

**LAPORAN PRAKTIKUM**  
**MODUL 2**  
**PENGENALAN C++ (BAGIAN KEDUA)**



**Disusun Oleh:**  
**Dhiemas Tulus Ikhsan 2311104046**  
**SE-07-02**

**Dosen :**  
**Wahyu Andi Saputra, S.Pd., M.Eng**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY**  
**PURWOKERTO**  
**2024**

## 1. Tujuan

- a. Memahami penggunaan pointer dan alamat memori
- b. Mengimplementasikan fungsi dan prosedur dalam pemrograman

## 2. Landasan Teori

### a. Array

Array merupakan struktur data yang menyimpan kumpulan elemen dengan tipe yang sama di dalam satu wadah, yang dapat diakses menggunakan indeks. Setiap elemen dalam array diidentifikasi dengan indeks, yang biasanya dimulai dari 0. Misalnya, dalam array `int arr[5] = {1, 2, 3, 4, 5};`, elemen pertama dapat diakses dengan `arr[0]`, yang menghasilkan nilai 1. Keuntungan menggunakan array termasuk akses cepat ke elemen melalui indeks dan kemudahan dalam iterasi, namun array memiliki ukuran tetap yang ditentukan saat pembuatan dan tidak dapat diubah selama eksekusi program.

### b. Array Multidimensi

Array multidimensi adalah struktur data yang memungkinkan penyimpanan elemen dalam lebih dari satu dimensi, biasanya digunakan untuk mewakili data yang terorganisir dalam bentuk tabel atau matriks. Contohnya adalah array dua dimensi yang dapat dideklarasikan sebagai `int arr[3][4];`, yang berarti array tersebut memiliki 3 baris dan 4 kolom. Setiap elemen dalam array multidimensi diakses menggunakan lebih dari satu indeks, misalnya, `arr[1][2]` untuk mengakses elemen pada baris kedua dan kolom ketiga. Keuntungan utama dari array multidimensi adalah kemampuannya untuk merepresentasikan data yang lebih kompleks, seperti matriks atau grid, dengan cara yang terstruktur, namun ukuran setiap dimensi harus ditentukan saat deklarasi, sehingga tidak bisa diubah selama program berjalan.

### c. Pointer

Variabel pointer adalah tipe variabel yang menyimpan alamat memori dari variabel lain, yang biasanya ditunjukkan dalam format heksadesimal. Dengan menggunakan pointer, Anda dapat mengakses dan memanipulasi nilai dari variabel yang ditunjuk oleh alamat tersebut. Misalnya, jika Anda memiliki variabel `int x`, Anda dapat mendeklarasikan pointer `int *p` yang menyimpan alamat dari `x`. Ini memungkinkan Anda untuk melakukan operasi seperti dereferensi untuk mendapatkan nilai yang disimpan di alamat tersebut, sehingga pointer menjadi alat yang kuat untuk pengelolaan memori dinamis, pengoperasian struktur data, dan pengimplementasian fungsi yang efisien dalam pemrograman C++.

### 3. GUIDED

#### a. Array

Program yang menunjukkan cara mendeklarasikan dan mencetak elemen dari sebuah array. Berikut adalah penjelasan rinci tentang setiap bagiannya:

```
int main() {  
    int nilai[5]={1,2,3,4,5};
```

1.

Ini adalah titik masuk utama dari program, di mana eksekusi dimulai. Di sini, sebuah array bernama `nilai` dideklarasikan dengan ukuran 5 dan diisi dengan nilai 1, 2, 3, 4, dan 5. Elemen-elemen array diindeks mulai dari 0 hingga 4.

```
    cout << nilai[0];  
    cout << nilai[1];  
    cout << nilai[2];  
    cout << nilai[3];  
    cout << nilai[4];
```

2.

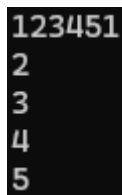
Bagian ini mencetak setiap elemen array secara individual ke layar. Namun, tidak ada spasi atau pemisah antara elemen yang dicetak, sehingga hasilnya akan terlihat berurutan.

```
    for(int i=0; i<5; i++){  
        cout << nilai[i] << endl;  
    }
```

3.

```
    return 0;
```

Di sini, loop `for` digunakan untuk mencetak semua elemen array secara berurutan. Loop ini mengiterasi dari 0 hingga 4 (jumlah elemen dalam array) dan mencetak setiap elemen diikuti dengan `endl`, yang menambahkan baris baru setelah setiap elemen. Ini membuat output lebih rapi, dengan setiap nilai ditampilkan di baris terpisah.



```
123451  
2  
3  
4  
5
```

4.

