Nama : Ilham Yusuf Wi'am
NIM : 24/539979/TK/59890
No. Kursi : 33

1. Kode :

```cpp
#include <iostream>
#include <array>
#include <cmath>

using namespace std;

const int JUMLAH_TITIK = 10;
typedef array<double, 3> Titik;

double hitungJarak(Titik A, Titik B) {
    double total = 0;
    for (int i = 0; i < 3; ++i)
        total += pow(A[i] - B[i], 2);
    return sqrt(total);
}

int main() {
    ios::sync_with_stdio(false);

    Titik acuan = {6, 7, 8};
    Titik daftarTitik[JUMLAH_TITIK];

    cout << "Masukkan koordinat untuk setiap titik:\n";
    for (int i = 0; i < JUMLAH_TITIK; ++i) {
        cout << "Titik ke-" << i << ": ";
        for (int j = 0; j < 3; ++j)
            cin >> daftarTitik[i][j];
    }

    double jarak[JUMLAH_TITIK];
    for (int i = 0; i < JUMLAH_TITIK; ++i)
        jarak[i] = hitungJarak(daftarTitik[i], acuan);

    for (int i = 0; i < JUMLAH_TITIK; ++i) {
        cout << "Jarak dari titik ke-" << i << " ke titik acuan ("
             << acuan[0] << ", " << acuan[1] << ", " << acuan[2]
             << "): " << jarak[i] << '\n';
    }

    return 0;
}
```

Output :

```
Praktikan@CDSR3-33 MSYS /c/iam
$ ./a.exe
Masukkan koordinat untuk setiap titik:
Titik ke-0: 1
2
3
Titik ke-1: 4
5
6
Titik ke-2: 7
8
9
Titik ke-3: 1
2
5
Titik ke-4: 2
3
6
Titik ke-5: 4
7
9
Titik ke-6: 1
0
3
Titik ke-7: 2
5
8
Titik ke-8: 3
2
5
Titik ke-9: 1
7
3
Jarak dari titik ke-0 ke titik acuan (6, 7, 8): 8.66025
Jarak dari titik ke-1 ke titik acuan (6, 7, 8): 3.4641
Jarak dari titik ke-2 ke titik acuan (6, 7, 8): 1.73205
Jarak dari titik ke-3 ke titik acuan (6, 7, 8): 7.68115
Jarak dari titik ke-4 ke titik acuan (6, 7, 8): 6
Jarak dari titik ke-5 ke titik acuan (6, 7, 8): 2.23607
Jarak dari titik ke-6 ke titik acuan (6, 7, 8): 9.94987
Jarak dari titik ke-7 ke titik acuan (6, 7, 8): 4.47214
Jarak dari titik ke-8 ke titik acuan (6, 7, 8): 6.55744
Jarak dari titik ke-9 ke titik acuan (6, 7, 8): 7.07107
```

2. Kode :

```cpp
#include <iostream>
#include <vector>
using namespace std;

// Fungsi untuk memecah dan mencetak digit-digit angka dalam urutan yang benar
void printDigits(int number) {
    vector<int> digits;

    int original = number;
    while (number > 0) {
        digits.push_back(number % 10);
        number /= 10;
    }

    cout << "Digit-digit dari angka " << original << " adalah: ";
    for (auto it = digits.rbegin(); it != digits.rend(); ++it)
        cout << *it << " ";
    cout << '\n';
}

// Fungsi untuk mengecek apakah angka habis dibagi 9
bool isDivisibleBy9(int number) {
    return number % 9 == 0;
}

int main() {
    const int digit[] = {154368, 421594, 123456};
    const int size = sizeof(digit) / sizeof(digit[0]);

    for (int i = 0; i < size; ++i) {
        int number = digit[i];

        printDigits(number);

        cout << "Angka " << number
             << (isDivisibleBy9(number) ? " habis" : " tidak habis")
             << " dibagi 9.\n\n";
    }

    return 0;
}
```

Output :

```
Praktikan@CDSR3-33 MSYS /c/iam
$ ./a.exe
Digit-digit dari angka 154368 adalah: 1 5 4 3 6 8
Angka 154368 habis dibagi 9.

Digit-digit dari angka 421594 adalah: 4 2 1 5 9 4
Angka 421594 tidak habis dibagi 9.

Digit-digit dari angka 123456 adalah: 1 2 3 4 5 6
Angka 123456 tidak habis dibagi 9.
```

