

LAPORAN PRAKTIKUM
Modul 2



Disusun Oleh:
Razhendriya Vania Ramadhan Suganjarsarwat - 2311104048
Kelas SE0702

Dosen :
WAHYU ANDI SAPUTRA

PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY
PURWOKERTO
2024

Tujuan

Memahami penggunaan *pointer* dan alamat memori
Mengimplementasikan fungsi dan prosedur dalam program

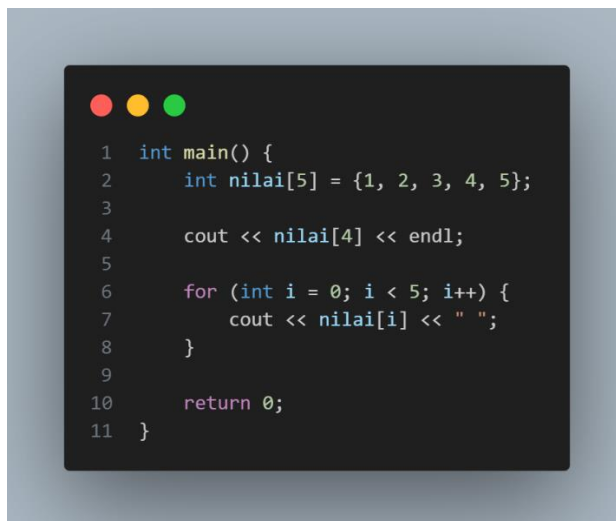
Landasan Teori

Array

Array merupakan kumpulan data dengan nama yang sama dan setiap elemen bertipe data sama.

Untuk mengakses setiap komponen / elemen *array* berdasarkan indeks dari setiap elemen.

Guided



```
1  int main() {
2      int nilai[5] = {1, 2, 3, 4, 5};
3
4      cout << nilai[4] << endl;
5
6      for (int i = 0; i < 5; i++) {
7          cout << nilai[i] << " ";
8      }
9
10     return 0;
11 }
```

Penjelasan:

- `int nilai[5] = {1, 2, 3, 4, 5};`: Mendeklarasikan array integer bernama nilai dengan 5 elemen yang diinisialisasi dengan nilai 1 hingga 5.
- `cout << nilai[4] << endl;`: Mencetak nilai elemen ke-5 pada array (indeks 4), yaitu 5.
- `for` loop: Mengiterasi array dan mencetak setiap nilai dari indeks 0 hingga 4 (yaitu, 1 2 3 4 5).
- `return 0;`: Mengakhiri program dengan sukses.

outputnya

5

1 2 3 4 5

```
1  int main() {  
2      int nilai[5] = {1, 2, 3, 4, 5};  
3  
4      cout << nilai[4] << endl;  
5  
6      for (int i = 0; i < 5; i++) {  
7          cout << nilai[i] << " ";  
8      }  
9  
10     return 0;  
11 }
```

Penjelasan:

- `int nilai[3][4]`: Mendeklarasikan array 2D integer dengan 3 baris dan 4 kolom.
- Loop bersarang for: Loop luar mengiterasi baris (i dari 0 hingga 2) dan loop dalam mengiterasi kolom (j dari 0 hingga 3). Nilai dari array dicetak baris per baris.
- `cout << nilai[i][j]`: Mengakses setiap elemen dalam array 2D dan mencetaknya.

outputnya

1 2 3 4

5 6 7 8

9 10 11 12

```
1  int main() {
2      int x, y;
3      int *px;
4
5      x = 10;
6      y = 20;
7      px = &x;
8
9      cout << "Alamat x = " << &x << endl;
10     cout << "Isi px = " << px << endl;
11     cout << "Nilai yang ditunjuk px = " << *px << endl;
12     cout << "Nilai y = " << y << endl;
13
14     getch();
15
16     return 0;
17 }
18
```

Penjelasan:

