

# Tugas Mandiri

Matakuliah Programming Language Pragmatics  
Pertemuan 4 (Selasa, 9 September 2025)

# Pertanyaan

Analisis sintaks dan semantik dari program terlampir !

Kemudian beri penjelasan terkait dua hal berikut !

1. Efek memori overlap memori pada union.
2. Perbandingan ukuran struct vs union dengan sizeof().

# Program 1

```
#include <stdio.h>

// definisi union
union Data {
    int i;
    float f;
    char c;
};

int main() {
    union Data data;

    data.i = 10;
    printf("data.i = %d\n", data.i);

    data.f = 3.14;
    printf("data.f = %.2f\n", data.f);

    data.c = 'A';
    printf("data.c = %c\n", data.c);

    return 0;
}
```

# Program 2

```
#include <stdio.h>

struct Hobi {
    unsigned music      : 1;
    unsigned sport      : 1;
    unsigned travel     : 1;
    unsigned reading    : 1;
    unsigned gaming     : 1;
    unsigned cooking    : 1;
    unsigned art         : 1;
    unsigned coding     : 1;
    unsigned movie      : 1;
    unsigned photography : 1;
};

int main() {
    struct Hobi hobi = {0}; // semua 0

    // Mahasiswa suka musik, gaming, dan coding
    hobi.music = 1;
    hobi.gaming = 1;
    hobi.coding = 1;

    printf("Music: %d\n", hobi.music);
    printf("Gaming: %d\n", hobi.gaming);
    printf("Coding: %d\n", hobi.coding);
    printf("Travel: %d\n", hobi.travel);

    return 0;
}
```

# Program 3

```
#include <stdio.h>

// definisi bit untuk setiap hobi
#define MUSIC      (1 << 0)
#define SPORT      (1 << 1)
#define TRAVEL     (1 << 2)
#define READING    (1 << 3)
#define GAMING     (1 << 4)
#define COOKING    (1 << 5)
#define ART        (1 << 6)
#define CODING     (1 << 7)
#define MOVIE      (1 << 8)
#define PHOTO      (1 << 9)

int main() {
    unsigned int hobi = 0;

    // Mahasiswa punya hobi Music, Gaming, dan Coding
    hobi |= MUSIC;
    hobi |= GAMING;
    hobi |= CODING;

    // cek
    if (hobi & MUSIC) printf("Suka musik\n");
    if (hobi & TRAVEL) printf("Suka traveling\n");
    if (hobi & CODING) printf("Suka coding\n");

    return 0;
}
```

# Program 4

```
#include <stdio.h>

enum DataType { INT, FLOAT, CHAR };

union Data {
    int i;
    float f;
    char c;
};

struct Variant {
    enum DataType type;
    union Data value;
};

int main() {
    struct Variant v;
    v.type = FLOAT;
    v.value.f = 3.14;

    if (v.type == FLOAT)
        printf("Value = %.2f\n", v.value.f);

    return 0;
}
```