Pernyataan ini menandakan bahwa program telah selesai dijalankan dengan sukses.

## b. Array 2 Dimensi

Program ini mencetak elemen-elemen dari array tersebut dalam bentuk matriks. Berikut adalah penjelasan dari setiap bagiannya:

```
int main() {  
    int nilai[3][4] = {  
        {1, 2, 3, 4},  
        {5, 6, 7, 8},  
        {9, 10, 11, 12}  
    };  
}
```

1. Fungsi ini merupakan titik awal dari program, tempat di mana eksekusi dimulai. Array dua dimensi `nilai` dengan ukuran `3x4` dideklarasikan dan diinisialisasi. Artinya, array ini memiliki 3 baris dan 4 kolom, dengan elemen-elemen yang diatur dalam format matriks. Misalnya, elemen `nilai[0][0]` adalah 1, `nilai[1][2]` adalah 7, dan seterusnya.

```
for (int i = 0; i < 3; i++) {  
    for (int j = 0; j < 4; j++) {  
        cout << nilai[i][j] << " ";  
    }  
    cout << endl;  
}
```

2. Bagian ini menggunakan dua loop `for` bersarang untuk mencetak semua elemen dari array:
  - Loop luar (`for (int i = 0; i < 3; i++)`) mengiterasi melalui baris (3 baris).
  - Loop dalam (`for (int j = 0; j < 4; j++)`) mengiterasi melalui kolom (4 kolom) dari setiap baris.
  - `cout << nilai[i][j] << " ";` mencetak elemen pada indeks `i` dan `j`, diikuti oleh spasi agar elemen-elemen ditampilkan berdampingan.
  - `cout << endl;` digunakan setelah loop dalam untuk memulai baris baru setelah semua elemen dari satu baris dicetak, sehingga output ditampilkan dalam format matriks.

### 3. Hasil Output:

Program ini akan menghasilkan output sebagai berikut:

```
1 2 3 4  
5 6 7 8  
9 10 11 12
```

## c. Pointer

Program ini mengakses alamat memori dan nilai dari suatu variabel. penjelasan dari setiap bagiannya:

```
int main() {  
    int x, y;  
    int *px;
```

- 1.

Fungsi utama di mana program mulai dieksekusi.

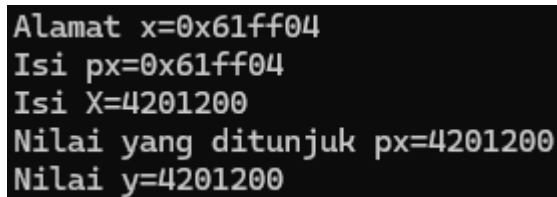
- `x` dan `y` adalah variabel integer.
- `\*px` adalah variabel pointer bertipe `int\*`, yang berarti dapat menyimpan alamat memori dari variabel bertipe `int`.

```
px = &x;  
y = *px;
```

- Baris ini menginisialisasi `px` dengan alamat memori dari `x` menggunakan operator `&`. Dengan demikian, `px` sekarang menyimpan alamat `x`. Di sini, `\*px` (disebut dereferensi pointer) mengacu pada nilai yang ada di alamat yang ditunjuk oleh `px`, yaitu `x`. Karena `x` belum diberi nilai, nilainya bisa jadi acak atau default (tergantung sistem). Nilai tersebut kemudian disalin ke `y`.

```
cout << "Alamat x=" << &x << endl;  
cout << "Isi px=" << px << endl;  
cout << "Isi X=" << x << endl;  
cout << "Nilai yang ditunjuk px=" << *px << endl;  
cout << "Nilai y=" << y << endl;
```

- `cout << "Alamat x=" << &x << endl;` menampilkan alamat memori dari `x`.
  - `cout << "Isi px=" << px << endl;` menampilkan alamat `x` yang disimpan dalam `px`.
  - `cout << "Isi X=" << x << endl;` menampilkan nilai `x` (karena belum diinisialisasi secara eksplisit, nilainya bisa acak).
  - `cout << "Nilai yang ditunjuk px=" << \*px << endl;` menampilkan nilai `x` melalui pointer `px`.
  - `cout << "Nilai y=" << y << endl;` menampilkan nilai `y`, yang merupakan hasil penyalinan dari nilai yang ditunjuk oleh `px` (nilai `x`).



```
Alamat x=0x61ff04  
Isi px=0x61ff04  
Isi X=4201200  
Nilai yang ditunjuk px=4201200  
Nilai y=4201200
```

- Menandakan bahwa program berakhir dengan status normal.