- `int *px;`: Mendeklarasikan pointer `px` yang bisa menyimpan alamat integer.
- `px = &x;`: Menyimpan alamat variabel `x` dalam `px`, sehingga `px` menunjuk ke `x`.
- `*px`: Mendereferensi pointer `px` untuk mendapatkan nilai dari `x`, yaitu 10.
- `cout << &x;`: Mencetak alamat memori dari variabel `x`.
- `cout << px;`: Mencetak nilai yang disimpan dalam `px`, yaitu alamat dari `x`.
- `cout << *px`: Mendereferensi `px` untuk mencetak nilai yang ditunjuk, yaitu nilai `x` (10).
- `getch()`: Menunggu penekanan tombol sebelum melanjutkan (tergantung pada kompiler tertentu).

```
1  int main() {
2      int x, y;
3      int *px;
4
5      x = 10;
6      y = 20;
7      px = &x;
8
9      cout << "Alamat x = " << &x << endl;
10     cout << "Isi px = " << px << endl;
11     cout << "Nilai yang ditunjuk px = " << *px << endl;
12     cout << "Nilai y = " << y << endl;
13
14     getch();
15
16     return 0;
}
```

Penjelasan:

- penjumlahan(int a, int b): Sebuah fungsi yang menerima dua integer a dan b dan mengembalikan jumlahnya.
- int hasil = penjumlahan(5, 3);: Memanggil fungsi penjumlahan dengan argumen 5 dan 3, menyimpan hasilnya (8) dalam variabel hasil.
- cout << hasil: Mencetak hasil dari penjumlahan.

Outputnya

Hasil penjumlahan: 8

```
1 #include <iostream>
2 using namespace std;
3
4 int penjumlahan(int a, int b) {
5     return a + b;
6 }
7
8 void greet(string name) {
9     cout << "Hello, " << name << "!" << endl;
10 }
11
12 int main() {
13     int hasil = penjumlahan(5, 3);
14     cout << "Hasil = " << hasil << endl;
15
16     greet("Alice");
17
18     return 0;
19 }
```

Penjelasan:

- `penjumlahan(int a, int b)`: Fungsi yang mengembalikan jumlah dua bilangan integer.
- `void greet(string name)`: Fungsi void (tidak mengembalikan nilai) yang menerima sebuah argumen string dan mencetak pesan sapaan dengan nama tersebut.
- `greet("Alice")`: Memanggil fungsi `greet` dan mencetak "Hello, Alice!".

outputnya

Hasil = 8

Hello, Alice!

```
1  int main() {
2      int n;
3      cout << "Masukkan panjang array: ";
4      cin >> n;
5
6      vector<int> data_array(n), nomor_genap, nomor_ganjil;
7
8      for (int i = 0; i < n; i++) {
9          cout << "Masukkan elemen ke-" << i+1 << " : ";
10         cin >> data_array[i];
11     }
12
13     for (int i = 0; i < n; i++) {
14         if (data_array[i] % 2 == 0) {
15             nomor_genap.push_back(data_array[i]);
16         } else {
17             nomor_ganjil.push_back(data_array[i]);
18         }
19     }
20
21     cout << "Data Array: ";
22     for (int i = 0; i < n; i++) {
23         cout << data_array[i] << " ";
24     }
25     cout << endl;
26
27     cout << "Nomor Genap: ";
28     for (int i = 0; i < nomor_genap.size(); i++) {
29         cout << nomor_genap[i] << " ";
30     }
31     cout << endl;
32
33     cout << "Nomor Ganjil: ";
34     for (int i = 0; i < nomor_ganjil.size(); i++) {
35         cout << nomor_ganjil[i] << " ";
36     }
37     cout << endl;
38
39     return 0;
40 }
```

Penjelasan:

- Program meminta input panjang array dari user.
- Setelah itu, user memasukkan elemen-elemen array.
- Program akan memisahkan elemen array ke dalam dua kategori: genap dan ganjil.
- Program kemudian mencetak seluruh elemen array, elemen genap, dan elemen ganjil.