3. Kode :

```cpp
#include <iostream>
#include <vector>
#include <iomanip>
using namespace std;

int inputJumlahTeman() {
    int jumlah;
    do {
        cout << "Masukkan jumlah teman (1 < jumlah teman <= 999999): ";
        cin >> jumlah;
    } while (jumlah <= 1 || jumlah > 999999);
    return jumlah;
}

double inputPersentaseDiskon() {
    double diskon;
    do {
        cout << "Masukkan persentase diskon (1 < n < 50): ";
        cin >> diskon;
    } while (diskon <= 1 || diskon >= 50);
    return diskon;
}

vector<double> inputTagihanTeman(int jumlahTeman) {
    vector<double> tagihan(jumlahTeman);
    for (int i = 0; i < jumlahTeman; ++i) {
        double nilai;
        do {
            cout << "Masukkan tagihan teman ke-" << (i + 1) << " (>= 1000): ";
            cin >> nilai;
        } while (nilai < 1000);
        tagihan[i] = nilai;
    }
    return tagihan;
}

void tampilkanTagihanSetelahDiskon(const vector<double>& tagihan, double diskon) {
    double total = 0;
    double totalDiskon = 0;

    cout << fixed << setprecision(2);

    for (int i = 0; i < tagihan.size(); ++i) {
        double potongan = tagihan[i] * (diskon / 100);
        double akhir = tagihan[i] - potongan;
        cout << "Tagihan teman ke-" << (i + 1) << " setelah diskon: " << akhir << endl;
        total += tagihan[i];
        totalDiskon += potongan;
    }

    cout << "Diskon total yang didapat: " << totalDiskon << endl;
    cout << "Total harga setelah diskon: " << total - totalDiskon << endl;
}

int main() {
    int jumlahTeman = inputJumlahTeman();
    double diskon = inputPersentaseDiskon();
    vector<double> tagihan = inputTagihanTeman(jumlahTeman);
    tampilkanTagihanSetelahDiskon(tagihan, diskon);

    return 0;
}
```

Output :

```
Praktikan@CDSR3-33 MSYS /c/iam
$ ./a.exe
Masukkan jumlah teman (1 < jumlah teman <= 999999): 10
Masukkan persentase diskon (1 < n < 50): 5
Masukkan tagihan teman ke-1 (>= 1000): 1000
Masukkan tagihan teman ke-2 (>= 1000): 20000
Masukkan tagihan teman ke-3 (>= 1000): 30000
Masukkan tagihan teman ke-4 (>= 1000): 40000
Masukkan tagihan teman ke-5 (>= 1000): 400000
Masukkan tagihan teman ke-6 (>= 1000): 50000
Masukkan tagihan teman ke-7 (>= 1000): 7000
Masukkan tagihan teman ke-8 (>= 1000): 900000
Masukkan tagihan teman ke-9 (>= 1000): 1000000
Masukkan tagihan teman ke-10 (>= 1000): 750000
Tagihan teman ke-1 setelah diskon: 950.00
Tagihan teman ke-2 setelah diskon: 19000.00
Tagihan teman ke-3 setelah diskon: 28500.00
Tagihan teman ke-4 setelah diskon: 38000.00
Tagihan teman ke-5 setelah diskon: 380000.00
Tagihan teman ke-6 setelah diskon: 47500.00
Tagihan teman ke-7 setelah diskon: 6650.00
Tagihan teman ke-8 setelah diskon: 855000.00
Tagihan teman ke-9 setelah diskon: 950000.00
Tagihan teman ke-10 setelah diskon: 712500.00
Diskon total yang didapat: 159900.00
Total harga setelah diskon: 3038100.00
```

4. Kode :

```cpp
#include <iostream>
using namespace std;

int main(){
    int a, b;
    int *pa = &a, *pb = &b;
    int sum, absDifference;

    cin >> a >> b;

    if (1 <= a <= b <= 99){
        sum = a + b;
        absDifference =  a - b;
        absDifference = -(absDifference);
    } else {
        cout << "Mohon masukkan nilai yang benar";
    }

    cout << sum << "\n" << absDifference;
    return 0;
}
```

Output :

```
Praktikan@CDSR3-33 MSYS /c/iam
$ ./a.exe
4
5
9
1
```

5. Kode :

```cpp
#include <iostream>
#include <algorithm>

using namespace std;

struct Book {
    string bookName;
    string authorName;
    int year;
    long cost;
}book[5];

void getData();
bool compareBooks( Book a, Book b);
void showData();

int main()
{
    cout << "============= MY Favorite Books =============\n";
    cout << "\n=> Enter your Five Favorite Books Detail:" <<endl;
    getData();

    sort(book, book+5, compareBooks);
    showData();

    return 0;
}

void getData() {
    for (int i = 0; i < 5; ++i) {
        cout << "\nBook #" << i << endl;

        cout << "Enter Name of the Book: ";
        cin >> book[i].bookName;

        cout << "Enter Author Name of the Book: ";
        cin >> book[i].authorName;

        cout << "Enter Published Year of the Book: ";
        cin >> book[i].year;

        cout << "Enter Cost of the Book: ";
        cin >> book[i].cost;
    }
}

void showData() {
    cout << "\n\n============= Favorite Books Record =============\n";
    cout << "\n\n";
    for (int i = 0; i < 5; i++) {
        cout << i+1 << ") Book Name: "<< book[i].bookName<< " ( " << book[i].year << ")" <<endl;
        cout << "Author Name: " <<book[i].authorName <<endl;
        cout << "Book cost: " << book[i].cost <<endl;
        cout << "\n";
    }
}

bool compareBooks(Book a, Book b){
    if (a.cost < b.cost){
        return true;
    } return false;
}
```

