

STRUKTUR DATA



NAMA: Bagas Diatama Wardoyo
NPM: 140810230061

Dikumpulkan tanggal :
4 Maret 2024

Universitas Padjadjaran
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
Program Studi S-1 Teknik Informatika
2024

- Soal 1

Source Code

```
/* Nama program : soal_1

Nama      : Bagas Diatama Wardoyo
NPM       : 140810230061
Tanggal buat : 14/03/2024
Deskripsi  : program modular untuk mengolah suatu kumpulan data bertipe Array of Integer
*****/

#include <iostream>

using namespace std;

void inputData(int arr[], int n);
int nilaiTertinggi(int arr[], int n);
int nilaiTerendah(int arr[], int n);
void sorting(int arr[], int n);
void swap(int& a, int& b);
void searching(int arr[], int n, int key);
void outputData(int arr[], int n);

int main(){
    int n, key;
    cout << "Banyak data: "; cin >> n;
    int arr[n];
    inputData(arr, n);
    cout << nilaiTertinggi(arr, n) << endl;
    cout << nilaiTerendah(arr, n) << endl;
    sorting(arr, n);
    outputData(arr, n);
    cin >> key;
    searching(arr, n, key);

    return 0;
}

//Fungsi input data
void inputData(int arr[], int n){
    cout << "Masukkan Data: ";
    for (int i = 0; i < n; i++)
    {
        cin >> arr[i];
    }
}

//Fungsi mencari nilai tertinggi
int nilaiTertinggi(int arr[], int n){
    int max = arr[0];
    for (int i = 1; i < n; i++)
    {
        if(arr[i] > max){
            max = arr[i];
        }
    }
    return max;
}
```

```
//Fungsi mencari nilai terendah
int nilaiTerendah(int arr[], int n){
    int min = arr[0];
    for (int i = 1; i < n; i++)
    {
        if(arr[i] < min){
            min = arr[i];
        }
    }
    return min;
}
```

```
//Fungis untuk melakukan swap dalam sorting
void swap(int&a, int& b){
    int temp = a;
    a = b;
    b = temp;
}
```

```
//Fungsi sorting
void sorting(int arr[], int n){
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j + i < n - 1; j++)
        {
            if (arr[j] > arr[j + 1])
            {
                swap(arr[j], arr[j+1]);
            }
        }
    }
}
}
```

```
//Fungsi untuk menampilkan data
void outputData(int arr[], int n){
    for (int i = 0; i < n; i++)
    {
        cout << arr[i] << " ";
    }
    cout << endl;
}
```

```
//Fungsi untuk mencari data
void searching(int arr[], int n, int key){
    bool found;
    for (int i = 0; i < n; i++)
    {
        if (arr[i] == key)
        {
            found = 1;
            break;
        }else{
            found = 0;
        }
    }
}
```

```

    }
    cout << (found == 1) ? "Ditemukkan" : "Tidak Ditemukkan";
}

```

Output program soal 1

```

Banyak data: 5
Masukkan Data: 6 8 4 5 10
Nilai tertinggi: 10
Nilai terendah: 4
Nilai disorting: 4 5 6 8 10
8
ditemukkan

```

• Soal 2

Source Code

```

/* Nama program : soal_2

Nama      : Bagas Diatama Wardoyo
NPM       : 140810230061
Tanggal buat : 14/03/2024
Deskripsi  : program modular untuk mencari jumlah dari setiap baris dan kolom suatu matriks.
*****

```

```
#include <iostream>
```

```
using namespace std;
```

```

void inputMatriks(int arr[][3]);
void sumMatriks_baris(int arr[][3]);
void sumMatriks_kolom(int arr[][3]);

```

```

int main()
{
    int matriks[3][3];
    inputMatriks(matriks);
    cout << endl;
    sumMatriks_baris(matriks);
    cout << endl;
    sumMatriks_kolom(matriks);

```

```
    return 0;
}
```

```

//Fungsi input matriks
void inputMatriks(int arr[][3])
{
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            cin >> arr[i][j];
        }
    }
}

```

```

//Fungsi untuk menjumlahkan setiap baris matriks
void sumMatriks_baris(int arr[][3])
{
    int sumBaris[] {0, 0, 0};
    for (int i = 0; i < 3; i++)