Outputnya

Masukkan panjang array: 10

Masukkan elemen ke-1: 1

Masukkan elemen ke-2: 2

Masukkan elemen ke-3: 3

Masukkan elemen ke-4: 4

Masukkan elemen ke-5: 5

Masukkan elemen ke-6: 6

Masukkan elemen ke-7: 7

Masukkan elemen ke-8: 8

Masukkan elemen ke-9: 9

Masukkan elemen ke-10: 10

Data Array: 1 2 3 4 5 6 7 8 9 10

Nomor Genap: 2 4 6 8 10

Nomor Ganjil: 1 3 5 7 9

```
1
2  int main() {
3      int x, y, z;
4      cout << "Masukkan ukuran dimensi pertama: ";
5      cin >> x;
6      cout << "Masukkan ukuran dimensi kedua: ";
7      cin >> y;
8      cout << "Masukkan ukuran dimensi ketiga: ";
9      cin >> z;
10
11     int array_3d[x][y][z];
12
13     for (int i = 0; i < x; i++) {
14         for (int j = 0; j < y; j++) {
15             for (int k = 0; k < z; k++) {
16                 cout << "Masukkan elemen (" << i << ", " << j << ", " << k << "): ";
17                 cin >> array_3d[i][j][k];
18             }
19         }
20     }
21
22     cout << "Array 3 Dimensi:" << endl;
23     for (int i = 0; i < x; i++) {
24         for (int j = 0; j < y; j++) {
25             for (int k = 0; k < z; k++) {
26                 cout << "array_3d[" << i << "][" << j << "][" << k << "] = " << array_3d[i][j][k] << endl;
27             }
28         }
29     }
30
31     return 0;
32 }
33
```

Penjelasan:

- Program meminta tiga ukuran dimensi dari user (x, y, z).
- Setelah itu, user diminta untuk mengisi elemen-elemen array tiga dimensi sesuai dengan indeksinya.
- Program kemudian menampilkan semua elemen dari array tiga dimensi yang telah diinput oleh user.


```
1
2 int main() {
3     int x, y, z;
4     cout << "Masukkan ukuran dimensi pertama: ";
5     cin >> x;
6     cout << "Masukkan ukuran dimensi kedua: ";
7     cin >> y;
8     cout << "Masukkan ukuran dimensi ketiga: ";
9     cin >> z;
10
11     int array_3d[x][y][z];
12
13     for (int i = 0; i < x; i++) {
14         for (int j = 0; j < y; j++) {
15             for (int k = 0; k < z; k++) {
16                 cout << "Masukkan elemen (" << i << ", " << j << ", " << k << "): ";
17                 cin >> array_3d[i][j][k];
18             }
19         }
20     }
21
22     cout << "Array 3 Dimensi:" << endl;
23     for (int i = 0; i < x; i++) {
24         for (int j = 0; j < y; j++) {
25             for (int k = 0; k < z; k++) {
26                 cout << "array_3d[" << i << "][" << j << "][" << k << "] = " << array_3d[i][j][k] << endl;
27             }
28         }
29     }
30     return 0;
31 }
32 }
```

Penjelasan:

- Program meminta input jumlah elemen array dari user.
 - Setelah itu, user memasukkan nilai-nilai elemen array.
 - User dapat memilih apakah ingin menghitung nilai maksimum, minimum, atau rata-rata dari elemen-elemen array tersebut.
 - Program akan menghitung dan menampilkan hasil sesuai pilihan user.
- outputnya

Masukkan jumlah elemen array: 5

Masukkan elemen ke-1: 10

Masukkan elemen ke-2: 20

Masukkan elemen ke-3: 15

Masukkan elemen ke-4: 30

Masukkan elemen ke-5: 25

Pilih operasi:

1. Nilai Maksimum

2. Nilai Minimum

3. Nilai Rata-rata

Masukkan pilihan (1/2/3): 1

Nilai Maksimum: 30

Array: Digunakan untuk menyimpan kumpulan data dalam satu variabel, baik dalam bentuk satu dimensi maupun dua dimensi. Kita belajar bagaimana mengakses dan menampilkan elemen-elemen array menggunakan loop.

Pointer: Berguna untuk menyimpan alamat memori dari variabel lain. Pointers memungkinkan kita untuk mengakses dan memodifikasi data melalui alamatnya.

Fungsi: Digunakan untuk memisahkan logika program menjadi bagian-bagian yang terpisah. Kita mempelajari fungsi dengan nilai kembali (int penjumlahan) dan fungsi tanpa nilai kembali (void greet), yang membantu dalam modularisasi dan penggunaan ulang kode.



