammad Wisnu](#)

c. Hasil Eksekusi

```
PS C:\Users\muhwi\Documents\GitHub\Pemrograman_Dasar_Ganjil_2025\Exercises> g++ student_grade.cpp -o student_grade.exe
PS C:\Users\muhwi\Documents\GitHub\Pemrograman_Dasar_Ganjil_2025\Exercises> ./student_grade.exe
Pemrosesan selesai. Silakan periksa file 'results.txt'.
PS C:\Users\muhwi\Documents\GitHub\Pemrograman_Dasar_Ganjil_2025\Exercises> []
```

studentScore.txt :

Budi Santoso 85
Ani Wijaya 92
Eko Hartono 78
Siti Rahayu 99
Agus Hidayat 65
Dewi Lestari 88
Joko Nugroho 71
Rina Handayani 95
Hendra Gunawan 81
Maya Puspita 76
Doni Saputra 95
Fitri Wulandari 68
Asep Prasetyo 82
Indah Astuti 89
Yusuf Mulyadi 70

Lestari Novitasari 99
Rizky Ramadhan 60
Wulan Utami 83
Dedi Susanto 55
Putri Setiawan 91

studentResults.txt

Nama Siswa	Skor	Nilai
Santoso, Budi	85	B
Wijaya, Ani	92	A
Hartono, Eko	78	C
Rahayu, Siti	99	A
Hidayat, Agus	65	D
Lestari, Dewi	88	B
Nugroho, Joko	71	C
Handayani, Rina	95	A
Gunawan, Hendra	81	B
Puspita, Maya	76	C
Saputra, Doni	95	A
Wulandari, Fitri	68	D
Prasetyo, Asep	82	B
Astuti, Indah	89	B
Mulyadi, Yusuf	70	C
Novitasari, Lestari	99	A
Ramadhan, Rizky	60	D
Utami, Wulan	83	B
Susanto, Dedi	55	F
Setiawan, Putri	91	A

Skor Tes Tertinggi: 99
Siswa dengan skor tertinggi:
Rahayu, Siti
Novitasari, Lestari