

TUGAS PEMROGRAMAN DASAR (Struct and Union)

Oleh: Zulfa Arsyandhi Azra

NIM: 25/565145/TK/63744



PROGRAM STUDI TEKNOLOGI INFORMASI

DEPARTEMEN TEKNIK ELEKTRO DAN TEKNOLOGI INFORMASI

PROGRAM SARJANA - UNIVERSITAS GADJAH MADA

YOGYAKARTA

2025

1. Consider the following statements:

```
struct nameType struct courseType struct studentType
{ { {
    string first; string name; nameType name;
    string last; int callNum; double gpa;
}; int credits; courseType course;
char grade; };
};

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

Mark the following statements as valid or invalid. If a statement is invalid, explain why.

- a. student.course.callNum = "CSC230";
- b. cin >> student.name;
- c. classList[0] = name;
- d. classList[1].gpa = 3.45;
- e. name = classList[15].name;
- f. student.name = name;
- g. cout << classList[10] << endl;
- h. for (int j = 0; j < 100; j++)
classList[j].name = name;
- i. classList.course.credits = 3;
- j. course = studentType.course;

Answer:

- a. Invalid: callNum bertipe int, tetapi diberi string literal "CSC230", jadi tipe tidak cocok.
- b. Invalid: student.name bertipe struct nameType. cin tidak bisa langsung input struct tanpa overload >>. Harus input field-nya satu-satu
- c. Invalid: classList[0] bertipe studentType, sedangkan name bertipe nameType. Tipenya berbeda, jadi tidak bisa assign.
- d. Valid: gpa bertipe double, dan 3.45 adalah nilai double.
- e. Valid: Kedua sisi bertipe nameType. Assign struct ke struct yang sama.
- f. Valid: Kedua variabel bertipe nameType. Assign struct yang sama.

- g. Invalid: classList[10] adalah studentType. Tidak ada operator<< untuk struct ini, sehingga cout tidak tahu cara mencetaknya.
- h. Valid: Mengakses tiap elemen array classList dan men-assign field name yang sama jenisnya (nameType). Loop juga benar untuk array ukuran 100.
- i. Invalid: lassList adalah array, bukan satu struct studentType. Tidak bisa langsung akses classList.course. Harus pilih index elemen
- j. Invalid: studentType adalah nama tipe (struct), bukan variabel. Tidak bisa akses field dari nama tipe.

2. Write a program that reads students' names followed by their test scores. The program should output each student's name followed by the test scores and the relevant grade. It should also find and print the highest test score and the name of the students having the highest test score.

Student data should be stored in a **struct** variable of type **studentType**, which has four components: **studentFName** and **studentLName** of type **string**, **testScore** of type **int** (testScore is between 0 and 100), and **grade** of type **char**. Suppose that the class has 20 students. Use an array of 20 components of type **studentType**.

Your program must contain at least the following functions:

- a. A function to read the students' data into the array.
- b. A function to assign the relevant grade to each student.
- c. A function to find the highest test score.
- d. A function to print the names of the students having the highest test score.

Your program must output each student's name in this form: last name followed by a comma, followed by a space, followed by the first name; the name must be left justified. Moreover, other than declaring the variables and opening the input and output files, the function **main** should only be a collection of function calls.

Answer:

- A. Program menyiapkan tempat untuk data siswa

Program membuat wadah/array berisi 20 siswa.
Setiap siswa punya data:

- Nama depan
- Nama belakang
- Nilai tes (0–100)

- Grade (A–F)

Untuk menyiapkan tabel kosong untuk menulis data 20 siswa.

- B. Program meminta kamu memasukkan data siswa satu per satu

User memasukkan:

NamaDepan NamaBelakang Nilai

Contoh:

Ali Pratama 90

Pada tahap ini, program hanya menyimpan apa yang ditulis tanpa memproses nilai atau grade.

- C. Program memberikan grade untuk setiap siswa

Setelah semua data masuk, program mengecek nilai setiap siswa dan memberi grade:

- 90–100 → A
- 80–89 → B
- 70–79 → C
- 60–69 → D
- < 60 → F

- D. Program mencari nilai tertinggi

Program membaca semua nilai siswa satu per satu, lalu:

- Menyimpan nilai yang paling tinggi
- Membandingkannya dengan nilai lain

- E. Program menampilkan semua data siswa

Ditampilkan dalam format:

NamaBelakang, NamaDepan Nilai Grade

Nama belakang di depan agar urut dan formal, seperti daftar ujian universitas.

- F. Program menampilkan siswa dengan nilai tertinggi

Jika ada lebih dari satu siswa yang nilai tertingginya sama, semuanya ditampilkan.

Karena bisa saja ada lebih dari satu siswa yang dapat nilai maksimal.

- G. Hasil akhir yang ditampilkan program

- Seluruh data siswa: nama, nilai, grade
- Nilai tertinggi
- Nama siswa yang memiliki nilai tertinggi