J. Output :

```
Praktikan@CDSR3-33 MSYS /c/iam
$ ./a.exe
============= MY Favorite Books =============

=> Enter your Five Favorite Books Detail:

Book #0
Enter Name of the Book: raja
Enter Author Name of the Book: tuna
Enter Published Year of the Book: 202
Enter Cost of the Book: 5000

Book #1
Enter Name of the Book: nera
Enter Author Name of the Book: rena
Enter Published Year of the Book: 45
Enter Cost of the Book: 4500

Book #2
Enter Name of the Book: roti
Enter Author Name of the Book: bakar
Enter Published Year of the Book: 2000
Enter Cost of the Book: 2020

Book #3
Enter Name of the Book: toni
Enter Author Name of the Book: break
Enter Published Year of the Book: 2045
Enter Cost of the Book: 3050

Book #4
Enter Name of the Book: rana
Enter Author Name of the Book: rona
Enter Published Year of the Book: 1980
Enter Cost of the Book: 3400


============= Favorite Books Record =============


1) Book Name: roti ( 2000)
Author Name: bakar
Book cost: 2020

2) Book Name: toni ( 2045)
Author Name: break
Book cost: 3050

3) Book Name: rana ( 1980)
Author Name: rona
Book cost: 3400

4) Book Name: nera ( 45)
Author Name: rena
Book cost: 4500

5) Book Name: raja ( 202)
Author Name: tuna
Book cost: 5000
```

Penjelasan :

a. File header : library include dan algorithm
b. Built-in function : struct Book
c. User-defined function : void getData(), void showData(), bool compareBooks
d. Function prototype :
e. Function call : getData() in int main()
f. Function implementation : sort(book, book+5, compareBooks)
g. Parameter formal : Book a, Book b
h. Parameter aktual : Book
i. Array structure : book[5]

j. Output :

```
Praktikan@CDSR3-33 MSYS /c/iam
$ ./a.exe
============= MY Favorite Books =============

=> Enter your Five Favorite Books Detail:

Book #0
Enter Name of the Book: raja
Enter Author Name of the Book: tuna
Enter Published Year of the Book: 202
Enter Cost of the Book: 5000

Book #1
Enter Name of the Book: nera
Enter Author Name of the Book: rena
Enter Published Year of the Book: 45
Enter Cost of the Book: 4500

Book #2
Enter Name of the Book: roti
Enter Author Name of the Book: bakar
Enter Published Year of the Book: 2000
Enter Cost of the Book: 2020

Book #3
Enter Name of the Book: toni
Enter Author Name of the Book: break
Enter Published Year of the Book: 2045
Enter Cost of the Book: 3050

Book #4
Enter Name of the Book: rana
Enter Author Name of the Book: rona
Enter Published Year of the Book: 1980
Enter Cost of the Book: 3400


============= Favorite Books Record =============


1) Book Name: roti ( 2000)
Author Name: bakar
Book cost: 2020

2) Book Name: toni ( 2045)
Author Name: break
Book cost: 3050

3) Book Name: rana ( 1980)
Author Name: rona
Book cost: 3400

4) Book Name: nera ( 45)
Author Name: rena
Book cost: 4500

5) Book Name: raja ( 202)
Author Name: tuna
Book cost: 5000
```

