



Program Studi Teknik Elektro ITB

Nama Kuliah (Kode) : Praktikum Pemecahan Masalah dengan C (EL2208)
Tahun / Semester : 2020-2021 / Genap
Modul : 4 - *Structures and Dynamic Arrays*
Hari, Tanggal Praktikum : Rabu, 17 Maret 2021

Naskah Soal Praktikum

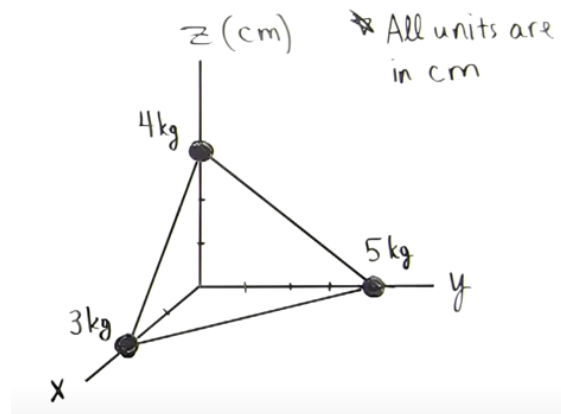
Pembuat Naskah: David Fauzi, Dismas Widyanto

Ketentuan:

1. Kerjakanlah satu dari dua soal berikut pada *template repository* yang anda peroleh ketika mengambil *assignment* di GitHub Classroom praktikum!
2. *Commit* yang dilakukan setelah sesi praktikum berakhir tidak akan dipertimbangkan dalam penilaian.
3. *Header* setiap *file* harus mengikuti format yang telah disediakan pada *file template repository*. *Header* yang tidak mengikuti format tersebut tidak akan dinilai.
4. Buku catatan laboratorium yang berisi *flowchart* dan *data flow diagram* dari solusi yang anda buat dikumpulkan ke tugas.stei.itb.ac.id paling lambat pukul 11.00 WIB satu hari kerja setelah sesi praktikum. Tulis pula alasan anda memilih mengerjakan soal yang anda kerjakan dan mengapa bukan soal yang lainnya!
5. Solusi soal pertama harus dapat dikompilasi dengan perintah `make soal-01` dan menghasilkan *file executable* dengan nama `soal-01`. Demikian pula, soal kedua harus dapat dikompilasi dengan perintah `make soal-02` dan menghasilkan *file executable* dengan nama `soal-02`.
6. Bila diperlukan, sesuaikanlah isi *Makefile* yang tersedia pada *template repository* untuk memenuhi syarat kompilasi dan *file* keluaran di atas!

Soal 1

Jono adalah anak TPB yang sedang keos belajar mata kuliah fisika dasar. Suatu hari, Jono sedang mempelajari konsep dari pusat massa (*center of mass*) dari beberapa partikel titik dengan massa tertentu pada bidang 3 dimensi. *Center of Mass* merupakan suatu posisi rerata dari semua massa yang ada di dalam suatu sistem.



Perhitungan titik *center of mass* dapat dihitung dengan formula berikut :

$$x_{cm} = \frac{\sum_{i=1}^N m_i x_i}{M} \quad y_{cm} = \frac{\sum_{i=1}^N m_i y_i}{M} \quad z_{cm} = \frac{\sum_{i=1}^N m_i z_i}{M}$$

dengan M_i merupakan massa partikel ke- i , x_i, y_i, z_i merupakan koordinat x, y, z dari partikel ke- i , dan M merupakan total massa dari seluruh partikel.

Jono lalu meminta anda untuk membuat suatu program dalam bahasa C yang mampu menerima input berupa N buah partikel dimana masing-masing partikel memiliki massa dan koordinat di bidang 3 dimensi. Program akan berhenti menginput data partikel jika pengguna memasukan angka 0 di ketika diberikan pilihan untuk lanjut atau tidak. Program lalu akan mengeluarkan output berupa titik pusat massa dari sistem yang terdiri dari N buah partikel tersebut berikut dengan kapasitas dari array dinamis yang dipakai.

Data partikel akan disimpan dalam suatu array of `Particle` dimana `Particle` merupakan tipe data bentukan yang didalamnya terdapat massa dan `Point`. `Point` juga merupakan tipe data bentukan yang

berisi 3 buah float koordinat (xyz). Namun, untuk menghemat penggunaan memori, *array of Particle* harus bersifat dinamik (*dynamic array*). Pada awalnya, kapasitas array hanya akan mampu menyimpan 2 elemen. Namun, ketika ada elemen baru yang ingin dimasukkan tetapi kapasitas *array* sudah penuh, ukuran dari *array* harus dapat di *resize* atau diperbesar lagi menjadi **dua kali lipat** kapasitas sebelumnya.

Ketentuan:

- Gunakan template kode yang diberikan di repo dan praktikan **hanya perlu mengisi** implementasi pada bagian pembacaan input di `main()` dan implementasi dari fungsi `createArray`, `pushToArray`, dan `calculateCenterMass` yang dengan deskripsi dari fungsi yang telah dijelaskan.
- **Tidak dibolehkan** untuk mengganti bagian kode yang lainnya mulai dari *library header*, tipe data yang telah dideklarasikan dan formatting output program pada fungsi utama (`main`), isi *structure*, hingga definisi dari fungsi (*return type* dan *input parameter*). Namun, praktikan diperbolehkan untuk mendeklarasikan variabel tambahan didalam fungsi dan `main()`.
- Pelajari dan baca kembali konsep *pointer*, *array*, dan *structure*. Untuk dapat mengimplementasikan *array of structures* dengan baik, pelajari cara mengalokasikan memori secara dinamis menggunakan `malloc` dan `realloc`.
- Asumsikan seluruh input valid.