```

Users > zulfaarsyandhiazra > Documents > ZULFA > KULIAH > MATERI > CODING > belajar.cpp > tampilSemua(studentType[])
1 #include <iostream>
2 #include <iomanip>
3 #include <string>
4 using namespace std;
5
6 const int JUMLAH_SISWA = 20;
7
8 struct studentType {
9     string namaDepan;
10    string namaBelakang;
11    int nilaiTes;
12    char grade;
13 };
14
15 void inputData(studentType siswa[]);
16 void beriGrade(studentType siswa[]);
17 int cariNilaiTertinggi(studentType siswa[]);
18 void tampilSiswaTertinggi(studentType siswa[], int nilaiTertinggi);
19 void tampilSemua(studentType siswa[]);
20
21 int main() {
22     studentType siswa[JUMLAH_SISWA];
23
24     inputData(siswa);
25     beriGrade(siswa);
26     int tertinggi = cariNilaiTertinggi(siswa);
27
28     tampilSemua(siswa);
29     cout << "Nilai Tertinggi: " << tertinggi << endl;
30     tampilSiswaTertinggi(siswa, tertinggi);
31
32     return 0;
33 }
34
35 void inputData(studentType siswa[]) {
36     cout << "Masukkan nama depan, nama belakang, dan nilai tes untuk 20 siswa:\n";
37     for (int i = 0; i < JUMLAH_SISWA; i++) {
38         cin >> siswa[i].namaDepan >> siswa[i].namaBelakang >> siswa[i].nilaiTes;
39     }
40 }
41 void beriGrade(studentType siswa[]) {
42     for (int i = 0; i < JUMLAH_SISWA; i++) {
43         if (siswa[i].nilaiTes >= 90) siswa[i].grade = 'A';
44         else if (siswa[i].nilaiTes >= 80) siswa[i].grade = 'B';
45         else if (siswa[i].nilaiTes >= 70) siswa[i].grade = 'C';
46         else if (siswa[i].nilaiTes >= 60) siswa[i].grade = 'D';
47         else siswa[i].grade = 'F';
48     }
49 }
50
51 int cariNilaiTertinggi(studentType siswa[]) {
52     int tertinggi = siswa[0].nilaiTes;
53     for (int i = 1; i < JUMLAH_SISWA; i++) {
54         if (siswa[i].nilaiTes > tertinggi) tertinggi = siswa[i].nilaiTes;
55     }
56     return tertinggi;
57 }
58 void tampilSiswaTertinggi(studentType siswa[], int nilaiTertinggi) {
59     cout << "Siswa dengan nilai tertinggi:\n";
60     for (int i = 0; i < JUMLAH_SISWA; i++) {
61         if (siswa[i].nilaiTes == nilaiTertinggi) tertinggi = siswa[i].nilaiTes;
62         cout << siswa[i].namaBelakang << " " << siswa[i].namaDepan << endl;
63     }
64 }
65 void tampilSemua(studentType siswa[]) {
66     cout << left << setw(20) << "Nama"
67     << setw(10) << "Nilai"
68     << "Grade" << endl;
69     cout << "-----\n";
70
71     for (int i = 0; i < JUMLAH_SISWA; i++) {
72         string nama = siswa[i].namaBelakang + ", " + siswa[i].namaDepan;
73         cout << left << setw(20) << nama
74             << setw(10) << siswa[i].nilaiTes
75             << siswa[i].grade << endl;
76     }
77 }

```

Tono Hartono 68cd "/Users/zulfaarsyandhiazra/Documents/ZULFA/KULIAH/MATERI/CODING" & g++ belajar.cpp -o belajar & "/Users/zulfaarsyandhiazra/Desktop/belajar"
 + ... | ☰ x

PROBLEMS 0/0 OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER 100

Code

Nama	Nilai	Grade
Pratama, Ali	90	A
Santoso, Budi	85	B
Dewi, Citra	78	C
Widodo, Deni	92	A
Putri, Erna	60	D
Saputra, Fajair	88	B
Melati, Gina	95	A
Widadi	70	C
Sari, Ika	82	B
Rahman, Joko	55	F
Setiawan, Kiki	77	C
Husnawati, Rutfi	83	B
Ayu, Mira	99	A
Larasati, Nini	65	D
Widayaty, Sri	58	F
Ramadhan, Putra	74	C
Alamsyah, Qori	91	A
Widatri, Rani	80	B
Ayudha, Sinta	87	B
Hartono, Tono	68	D

Nilai Tertinggi: 99

Siswa dengan nilai tertinggi:
Ayu, Mira

zulfaarsyandhiazra@Zulfa-MacBook-Air: ~/Desktop