

2022



PRAKTIKUM STRUKTUR DATA

NAFIATUN SHOLIAH, S.KOM., M.CS.

PRODI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA

PRAKTIKUM 2

Array

1. Array 1 Dimensi

Deklarasi :

Array 1 Dimensi : `TipeData NamaArray[Ukuran];`

Inisialisasi :

1. Statis

1. Implisit

```
int ar[] = {4, 8, 9, 2, 7};  
int ar[]{4, 8, 9, 2, 7}; // atau menggunakan direct  
                        // initialization list
```

2. Eksplisit

```
int ar[5] = {4, 8, 9, 2, 7};  
int ar[5]{4, 8, 9, 2, 7}; // direct initialization list
```

2. Dinamis

```
for(int i = 0; i < 5; ++i) {  
    cout << "elemen ke-" << i << ": ";  
    cin >> ar[i];  
}
```

Pengaksesan :

1. Merujuk pd nilai indeks

```
cout << "elemen ke-0: " << ar[0] << endl;  
cout << "elemen ke-1: " << ar[1] << endl;  
cout << "elemen ke-2: " << ar[2] << endl;  
cout << "elemen ke-3: " << ar[3] << endl;  
cout << "elemen ke-4: " << ar[4] << endl;
```

2. Menggunakan struktur perulangan

```
for(int i = 0; i < 5; ++i)  
cout << "elemen ke-" << i << ": "  
<< ar[i] << endl;
```

3. Menggunakan *range-based-for*

```
for(auto n : ar) {  
    cout << n << " ";  
}
```

Struktur di atas dapat dibaca sebagai "untuk semua n dalam ar, lakukan ...".

2. Array Multidimensi

Deklarasi :

- Array 2 Dimensi : `TipeData NamaArray[baris][kolom];`
- Array n-Dimensi : `TipeData NamaArray[Ukuran1]...[UkuranN]`

Contoh deklarasi array 2 dimensi ukuran 4x3

```
char ar2d[4][3];
```

Inisialisasi :

1. Statis

```
char ar2d[4][3] = {  
    {'d', 'z', 'x'},  
    {'c', 'v', 'b'},  
    {'a', 'm', 'n'},  
    {'s', 'd', 'f'}  
};
```

Atau

```
ar2d[0][0] = 'd';  
ar2d[0][1] = 'z';  
ar2d[0][2] = 'x';  
ar2d[1][0] = 'c';  
ar2d[1][1] = 'v';  
ar2d[1][2] = 'b';  
ar2d[2][0] = 'a';  
ar2d[2][1] = 'm';  
ar2d[2][2] = 'n';  
ar2d[3][0] = 's';  
ar2d[3][1] = 'd';  
ar2d[3][2] = 'f';
```

2. Dinamis

```
for(int i = 0; i < 4; ++i)  
{  
    for(int j = 0; j < 3; ++j)  
    {  
        cout << "elemen [" << i << ", " << j << "]:" ;  
        cin >> ar2d[i][j];  
    }  
}
```

Atau

```

for(int i = 0; i < 4; ++i)
{
    for(int j = 0; j < 3; ++j)
    {
        for(int k = 0; k < 2; ++k)
        {
            cout << "elemen [" << i << ", "
                    << j << ", "
                    << k << "]: ";
            cin >> ar3d[i][j][k];
        }
    }
}

```

Pengaksesan :

Pengaksesan dapat dilakukan satu persatu langsung merujuk pada indeks atau serentak. Contoh pengaksesan secara serentak pada array 2 dimensi :

```

for(int i = 0; i < 4; ++i)
{
    for(int j = 0; j < 3; ++j)
    {
        cout << arr2d[i][j] << "\t" ;
        cout << '\n';
    }
}

```

3. Studi Kasus : Statistik Sederhana

Program untuk perhitungan statistik sederhana yang dapat mencari data terbesar, terkecil, dan nilai rata – rata. Ketentuan :

1. Data disimpan dalam array berukuran maksimal 20
2. Input dilakukan secara dinamis oleh pengguna melalui keyboard dengan jumlah maksimal 20 data (boleh kurang)

Penyelesaian:

```
/**
 * program statistik sederhana
 * menampilkan data terbesar, terkecil, dan rata - rata
 */

#include <iostream>
using namespace std;

int get_max(int ar[], int sz);
int get_min(int ar[], int sz);
double get_avg(int ar[], int sz);

int main()
{
    int data[20];
    int n;

    cout << "banyak data (max 20): ";
    cin >> n;

    for(int i = 0; i < n; ++i)
    {
        cout << "data ke-" << i << ": ";
        cin >> data[i];
    }

    cout << "max: " << get_max(data, n) << endl;
    cout << "min: " << get_min(data, n) << endl;
    cout << "avg: " << get_avg(data, n) << endl;

    return 0;
}

int get_max(int ar[], int sz)
{
    int imax{0};
    for(int i = 1; i < sz; ++i)
        if(ar[i] > ar[imax])
            imax = i;
    return ar[imax];
}
```

```

int get_min(int ar[], int sz)
{
    int imin{0};
    for(int i = 1; i < sz; ++i)
        if(ar[i] < ar[imin])
            imin = i;
    return ar[imin];
}

double get_avg(int ar[], int sz)
{
    double sum{0};
    for(int i = 0; i < sz; ++i)
        sum += ar[i];
    return sum / sz;
}

```

4. Latihan

1. Tuliskan nilai setiap elemen dari variabel array a di bawah ini

```
int a[3][3] = { {1, 8}, {2, 4, 6}, {5} };
```

2. Buatlah sebuah fungsi bernama testPalindrome untuk menguji apakah sebuah string bersifat palindrome, artinya dibaca dari kiri sama dengan dibaca dari kanan. Fungsi akan menghasilkan nilai 1 jika benar, dan 0 bila tidak.
3. Buatlah fungsi rekursif untuk mendapatkan bilangan terkecil dari n buah bilangan bulat yang tersimpan dalam sebuah variabel array.
4. Buatlah fungsi rekursif untuk membalik isi sebuah variabel array.
5. Apa yang dilakukan oleh program berikut:

```

#include <stdio.h>
#define SIZE 10

int whatIsThis(int [], int);

main() {
    int total, a[SIZE] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    total = whatIsThis(a, SIZE);
    printf("\nNilai variabel total adalah %d", total);
    return 0;
}

int whatIsThis(int b[], int size) {
    if (size == 1)
        return b[0];
    else
        return b[size-1] + whatIsThis(b, size-1);
}

```

5. Membuat Laporan Praktikum

Format :

Matakuliah	
Pertemuan ke	
Nama Praktikan	
NIM	
NILAI (diisi oleh dosen / asisten praktikum)	

A. Tujuan

1. Pengenalan dan Pemahaman Array dengan Studi Kasus

B. Praktikum

1. Latihan 1

- a) Tampilan Coding
- b) Hasil Running
- c) Penjelasan

2. Latihan 2

- a) Tampilan Coding
- b) Hasil Running
- c) Penjelasan

3. Latihan 3

- a) Tampilan Coding
- b) Hasil Running
- c) Penjelasan

4. dst

C. Kesimpulan

D. Referensi (optional)