

MODUL 8

POINTER

A. Tujuan Praktikum

Setelah mempelajari bab ini mahasiswa diharapkan mampu :

1. Menjelaskan fungsi pointer di dalam bahasa C++
2. Memahami perbedaan pointer dengan array

B. Waktu Praktikum

4 x 1 jam

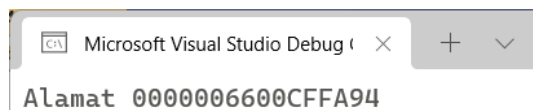
C. Petunjuk

1. Awali setiap kegiatan dengan berdoa dan tekad kuat, agar diberi kelancaran dalam belajar dan mendapatkan ilmu yang bermanfaat.
2. Pahami tujuan praktikum, dasar teori, dan latihan praktikum dengan baik.
3. Kerjakan tugas praktikum dengan baik, jujur, sabar dan teliti.
4. Tanyakan kepada asisten laboratorium apabila ada hal-hal yang kurang jelas.

D. Dasar Teori

Pointer di dalam bahasa C++ adalah variabel yang menyimpan alamat memori dari variabel lain. Alamat dalam variabel adalah alamat yang merepresentasikan lokasi data yang tersimpan di dalam RAM. Alamat dapat dilihat dengan mencetak &variabel pada c++.

```
int var = 20;  
cout << "Alamat " << &var;
```



Pointer dapat dideklarasikan dengan menggunakan sintaks berikut. Operator * digunakan setelah tipe data untuk mendeklarasikan pointer.

```
int* pVar;
```

Untuk menyimpan alamat variabel ke dalam pointer dapat dilihat pada contoh dibawah.

```
int var;  
int* pvar;  
var = 20;  
pvar = &var;
```

```
// Atau
```

```
int var = 20;  
int* pvar = &var;
```

Karena pointer merujuk pada alamat variabel lain, maka kita bisa mencetak nilai dari variabel yang dirujuk.

```
int var = 20;  
int* pvar = &var;
```

```
cout << *pvar; // Output: 20
```

Kita juga bisa mengubah nilai yang disimpan ke dalam var dengan menggunakan pointer.

```
int var = 20;  
int* pvar = &var;
```

```
*pvar = 30;
```

```
cout << *pvar; // Output: 30
```

Pointer dan array

Pointer juga dapat digunakan untuk menyimpan alamat cell di sebuah array dan digunakan untuk mengakses array tersebut. Perhatikan contoh berikut pointer ptr digunakan untuk merujuk elemen pertama pada array.

```
int data[5];  
int* ptr;  
  
ptr = data;  
// Penggunaan diatas sama dengan  
ptr = &data[0];
```

Pointer juga dapat digunakan untuk membuat array dengan ukuran yang sesuai input pengguna di dalam visual studio. Perhatikan contoh berikut.

```
#include <iostream>  
  
using namespace std;  
  
int main() {  
    int n = 5;  
    cout << "Panjang array: ";  
    cin >> n;  
  
    int* data = new int[n];  
  
    for (int i = 0; i < n; i++) {  
        cout << "Data ke-" << i + 1 << ": ";  
        cin >> data[i];  
    }  
  
    for (int i = 0; i < n; i++) {
```

```

        cout << data[i] << " ";
    }
}

```

Indeks pada pointer

Sama seperti array, pointer yang merujuk pada array juga memiliki indeks. Penggunaan indeks pada pointer akan merujuk pada elemen di dalam array. Perhatikan contoh berikut.

```

int data[5] = { 10, 20, 30, 40, 50 };
int* ptr = data;

// ptr + 1 == &data[1]
// ptr + 2 == &data[2]
// ptr + 3 == &data[3]

// *(ptr + 1) == data[1]
// *(ptr + 2) == data[2]
// *(ptr + 3) == data[3]

```

Konstanta string dan pointer

Pointer juga dapat digunakan untuk membuat konstanta string. Yang berarti string tersebut tidak dapat diubah nilainya. Perhatikan kode dibawah ini akan error ketika di compile.

```

const char* str = "Hello";
cout << str; // OUTPUT: Hello

// Kode ini akan error
str[1] = 'a';