```

```

    {
        for (int j = 0; j < 3; j++)
        {
            sumBaris[i] += arr[i][j];
        }
    }
    for (int n : sumBaris)
    {
        cout << n << endl;
    }
}

```

```

//Fungsi untuk menjumlahkan setiap kolom matriks
void sumMatriks_kolom(int arr[][3])
{
    int sumBaris[] {0, 0, 0};
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            sumBaris[i] += arr[j][i];
        }
    }
    for (int n : sumBaris)
    {
        cout << n << endl;
    }
}

```

Output soal 2

```

1 0 4
5 2 1
2 1 2

5
8
5

8
3
7

```

- **Soal 3**

Source Code

```
/* Nama program : soal_3

Nama      : Bagas Diatama Wardoyo
NPM       : 140810230061
Tanggal buat : 14/03/2024
Deskripsi  : program modular untuk melakukan penjumlahan dan perkalian 2 buah matriks.
*****/

#include <iostream>

void inputArray(int arr_a[][3], int arr_b[][3]);
void perkalianMatriks(int arr_a[][3], int arr_b[][3]);
void penjumlahanMatriks(int arr_a[][3], int arr_b[][3]);
void cetakMatriks(int arr[][3]);

int main()
{
    int arr_a[3][3], arr_b[3][3];
    inputArray(arr_a, arr_b);
    perkalianMatriks(arr_a, arr_b);
    penjumlahanMatriks(arr_a, arr_b);

    return 0;
}

//Fungsi input nilai matriks
void inputArray(int arr_a[][3], int arr_b[][3])
{
    std::cout << "Masukkan array a (3x3): " << '\n';
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            std::cin >> arr_a[i][j];
        }
    }
    std::cout << "Masukkan array b (3x3): " << '\n';
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            std::cin >> arr_b[i][j];
        }
    }
}

//Fungsi mencetak matriks
void cetakMatriks(int arr[][3], int operasi)
{
    if (operasi == 1)
    {
        std::cout << "Hasil Perkalian: " << '\n';
    }
    else if (operasi == 2)
    {
        std::cout << "Hasil penjumlahan: " << '\n';
    }
}
```

```

    }
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            std::cout << arr[i][j] << " ";
        }
        std::cout << std::endl;
    }
}

```

```

//Fungsi perkalian matriks
void perkalianMatriks(int arr_a[][3], int arr_b[][3])
{
    int result[3][3];
    int sum = 0;
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            for (int k = 0; k < 3; k++)
            {
                sum += arr_a[i][k] * arr_b[k][j];
            }
            result[i][j] = sum;
            sum = 0;
        }
    }
    cetakMatriks(result, 1);
}

```

```

//Fungsi penjumlahan matriks
void penjumlahanMatriks(int arr_a[][3], int arr_b[][3])
{
    int result[3][3];
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            result[i][j] = arr_a[i][j] + arr_b[i][j];
        }
    }
    cetakMatriks(result, 2);
}

```

Output Soal 3

```
Masukkan array_a (3x3):
5 8 4
1 2 3
9 6 3
Masukkan array_b (3x3):
1 4 7
2 5 8
3 6 9
Hasil Perkalian:
33 84 135
14 32 50
30 84 138
Hasil penjumlahan:
6 12 11
3 7 11
12 12 12
```

- **Soal 4**

Source Code

```
/* Nama program : soal 4
Nama      : Bagus Diatama Wardoyo
NPM       : 140810230061
Tanggal buat : 14/03/2024
Deskripsi  : Pengolahan Data mahasiswa menggunakan struct
*****/
#include <iostream>

using namespace std;

//Struct mahasiswa berisi nama, npm, dan nilai
struct Mahasiswa
{
    string npm, nama;
    int nilai;
};

//Menginisialkan struct Mahasiswa sebagai dataMhs[10]
typedef Mahasiswa dataMhs[10];

//Menginisiasi berapa banyak data yang akan di olah
void banyakData(int &n)
{
    cout << "Banyak Data: ";
    cin >> n;
}

//Fungsi untuk menginput data mahasiswa
void input(dataMhs mhs, int n)
{
    for (int i = 0; i < n; i++)
    {
        cout << "Masukkan data mahasiswa ke- " << (i + 1) << endl;
        cout << "NPM\t: ";
        cin >> mhs[i].npm; //input npm
        cin.ignore();
        cout << "Nama\t: ";
```



```

        getline(cin, mhs[i].nama); // input nama
        cout << "Nilai\t: ";
        cin >> mhs[i].nilai; // input nilai
    }
}

