

UGD Linked List 2

Tipe A

Pak Dhiaz adalah seorang ahli Astronomi lulusan UAJY yang bekerja untuk NASA. Beliau bekerja dalam bidang pendataan planet. Pak Dhiaz sudah kewalahan menyimpan data-data planet secara manual dan menginginkan sebuah program untuk menyimpan data planet yang telah ditemui dan diobservasi. Sebab itu Beliau meminta bantuan anda untuk membuat sebuah program penyimpanan data planet-planet tersebut. Berikut ketentuan untuk program yang diinginkan.

Jenis data yang akan disimpan adalah sebagai berikut:

Planet
namaPlanet: String
jarakDariBumi: int
status: String

Berikut Menu Tampilan yang diminta oleh pak Dhiaz:

```
[0]

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> |
```

1. Tampilkan Data

Agar tidak perlu membuka-buka file lagi Pak Dhiaz meminta untuk datanya ditampilkan pada Menu Utama ketika program dijalankan seperti contoh dibawah ini

```
[0]

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> |
```

Tampilan Sebelum ada data

```
[0]
[1] 200000 km - Mars => Bahaya
[2] 1250000 km - Jupiter => Mematikan

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> |
```

Tampilan sudah ada isi data

Data akan berawal dengan 0, jika masih kosong maka hanya akan ada tampilan [0] seperti ilustrasi diatas. Format penampilan data adalah jarakDariBumi km – namaPlanet - status

2. Memasukan Data

untuk memasukan data, Pak Dhiaz mengusulkan bahwa inputan dilakukan berdasarkan nomor urutan keinginannya. Jadi, sebelum menginputkan data, akan diberikan pilihan posisi inputan.

```
[0]

Data ingin dimasukan di urutan : 0

Masukan Nama Planet: Mars

Masukan Jarak Planet Dari Bumi: 200000

Masukan Status Planet[Aman|Bahaya|Mematikan]: Bahaya

Data Inserted|
```

Ketentuan inputan adalah sebagai berikut:

1. Jika data ingin dimasukkan di urutan 0, lakukan insertFirst.
 2. Jika data ingin dimasukkan di urutan terakhir (jumlah data+1), maka lakukan insertLast
 3. Jika data ingin dimasukkan di urutan yang sudah terisi data lain, lakukan insertAfter data pada urutan tersebut. (*Contoh: data 1 sudah terisi, maka lakukan insertAfter data 1 dan isi data kedua*).
 4. Nama planet tidak boleh kosong
 5. Jarak tidak boleh lebih kecil dari 1
 6. Status hanya bisa aman, bahaya, atau mematikan
- ## 3. Menghitung data
- Pak Dhiaz ingin mencatat berapa banyak data yang sudah dimasukkan oleh-Nya ke dalam program untuk kemudian dilaporkan, oleh sebab itu buat operasi menghitung jumlah data yang sudah ada di dalam linked list!

```

[0]
[1] 200000 km - Mars => Bahaya
[2] 1250000 km - Jupiter => Mematikan

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> 2
Ada 2 Data|

```

Contoh operasi CountData dengan 2 data dalam Linked List

```

[0]

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> 2
Ada 0 Data|

```

Contoh operasi CountData dengan 0 data dalam Linked List

4. Menghapus data

Tentunya seorang manusia tidak lepas dari kesalahan, karena itu, Pak Dhiaz juga mengusulkan adanya operasi untuk menghapus data sesuai dengan inputan nama planet. Penghapusan data harus juga ada konfirmasi terlebih dahulu, dan jika planet tidak ditemukan dalam program, maka program akan memberikan error handling.

```

[0]
[1] 200000 km - Mars => Bahaya
[2] 1250000 km - Jupiter => Mematikan

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> 3
Nama Planet yang ingin dihapus: salahcari

[!] Nama Planet tidak ditemukan [!]|

```

contoh hapus data jika data tidak ditemukan

```

[0]
[1] 200000 km - Mars => Bahaya
[2] 1250000 km - Jupiter => Mematikan

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> 3
Nama Planet yang ingin dihapus: Mars

[!] Yakin ingin menghapus Planet Mars? [Y/N]
>>> Y

[!] Planet Mars telah di hapus [!]|

```

contoh hapus data berhasil

5. Menampilkan Kesimpulan

Pak Dhiaz juga menyampaikan bahwa setiap akhir pekerjaannya, beliau harus mencari planet terjauh dan terdekat yang ia temukan, lalu cari selisih jaraknya beserta biaya peluncuran dengan rumus (**jarak * 15**) dan melaporkan planet terjauh ke atasannya. Oleh sebab itu, buatlah operasi pencarian planet terdekat dan terjauh untuk pak Dhiaz.

```
[0]
[1] 200000 km - Mars => Bahaya
[2] 1250000 km - Jupiter => Mematikan

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> 4
Planet dengan jarak terjauh dari bumi:
      Jupiter dengan biaya peluncuran sebesar Rp 18750000

Planet dengan jarak pendek dari bumi:
      Mars dengan biaya peluncuran sebesar Rp 3000000

Selisih Jarak: 1050000|
```

