

# Class Person Lanjutan

Batas Waktu: 1 detik ; Batas Memori: 8 MB

## Deskripsi

Pada praktikum sebelumnya, kita sudah membuat sebuah class bernama Person yang merupakan implementasi pemrograman bagi obyek person (orang). Dari contoh tersebut dapat dipahami bahwa sebuah class memiliki dua komponen, yaitu (1) data members: nama, usia, tinggi, berat; dan (2) member functions: Person() yang merupakan default constructor, setPerson(), dsb. Secara garis besar, prototipe class Person dapat dituliskan sebagai berikut:

```
class Person {
    string nama;
    int usia;
    int tinggi;
    double berat;
public:
    Person();    // default constructor
    void setPerson(string nm, int u, int t, double b);
    string getNama();
    int getUsia();
    int getTinggi();
};
```

Semua member function dikelompokkan dalam akses public, artinya semua fungsi atau prosedur di dalamnya dapat diakses di luar class. Sedangkan semua data members tidak dikelompokkan yang berarti seluruhnya masuk ke dalam kelompok akses private, yang artinya hanya dapat diakses oleh prosedur di dalam class itu sendiri. Itulah mengapa diperlukan fungsi setPerson(), getNama(), getUsia(), dan getTinggi() agar fungsi lain dapat mengakses nama, usia, dan tinggi untuk diolah lebih lanjut. Fungsi yang diperlukan untuk mengisi nilai pada data members seperti setPerson() disebut sebagai mutator, sedangkan fungsi yang diperlukan untuk mengakses data members disebut sebagai acesor (asesor).

Satu lagi kelompok akses yaitu **protected**, yang dapat diakses oleh class itu sendiri beserta turunannya (akan dijelaskan pada bagian berikutnya).

Sebagai latihan, lengkapi member functions dari class Person tersebut untuk hal-hal berikut:

- mengakses data berat dengan fungsi asesor getBerat().
- menghitung indeks massa tubuh (IMT) menggunakan fungsi getIMT(), dimana nilai IMT adalah berat (kg) dibagi dengan kuadrat dari tinggi badan (meter).
- menentukan status gizi orang tersebut menggunakan fungsi getStatusGizi(), dimana seseorang berstatus "sangat kurus" jika  $IMT < 17.0$ , "kurus" jika  $17.0 \leq IMT < 18.5$ , "normal" jika  $18.5 \leq IMT < 25.0$ , "gemuk" jika  $25.0 \leq IMT < 28.0$ , dan "sangat gemuk" jika  $28.0 \leq IMT$ .

Selanjutnya, seperti pada latihan sebelumnya, bacalah n data nama, usia, tinggi, dan berat untuk dimasukkan ke dalam class Person. Tampilkan data tersebut sesuai ketentuan di bawah ini.

## Format Masukan

Baris pertama adalah sebuah bilangan bulat  $n$ ,  $1 \leq n \leq 100$ , yang menunjukkan banyaknya baris data yang akan dibaca. Dan  $n$  baris berikutnya berisi data nama, usia, tinggi badan, dan berat badan yang masing-masing dipisahkan oleh satu spasi. Nama orang berupa string dengan hanya terdiri atas satu kata.

## Format Keluaran

Daftar nama orang, usia, tinggi badan, berat badan (dalam dua digit di belakang tanda desimal), nilai IMT (dalam dua digit di belakang tanda desimal), dan status gizi yang dipisahkan oleh satu spasi, diikuti oleh nilai rata-rata tinggi badan (dengan dua digit di belakang tanda desimal), banyaknya orang yang tingginya di atas rata-rata, dan banyaknya orang dengan status "sangat kurus", "kurus", "normal", "gemuk" dan "sangat gemuk". Output program diakhiri dengan *newline*.

## Contoh Masukan

```
4
Dudung 20 165 60.5
Abdullah 21 160 70.2
Sempurna 19 170 80.1
Oskadon 17 155 90.0
```

## Contoh Keluaran

```
Dudung 20 165 60.50 22.22 normal
Abdullah 21 160 70.20 27.42 gemuk
Sempurna 19 170 80.10 27.72 gemuk
Oskadon 17 155 90.00 37.46 sangat gemuk
162.50
2
0 0 1 2 1
```