```

```

//Fungsi menukar index data mahasiswa sesuai npm
void swap(Mahasiswa &a, Mahasiswa &b)
{
    Mahasiswa temp = a;
    a = b;
    b = temp;
}

```

```

//Fungsi memngurutkan data mahasiswa sesuai npm
void ascendingSort(dataMhs a, int n)
{
    for (int i = n - 1; i > 0; i--)
    {
        for (int j = 0; j < i; j++)
        {
            if (a[j].npm > a[j + 1].npm)
            {
                swap(a[j], a[j + 1]);
            }
        }
    }
}

```

```

//Fungsi untuk menghitung rata rata dari seluruh nilai mahasiswa
void average(dataMhs a, int n, float &avg) {
    float total = 0;
    for (int i = 0; i < n; i++) {
        total += a[i].nilai;
    }
    avg = total / n;
}

```

```

//Fungsi untuk menentukan nilai maksimal dari seluruh data nilai mahasiswa
void findMax(dataMhs a, int n, float &maxValue) {
    maxValue = a[0].nilai;
    for (int i = 1; i < n; i++) {
        if (a[i].nilai > maxValue) {
            maxValue = a[i].nilai;
        }
    }
}

```

```

//Fungsi untuk mencetak seluruh data
void cetakMahasiswa(dataMhs mhs, int n){
    cout << "\nDaftar Nilai Mahasiswa Struktur Data " << endl;
    cout << "-----" << endl;
    cout << "No\t " << "NMP\t" << "Nama\t" << "Nilai\t" << endl;
    cout << "-----" << endl;
    for (int i = 0; i < n; i++) {
        cout << i + 1 << "\t" << mhs[i].npm << "\t" << mhs[i].nama << "\t" << mhs[i].nilai;
        cout << endl;
    }
}

```

```

    }
    cout << "-----" << endl;
}

```

```

//Fungsi untuk mencari data mahasiswa berdasarkan NPM
void cariData(string npm, dataMhs data, int n){
    for (int i = 0; i < n; i++)
    {
        if (npm == data[i].npm)
        {
            cout << "Ditemukan NPM = " << npm << ", Nama = " << data[i].nama << ", Nilai = " <<
data[i].nilai << endl;
        }
    }
}
}

```

```

int main()
{
    dataMhs mhs; //deklarasi struct sebagai mhs
    int n; //deklarasi n sebagai banyak data, dan max sebagai nilai maksimum
    float avg, max; //deklarasi avg sebagai rata rata seluruh nilai
    string npm; //deklarasi npm sebagai input npm
    banyakData(n);
    input(mhs, n);
    ascendingSort(mhs, n);
    cetakMahasiswa(mhs, n);
    average(mhs, n, avg);
    findMax(mhs, n, max);
    cout << "Nilai Rata rata   : " << avg << endl;
    cout << "Nilai Tertinggi   : " << max << endl;
    cout << "-----Pencarian-----" << endl;
    cout << "Masukkan NPM Pencarian : "; cin >> npm;
    cariData(npm, mhs, n);

    return 0;
}

```

Output Soal 4

```
Banyak Data: 3
Masukkan data mahasiswa ke- 1
NPM      : 008
Nama     : Bagus
Nilai    : 85
Masukkan data mahasiswa ke- 2
NPM      : 004
Nama     : Diatama
Nilai    : 48
Masukkan data mahasiswa ke- 3
NPM      : 001
Nama     : Wardoyo
Nilai    : 100

Daftar Nilai Mahasiswa Struktur Data
-----
No       NMP      Nama      Nilai
-----
1        001      Wardoyo 100
2        004      Diatama 48
3        008      Bagus   85
-----

Nilai Rata rata   : 77.6667
Nilai Tertinggi   : 100
-----Pencarian-----
Masukkan NPM Pencarian : 001
Ditemukan NPM = 001, Nama = Wardoyo, Nilai = 100
```