Contoh eksekusi program (garis bawah menunjukkan input)

```
#1
Partikel ke-1!
Massa: 3
Koordinat x: 2
Koordinat y: 0
Koordinat z: 0
Lanjut?: 1
Partikel ke-2!
Massa: 5
Koordinat x: 0
Koordinat y: 4
Koordinat z: 0
Lanjut?: 1
Partikel ke-3!
Massa: 4
Koordinat x: 0
Koordinat y: 0
Koordinat z: 3
Lanjut?: 0

Pusat Massa:
```

Koordinat x: 0.50
Koordinat y: 1.67
Koordinat z: 1.00
Kapasitas Array: 4

#2

Partikel ke-1!
Massa: 9
Koordinat x: 24
Koordinat y: 21
Koordinat z: 18
Lanjut?: 1
Partikel ke-2!
Massa: 8
Koordinat x: 2
Koordinat y: 19
Koordinat z: 15
Lanjut?: 0

Pusat Massa:
Koordinat x: 13.65
Koordinat y: 20.06
Koordinat z: 16.59
Kapasitas Array: 2

#3

Partikel ke-1!
Massa: 13
Koordinat x: 13
Koordinat y: 17
Koordinat z: 11
Lanjut?: 1
Partikel ke-2!
Massa: 1
Koordinat x: 6
Koordinat y: 11
Koordinat z: 3
Lanjut?: 1
Partikel ke-3!
Massa: 14
Koordinat x: 12
Koordinat y: 15
Koordinat z: 13
Lanjut?: 1
Partikel ke-4!
Massa: 8
Koordinat x: 23
Koordinat y: 7

Koordinat z: 14

Lanjut?: 1

Partikel ke-5!

Massa: 12

Koordinat x: 4

Koordinat y: 8

Koordinat z: 24

Lanjut?: 0

Pusat Massa:

Koordinat x: 11.98

Koordinat y: 12.38

Koordinat z: 15.17

Kapasitas Array: 8

Soal 2

Mbak Mina beserta keluarganya ingin pergi berlibur. Mereka akan pergi menggunakan mobil yang memiliki rasio jarak dan bahan bakar sebesar **10 km/L**. Sebagai persiapan, mbak Mina ingin mensimulasikan perjalanan yang akan ditempuh nantinya agar tidak ada kendala. Mbak Mina merepresentasikan kota-kota yang dilewati sebagai titik-titik pada bidang kartesius dengan satuan kilometer (km). Simulasi yang akan dilakukan akan menerima masukan **posisi kota** yang dilalui dan **kecepatan** rata-rata dari kota sebelumnya hingga kota tersebut. Keluaran yang diperlukan oleh mbak Mina adalah **total jarak** yang ditempuh hingga sampai ke tujuan, **waktu** yang diperlukan, dan **banyak bahan bakar** yang diperlukan. Agar lebih mudah, Mbak Mina membuat file eksternal untuk input koordinat kota dan kecepatan.

Bantulah mbak Mina untuk membuat program dengan spesifikasi sebagai berikut

- Program menerima masukan berupa file eksternal dengan ekstensi .txt
- Program akan memiliki 3 buah fungsi yaitu:
 - **Distance**: untuk menghitung total jarak yang ditempuh
 - **Time**: untuk menghitung lama perjalanan
 - **Bahanbakar**: untuk menghitung banyak bahan bakar yang diperlukan

Persamaan dasar

Jarak antara 2 titik, yaitu titik **A** (a_x, a_y) dan titik **B** (b_x, b_y), dapat dihitung dengan persamaan

$$Jarak = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$$

Kecepatan

$$V = \frac{jarak}{waktu}$$

Catatan:

- Jika perlu menggunakan array, gunakan **array dinamis**.
- Wajib menggunakan **tipe data bentukan**.
- Kota yang dilalui sudah disusun secara urut
- Wajib merealisasikan ketiga fungsi di atas. Boleh menambah fungsi apabila diperlukan
- Gunakan library yang ada pada Bahasa C untuk mempermudah perhitungan

Contoh file eksternal (jalur1.txt)

```
3,0,10  
6,0,20  
6,8,50
```

File eksternal ditulis dengan format <koordinat_x>, <koordinat_y>, <kecepatan>. Dari contoh di atas artinya Mbak Mina akan melewati kota (3,0) dengan kecepatan 10 km/jam, kota (6,0) dengan kecepatan 20 km/jam, dan kota (6,8) dengan kecepatan 50 km/jam.

Contoh eksekusi program (garis bawah menandakan input)

#1

Masukkan jalur: jalur1.txt

Jarak yang ditempuh adalah: 14.00 KM

Waktu yang dibutuhkan adalah: 0 Jam 37 Menit

Bahan bakar yang dibutuhkan adalah: 1.40 Liter

#2

Masukkan jalur: jalur2.txt

Jarak yang ditempuh adalah: 76.43 KM

Waktu yang dibutuhkan adalah: 1 Jam 23 Menit

Bahan bakar yang dibutuhkan adalah: 7.64 Liter

#3

Masukkan jalur: jalur8.txt

Jarak yang ditempuh adalah: 592.94 KM

Waktu yang dibutuhkan adalah: 12 Jam 15 Menit

Bahan bakar yang dibutuhkan adalah: 59.29 Liter