6. Kode :

```c
#include <stdio.h>
#define MAKS 100

// Fungsi untuk input array
void inputArray(int arr[], int *n) {
    printf("Masukkan jumlah elemen array: ");
    scanf("%d", n);

    for (int i = 0; i < *n; i++) {
        printf("Elemen ke-%d: ", i);
        scanf("%d", &arr[i]);
    }
}

// Fungsi untuk mencari 3 angka terbesar dan indeksnya
void cariTigaTerbesar(int arr[], int n, int tigaTerbesar[], int indeks[]) {
    // Inisialisasi nilai awal
    for (int i = 0; i < 3; i++) {
        tigaTerbesar[i] = -2147483648;  // Nilai minimum dari int
        indeks[i] = -1;
    }

    for (int i = 0; i < n; i++) {
        if (arr[i] > tigaTerbesar[0]) {
            // Geser ke kanan
            tigaTerbesar[2] = tigaTerbesar[1];
            indeks[2] = indeks[1];

            tigaTerbesar[1] = tigaTerbesar[0];
            indeks[1] = indeks[0];

            tigaTerbesar[0] = arr[i];
            indeks[0] = i;
        } else if (arr[i] > tigaTerbesar[1]) {
            tigaTerbesar[2] = tigaTerbesar[1];
            indeks[2] = indeks[1];

            tigaTerbesar[1] = arr[i];
            indeks[1] = i;
        } else if (arr[i] > tigaTerbesar[2]) {
            tigaTerbesar[2] = arr[i];
            indeks[2] = i;
        }
    }
}

// Fungsi untuk menampilkan hasil
void tampilkanTigaTerbesar(int tigaTerbesar[], int indeks[]) {
    printf("\nTiga angka terbesar beserta indeksnya:\n");
    for (int i = 0; i < 3; i++) {
        if (indeks[i] != -1)
            printf("Ke-%d terbesar: %d (indeks %d)\n", i+1, tigaTerbesar[i], indeks[i]);
    }
}

// Fungsi utama
int main() {
    int arr[MAKS], n;
    int tigaTerbesar[3], indeks[3];

    inputArray(arr, &n);
    if (n < 3) {
        printf("Jumlah elemen harus minimal 3.\n");
        return 1;
    }

    cariTigaTerbesar(arr, n, tigaTerbesar, indeks);
    tampilkanTigaTerbesar(tigaTerbesar, indeks);

    return 0;
}
```

Output :

```
Praktikan@CDSR3-33 MSYS /c/iam
$ ./a.exe
Masukkan jumlah elemen array: 3
Elemen ke-0: 1
Elemen ke-1: 2
Elemen ke-2: 3

Tiga angka terbesar beserta indeksnya:
Ke-1 terbesar: 3 (indeks 2)
Ke-2 terbesar: 2 (indeks 1)
Ke-3 terbesar: 1 (indeks 0)
```

7. Kode :

```cpp
#include <iostream>
#include <string>
#include <array>

using namespace std;

const int MAX_MOVIES = 5;

struct Movie {
    string title;
    string genre;
    int year;
};

using MovieArray = array<Movie, MAX_MOVIES>;

void inputMoviesData(MovieArray& movies);
void displayMoviesData(const MovieArray& movies);

int main() {
    MovieArray favoriteMovies;

    cout << "\nEnter your 5 favorite movies:\n";
    inputMoviesData(favoriteMovies);

    cout << "\nYou have entered the movies:\n";
    displayMoviesData(favoriteMovies);

    return 0;
}

void inputMoviesData(MovieArray& movies) {
    cin.ignore();
    for (int i = 0; i < MAX_MOVIES; ++i) {
        cout << "Enter film title " << (i + 1) << ": ";
        getline(cin, movies[i].title);

        cout << "Enter film genre " << (i + 1) << ": ";
        getline(cin, movies[i].genre);

        cout << "Enter film year " << (i + 1) << ": ";
        cin >> movies[i].year;
        cin.ignore();
    }
}

void displayMoviesData(const MovieArray& movies) {
    for (int i = 0; i < MAX_MOVIES; ++i) {
        cout << (i + 1) << ". " << movies[i].title
             << ", " << movies[i].genre<< " (" << movies[i].year << ")\n";
    }
}
```

Output :

```
Praktikan@CDSR3-33 MSYS /c/iam
$ ./a.exe

Enter your 5 favorite movies:
toni
Enter film title 1: Enter film genre 1: hror
Enter film year 1: 2020
Enter film title 2: marto
Enter film genre 2: komedi
Enter film year 2: 3000
Enter film title 3: rano
Enter film genre 3: lawak
Enter film year 3: 4050
Enter film title 4: jova
Enter film genre 4: sick
Enter film year 4: 1920
Enter film title 5: tase
Enter film genre 5: roman
Enter film year 5: 2040

You have entered the movies:
1. oni, hror (2020)
2. marto, komedi (3000)
3. rano, lawak (4050)
4. jova, sick (1920)
5. tase, roman (2040)
```

8. Kode :

```cpp
#include <iostream>

using namespace std;

int main(){
    int deret;
    float a, b, sum, temp;

    cout << "Program Deret Matematika \n";
    cout << "Masukkan dimensi deret yang diinginkan \n";
    cin >> deret;
    cout << endl;

    for (int i=1; i <= deret; i++){
        a = 1 + (i-1)*2;
        b = 3 + (i-1)*2;
        temp = a/b;
        cout << a << "/" << b <<endl;
        sum = sum + temp;
    }
    cout << endl << "Hasil penjumlahan dari deret adalah : " << sum;
    return 0;
}
```

Output :

```
Praktikan@CDSR3-33 MSYS /c/iam
$ ./a.exe
Program Deret Matematika
Masukkan dimensi deret yang diinginkan
5

1/3
3/5
5/7
7/9
9/11

Hasil penjumlahan dari deret adalah : 3.24358
```