#### 4. UNGUIDED

- a. Buatlah program untuk menampilkan Output seperti berikut dengan data yang diinputkan oleh user!

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4
5 int main() {
6     int n;
7
8     // Input jumlah elemen array
9     cout << "Masukkan jumlah elemen array: ";
10    cin >> n;
11
12    vector<int> arr(n);
13
14    // Input elemen array
15    cout << "Masukkan elemen array: ";
16    for (int i = 0; i < n; i++) {
17        cin >> arr[i];
18    }
19
20    // Menampilkan Data Array
21    cout << "Data Array: ";
22    for (int i = 0; i < n; i++) {
23        cout << arr[i];
24        if (i < n - 1) {
25            cout << " "; // Spasi antara angka
26        }
27    }
28    cout << endl;
29
30    // Menampilkan nomor genap
31    cout << "Nomor Genap: ";
32    for (int i = 0; i < n; i++) {
33        if (arr[i] % 2 == 0) {
34            cout << arr[i] << ", ";
35        }
36    }
37
38    // Menghapus koma terakhir jika ada
39    if (n > 0) {
40        cout << "\b\b "; // Menghapus koma terakhir
41    }
42    cout << endl;
43
44    // Menampilkan nomor ganjil
45    cout << "Nomor Ganjil: ";
46    for (int i = 0; i < n; i++) {
47        if (arr[i] % 2 != 0) {
48            cout << arr[i] << ", ";
49        }
50    }
51
52    // Menghapus koma terakhir jika ada
53    if (n > 0) {
54        cout << "\b\b "; // Menghapus koma terakhir
55    }
56    cout << endl;
57
58    return 0;
59 }
```

menampilkan elemen array, serta memisahkan nomor genap dan ganjil. penjelasan:

1. Input Jumlah Elemen dan Deklarasi Array:
  - Meminta user memasukkan jumlah elemen n.
  - Membuat `vector<int> arr(n);` yang secara otomatis membuat array dengan n elemen.
2. Input Elemen Array:
  - Menggunakan loop for untuk mengisi elemen array dari input user.
3. Menampilkan Data Array:
  - Menggunakan loop for untuk mencetak elemen array dengan spasi antar elemen.
4. Menampilkan Nomor Genap:

- Menggunakan loop for dengan kondisi if ( $\text{arr}[i] \% 2 == 0$ ) untuk mencetak elemen yang genap, diikuti oleh koma.
5. Menghapus Koma Terakhir:
    - Menggunakan `\b\b` untuk menghapus koma dan spasi terakhir dari output genap dan ganjil, agar hasilnya lebih rapi.
  6. Menampilkan Nomor Ganjil:
    - Sama seperti genap, tapi menggunakan kondisi if ( $\text{arr}[i] \% 2 != 0$ ) untuk mencetak elemen yang ganjil.
- b.** Buatlah program Input array tiga dimensi tetapi jumlah atau ukuran elemennya diinputkan oleh user!

