

Laporan Program Praktikum 4: Pengurutan Vector

Penjelasan Kode Per 5 Baris

Baris 1-5: Include Library dan Deklarasi Fungsi

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <functional>
```

```
void printVector(const std::string& label, const std::vector<int>& vec) {
```

Fungsi: Bagian ini mengimpor library yang diperlukan untuk program. `iostream` untuk input/output, `vector` untuk struktur data vector, `algorithm` untuk fungsi sorting, dan `functional` untuk fungsi pembandingan seperti `greater`. Kemudian mendeklarasikan fungsi `printVector` yang menerima label string dan vector integer sebagai parameter untuk mencetak isi vector.

Baris 6-10: Isi Fungsi `printVector` - Bagian Cetak

```
    std::cout << label << std::endl;
    for (size_t i = 0; i < vec.size(); ++i) {
        std::cout << vec[i] << (i == vec.size() - 1 ? "" : " | ");
    }
    std::cout << std::endl;
```

Fungsi: Mencetak label yang diberikan, kemudian melakukan perulangan untuk mencetak setiap elemen vector. Setiap elemen dipisahkan dengan simbol " | ", kecuali elemen terakhir yang tidak diberi separator. Setelah semua elemen dicetak, ditambahkan baris baru.

Baris 11-15: Penutup Fungsi dan Deklarasi Array

```
    std::cout << "===== " <<
std::endl;
}
```

```
int main() {
    int data[11] = { 12, 17, 10, 5, 15, 25, 11, 7, 25, 16, 19 };
```

Fungsi: Mencetak garis pemisah untuk memperindah tampilan output, kemudian menutup fungsi `printVector`. Fungsi `main` dimulai dan mendeklarasikan array integer bernama `data` dengan 11 elemen yang berisi angka-angka acak yang akan diurutkan.

Baris 16-20: Konversi ke Vector dan Cetak Awal

```
std::vector<int> vec(data, data + sizeof(data) / sizeof(data[0]));

printVector("Isi vector awal:", vec);

std::sort(vec.begin(), vec.end());
```

Fungsi: Mengkonversi array data menjadi `std::vector` dengan menggunakan konstruktor yang menerima iterator awal dan akhir. Mencetak isi vector sebelum pengurutan menggunakan fungsi `printVector`. Kemudian mengurutkan vector secara ascending (menaik) dari nilai terkecil ke terbesar menggunakan `std::sort`.

Baris 21-25: Sorting Descending

```
printVector("Isi vector setelah di sorting:", vec);

std::sort(vec.begin(), vec.end(), std::greater<int>());

printVector("Isi vector setelah di di DESC Order:", vec);
```

Fungsi: Mencetak hasil vector setelah diurutkan secara ascending. Kemudian mengurutkan ulang vector secara descending (menurun) dari nilai terbesar ke terkecil menggunakan `std::sort` dengan parameter tambahan `std::greater<int>()` sebagai fungsi pembandingan. Mencetak hasil pengurutan descending.

Baris 26-28: Penutup Program

```
return 0;
}
```

Fungsi: Mengembalikan nilai 0 yang menandakan program berakhir dengan sukses tanpa error, kemudian menutup fungsi `main` yang merupakan akhir dari program.

Kesimpulan

Program ini merupakan implementasi sederhana untuk mendemonstrasikan pengurutan data menggunakan `std::vector` dan `std::sort` dalam C++. Program menunjukkan cara mengkonversi array menjadi vector, melakukan pengurutan ascending dan descending, serta menampilkan hasil dengan format yang rapi menggunakan fungsi bantuan `printVector`.