```

Pass by reference

Fungsi pada bahasa C++ memiliki 2 tipe parameter yaitu pass by value dan pass by reference. Pass by value berarti fungsi akan mengirimkan nilai atau data dari suatu variabel. Sedangkan pass by reference berarti mengirimkan pointer dari variabel tersebut sehingga apabila nilainya diubah, maka variabel asal juga akan berubah. Perhatikan kode berikut.

```

#include <iostream>

using namespace std;

// Pass by value
void fungsi1(int num) {
    num = 20;
}

// Pass by reference
void fungsi2(int& num) {

```

```

    num = 30;
}

int main() {
    int num = 10;

    fungsi1(num);
    cout << num << endl; // OUTPUT: 10

    fungsi2(num);
    cout << num << endl; // OUTPUT: 30
}

```

Pointer di dalam pointer

Sebuah variabel pointer juga dapat digunakan untuk merujuk pada pointer lain. Perhatikan contoh berikut.

```

#include <iostream>

using namespace std;

int main() {
    int num = 10;
    int* ptr1;
    int** ptr2;

    ptr1 = &num;
    ptr2 = &ptr1;

    cout << "tanpa pointer = " << num << endl;
    cout << "1 pointer = " << *ptr1 << endl;
    cout << "2 pointer = " << **ptr2 << endl;
}

```

Keuntungan Menggunakan Pointer

- Lebih sedikit waktu dalam eksekusi program
- Bekerja pada variabel asli
- Dengan bantuan pointer, kita dapat membuat struktur data (linked-list, stack, queue).
- Mengembalikan lebih dari satu nilai dari fungsi
- Mencari dan menyortir data besar dengan sangat mudah
- Alokasi memori secara dinamis

Kekurangan Menggunakan Pointer

- Terkadang dengan membuat pointer, kesalahan seperti itu masuk ke dalam program, yang sangat sulit untuk didiagnosis.
- Terkadang kebocoran pointer di memori juga dibuat.
- Jika memori tambahan tidak ditemukan maka program crash juga dapat terjadi.

E. Latihan Praktikum

1. Ketik kode program di bawah ini dengan menggunakan Visual Studio, kemudian compile dan jalankan program tersebut. Jelaskan dan simpulkan hasil program.

```
#include <iostream>

using namespace std;

int main() {
    int v[3] = { 10, 100, 200 };
    int* ptr;

    ptr = v;

    for (int i = 0; i < 3; i++) {
        cout << "Nilai *ptr = " << *ptr << endl;
        cout << "Nilai ptr = " << ptr << endl << endl;

        ptr++;
    }
}
```

2. Ketik kode program di bawah ini dengan menggunakan Visual Studio, kemudian compile dan jalankan program tersebut. Jelaskan dan simpulkan hasil program.

```
#include <iostream>

using namespace std;

int main() {
    int v[3] = { 10, 100, 200 };
    int* ptr;

    ptr = &v[1];

    cout << "Nilai *ptr = " << *(ptr - 1) << endl;
    cout << "Nilai ptr = " << ptr - 1 << endl << endl;

    cout << "Nilai *ptr = " << *ptr << endl;
    cout << "Nilai ptr = " << ptr << endl << endl;

    cout << "Nilai *ptr = " << *(ptr + 1) << endl;
    cout << "Nilai ptr = " << ptr << endl << endl;
}
```

3. Ketik kode program di bawah ini dengan menggunakan Visual Studio, kemudian compile dan jalankan program tersebut. Jelaskan dan simpulkan hasil program.

```
#include <iostream>

using namespace std;
```

```

void swap(int& a, int& b) {
    int temp = a;
    a = b;
    b = temp;
}

int main() {
    int a = 10;
    int b = 20;

    swap(a, b);

    cout << a << " " << b;
}

```

4. Ketik kode program di bawah ini dengan menggunakan Visual Studio, kemudian compile dan jalankan program tersebut. Jelaskan dan simpulkan hasil program.

```

#include <iostream>

using namespace std;

int main() {
    int nums[2][2] = {
        {1, 2},
        {3, 4}
    };

    cout << *(*nums) << " " << *(*nums + 1) << endl;
    cout << *(*nums + 1) << " " << *(*nums + 1) + 1 << endl;
}

```

F. Tugas Praktikum

1. Buatlah program yang menggunakan pointer untuk merujuk array 3 dimensi
2. Buatlah sebuah program yang menyelesaikan permasalahan di sekitar dengan pointer. Kemudian jelaskan hasil dari program tersebut.