6. Menampilkan Riwayat

Pak Dhiaz lupa menyampaikan bahwa dibutuhkan sebuah log data planet yang telah dihapus setelah mengakses menu, maka buatlah operasi menampilkan data Riwayat yang hanya bisa diakses ketika ada data yang telah di delete.

```
[0]
[1] 200000 km - Mars => Bahaya

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> 5
[List][Jupiter-Mematikan-1250000]->[NULL]|
```

BONUS

Sebagai tantangan dari pak Dhiaz, kalian diminat untuk membuat urutan data menjadi terbalik (Reverse List)

7. Reverse List

Menukar List yang ada sehingga input pertama menjadi input terakhir seperti contoh di bawah

```
[0]
[1] 200000 km - Mars => Bahaya
[2] 1250000 km - Jupiter => Mematikan
[3] 120000 km - Venus => Aman
[4] 1005000 km - Merkurius => Aman
[5] 10000000 km - Pluto => Mematikan
[6] 120000 km - Saturnus => Aman

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> |
```

List Sebelum ditukar

```
[0]
[1] 120000 km - Saturnus => Aman
[2] 10000000 km - Pluto => Mematikan
[3] 1005000 km - Merkurius => Aman
[4] 120000 km - Venus => Aman
[5] 1250000 km - Jupiter => Mematikan
[6] 200000 km - Mars => Bahaya

[Menu Angkasa AtmaCorp]

[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> |
```

Setelah List di tukar

Agar memperkecil memori yang dipakai maka pak Dhiaz meminta membuat reverse list ini hanya dapat menggunakan satu Linked List dan tanpa menggunakan list lain dalam (artian lain reverse list ini harus di buat secara iteratif).

TUGAS

8. **WAJIB! Menyelesaikan UGD terlebih dahulu!**

Sebagai Langkah pengetesan akhir kalian diminta untuk membuat operasi penukaran jarak terjauh dengan jarak terpendek planet-planet yang ada dalam Linked List seperti contoh di bawah ini:

```
[0]
[1] 200000 km - Mars => Bahaya
[2] 1250000 km - Jupiter => Mematikan
[3] 120000 km - Venus => Aman
[4] 1005000 km - Merkurius => Aman
[5] 10000000 km - Pluto => Mematikan
[6] 120000 km - Saturnus => Aman
```

[Menu Angkasa AtmaCorp]

```
[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
```

```
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> |
```

Sebelum Adanya Pertukaran data

Maka hanya data jarak terdekat dengan data jarak terjauh yang ditukar (pada kasus ini adalah data ke 3 dengan data ke 5).

```
[0]
[1] 200000 km - Mars => Bahaya
[2] 1250000 km - Jupiter => Mematikan
[3] 10000000 km - Pluto => Mematikan
[4] 1005000 km - Merkurius => Aman
[5] 120000 km - Venus => Aman
[6] 120000 km - Saturnus => Aman
```

[Menu Angkasa AtmaCorp]

```
[1]. Input Planet
[2]. Hitung Data
[3]. Hapus Planet
[4]. Kesimpulan
[5]. History Planet
```

```
-----
[6]. Reverse List [Bonus]
[7]. Swap Dari Jarak [Tugas]
[0]. Keluar
>>> |
```

Setelah Adanya penukaran data

LAPORAN

Laporan berisikan penjelasan dari Code UGD dan Tugas sesuai dengan ketentuan minimal 2 halaman dan maksimal 5 halaman termasuk *cover*. Bonus tidak perlu dimasukkan ke dalam laporan.

KETENTUAN

Format Pengumpulan:

- UGD : UGD7_X_YYYYY
- TGS : TGS7_X_YYYYY
- BONUS : BNS7_X_YYYYY
- LAPORAN : LAP7_X_YYYYY

Keterangan:

X = Huruf Kelas

Y = 5 Digit NPM

- UGD, TGS atau Bonus harus terkompresi menggunakan format **.zip!** dan Laporan harus dikumpulkan menggunakan **.pdf!**
- **DILARANG!** SAAT UGD, BERTANYA TENTANG TGS / BONUS KE ASISTEN. KECUALI KEJELASAN SOAL.
- **SEGALA TINDAK KECURANGAN TIDAK AKAN DITOLERANSI!!! (CHATGPT, MEMBAWA MODUL KELAS LAIN, DLL)**
- Untuk mengambil Bonus tidak perlu di cek oleh asisten, langsung kumpulkan saja ke uploader Bonus dan panggil asisten untuk memastikan bahwa file telah dikumpulkan dan pastikan bukan draft