```

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int x, y, z;
6
7      // Input ukuran array 3D
8      cout << "Masukkan ukuran dimensi x: ";
9      cin >> x;
10     cout << "Masukkan ukuran dimensi y: ";
11     cin >> y;
12     cout << "Masukkan ukuran dimensi z: ";
13     cin >> z;
14
15     int arr[x][y][z];
16
17     // Input elemen array
18     cout << "Masukkan elemen array 3D:" << endl;
19     for(int i = 0; i < x; i++) {
20         for(int j = 0; j < y; j++) {
21             for(int k = 0; k < z; k++) {
22                 cout << "Elemen [" << i << "]" << j << "]" << k << ": ";
23                 cin >> arr[i][j][k];
24             }
25         }
26     }
27
28     // Menampilkan elemen array 3D
29     cout << "\nElemen array 3D:" << endl;
30     for(int i = 0; i < x; i++) {
31         for(int j = 0; j < y; j++) {
32             for(int k = 0; k < z; k++) {
33                 cout << "arr[" << i << "]" << j << "]" << k << " = " << arr[i][j][k] << endl;
34             }
35         }
36     }
37
38     return 0;
39 }
40

```

menerima input ukuran untuk array 3D dan mengisi elemen-elemennya berdasarkan input pengguna:

1. Input Ukuran Array 3D:
  - Meminta pengguna untuk memasukkan ukuran dimensi x, y, dan z.
  - Membuat array 3D `arr[x][y][z]` sesuai ukuran yang dimasukkan.
2. Input Elemen Array:
  - Menggunakan tiga loop for bersarang untuk mengisi setiap elemen dari array 3D.
  - Pengguna memasukkan nilai untuk setiap indeks `[i][j][k]`.
3. Menampilkan Elemen Array 3D:

- Menggunakan tiga loop for bersarang untuk mencetak semua elemen array 3D.
  - Menampilkan nilai dari setiap elemen dengan format `arr[i][j][k]`
- c. Buatlah program menu untuk mencari nilai Maksimum, Minimum dan Nilai rata – rata dari suatu array dengan input yang dimasukan oleh user!

Syntax :

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int n, pilihan;
6     float sum = 0;
7
8     // Input jumlah elemen array
9     cout << "Masukkan jumlah elemen array: ";
10    cin >> n;
11
12    int arr[n];
13
14    // Input elemen array
15    cout << "Masukkan elemen array: ";
16    for(int i = 0; i < n; i++) {
17        cin >> arr[i];
18        sum += arr[i];
19    }
20
21    // Menu pilihan
22    do {
23        cout << "Menu Pilihan: " << endl;
24        cout << "1. Cari Nilai Maksimum" << endl;
25        cout << "2. Cari Nilai Minimum" << endl;
26        cout << "3. Hitung Nilai Rata-rata" << endl;
27        cout << "4. Keluar" << endl;
28        cout << "Masukkan pilihan: ";
29        cin >> pilihan;
30
31        switch(pilihan) {
32            case 1: {
33                int max = arr[0];
34                for(int i = 1; i < n; i++) {
35                    if(arr[i] > max) {
36                        max = arr[i];
37                    }
38                }
39                cout << "Nilai Maksimum: " << max << endl;
40                break;
41            }
42            case 2: {
43                int min = arr[0];
44                for(int i = 1; i < n; i++) {
45                    if(arr[i] < min) {
46                        min = arr[i];
47                    }
48                }
49                cout << "Nilai Minimum: " << min << endl;
50                break;
51            }
52            case 3: {
53                float rata_rata = sum / n;
54                cout << "Nilai Rata-rata: " << rata_rata << endl;
55                break;
56            }
57            case 4: {
58                cout << "Keluar dari program." << endl;
59                break;
60            }
61            default: {
62                cout << "Pilihan tidak valid!" << endl;
63            }
64        } while(pilihan != 4);
65    } while(pilihan != 4);
66    return 0;
67 }

```

Output:



```
Masukkan jumlah elemen array: 2
Masukkan elemen array: 3 4

Menu Pilihan:
1. Cari Nilai Maksimum
2. Cari Nilai Minimum
3. Hitung Nilai Rata-rata
4. Keluar
Masukkan pilihan: 3
Nilai Rata-rata: 3.5

Menu Pilihan:
1. Cari Nilai Maksimum
2. Cari Nilai Minimum
3. Hitung Nilai Rata-rata
4. Keluar
Masukkan pilihan: 4
Keluar dari program.
```

1. **Input Ukuran dan Elemen Array:**
  - User memasukkan jumlah elemen n.
  - Array arr dengan n elemen dibuat, dan user mengisi setiap elemennya.
  - Total nilai elemen disimpan dalam sum untuk perhitungan rata-rata.
2. **Menu Pilihan:**
  - Menyediakan pilihan untuk user dalam loop do-while agar menu terus muncul sampai user memilih keluar (pilihan 4).
  - switch digunakan untuk menangani pilihan menu:
    - 1: Mencari nilai maksimum dengan iterasi pada arr.
    - 2: Mencari nilai minimum dengan iterasi pada arr.
    - 3: Menghitung rata-rata dengan  $\text{sum} / n$ .
    - 4: Keluar dari program.
3. **Keluar dari Loop:**
  - Loop berhenti ketika user memilih 4.