

Kuis Pemrograman 1
Algoritma Sorting, Linked List, Stack, dan Queue

Struktur Data & Algoritma
IKI10400
Semester Genap 2013/2014

Fakultas Ilmu Komputer
Universitas Indonesia

Batas waktu pengumpulan kode sumber:
Senin, 28 April 2014 pukul 18.00 Waktu Scele

Kode sumber yang dinilai hanya yang **dikumpulkan melalui Auto Grader Scele**. Kode sumber yang dikumpulkan melalui mekanisme selain itu akan **diabaikan** dan **dianggap tidak mengumpulkan**.

Peringatan:

Jangan mengumpulkan pekerjaan beberapa menit menjelang batas waktu pengumpulan karena ada kemungkinan pengumpulan gagal dilakukan atau koneksi internet terputus!

Dilarang berkomunikasi dengan peserta lain, jika ada pertanyaan langsung ditanyakan ke Asisten yang mengawas. Segala tindak kecurangan akan diberikan sanksi sesuai dengan peraturan yang sudah dibuat di awal kuliah.

Kuis Pemrograman 1A

Balanced Symbol

Nama berkas kode sumber : SDA13141KPA.java
Batas waktu eksekusi program : 3 detik / kasus uji
Batas memori program : 16 MiB / kasus uji
Nilai : 35

Pada perkuliahan, kalian sudah mempelajari salah satu pengaplikasian dari Stack yaitu untuk memeriksa pasangan tanda kurung (Balanced Symbol). Misalnya: { [()] } merupakan pasangan tanda kurung yang valid, tetapi { ([)] } tidak valid. Implementasikan fungsi **isValid** yang menerima masukan array of char untuk melakukan pengecekan suatu masukan tanda kurung.

Anda **diwajibkan** menggunakan template yang sudah ada di Scele dan melengkapi method **isValid()** pada kelas SDA13141KPA saja. Kumpulkan kode Anda pada slot di Scele untuk menguji implementasi Anda. **Dilarang** mengubah implementasi lain pada kode template. **Dilarang** menggunakan struktur data dan method dari Java Collections API. Asisten akan memeriksa kode sumber Anda untuk penilaian White Box.

Format Masukan

Pada kuis ini, Anda tidak perlu membuat implementasi masukan dan keluaran.

Baris pertama masukan merupakan angka N ($1 \leq N \leq 15$) yaitu jumlah pasangan tanda kurung yang akan dicek. N baris selanjutnya merupakan baris kosong atau pasangan tanda kurung yang hanya terdiri dari simbol {, }, (,), [, dan].

Format Keluaran

Keluaran sebanyak N baris. Untuk setiap baris ke- $[X]$, cetak "Case $\#[X]$: Valid" (tanpa tanda kutip) jika pasangan tanda kurung pada baris itu valid dan cetak "Case $\#[X]$: Tidak Valid" (tanpa tanda kutip) jika pasangan tanda kurung pada baris itu tidak valid.

Contoh Masukan

```
5
()
{}
[(
{{{{{}}}}}
[]{}

```

Contoh Keluaran

Case #1: Valid

Case #2: Tidak Valid

Case #3: Tidak Valid

Case #4: Valid

Case #5: Valid

Kuis Pemrograman 1B

Linked List Intersection

Nama berkas kode sumber : SDA13141KPB.java
Batas waktu eksekusi program : 3 detik / kasus uji
Batas memori program : 16 MiB / kasus uji
Nilai : 35

Diberikan dua buah Linked List A dan B yang elemennya unik (himpunan), kembalikan Linked List C sedemikian sehingga $C = A \cap B$.

$C = A \cap B$, berarti elemen dari list C merupakan elemen-elemen yang ada di list A dan B. Elemen di list C terurut berdasarkan kemunculannya di list A. Contoh: Misalkan List A = [-6, **3**, 2, **1**, **9**] dan List B = [**1**, 7, -1, **3**, 6, **9**, 0], maka List C = [3, 1, 9].

Anda **diwajibkan** menggunakan template yang sudah ada di Scele dan melengkapi method **intersection()** pada kelas LinkedList saja. **Dilarang** mengubah implementasi lain pada kode template. **Dilarang** menggunakan struktur data dan method dari Java Collections API. Asisten akan memeriksa kode sumber Anda untuk penilaian White Box.

Format Masukan

Pada kuis ini, Anda tidak perlu membuat implementasi masukan dan keluaran.

Baris pertama masukan merupakan elemen dari list A dipisahkan dengan spasi. Baris kedua masukan merupakan elemen dari list B dipisahkan dengan spasi. Kedua list minimal terdiri dari satu elemen.

Format Keluaran

Elemen dari list C yang merupakan elemen-elemen yang ada di list A dan B terurut berdasarkan kemunculannya di list A dipisahkan dengan spasi. Jika $A \cap B = \emptyset$, cetak baris kosong.

Contoh Masukan

```
-6 3 2 1 9
1 7 -1 3 6 9 0
```

Contoh Keluaran

```
3 1 9
```

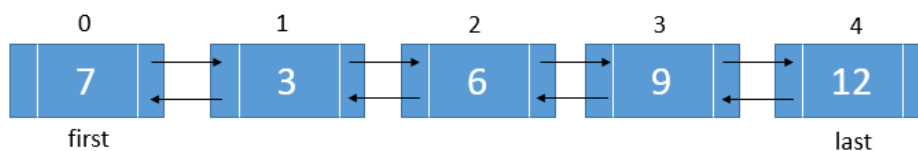
Kuis Pemrograman 1C

Delete Range Doubly Linked List

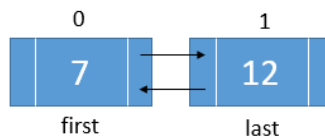
Nama berkas kode sumber : SDA13141KPC.java
Batas waktu eksekusi program : 3 detik / kasus uji
Batas memori program : 16 MiB / kasus uji
Nilai : 30

Pada materi perkuliahan, kita sudah mempelajari variasi dari Linked List yaitu Doubly Linked List, dimana list node menyimpan referensi node sebelum dan sesudahnya.

Implementasikan method **deleteRange(int x, int y)** untuk menghapus elemen linked list dari indeks x sampai indeks y. Contoh: Terdapat List A sebagai berikut,



Setelah dilakukan pemanggilan **deleteRange(1, 3)** hasilnya menjadi:



Anda **diwajibkan** menggunakan template yang sudah ada di Scele dan melengkapi method **deleteRange(int x, int y)** pada kelas DoublyLinkedList saja. **Dilarang** mengubah implementasi lain pada kode template. **Dilarang** menggunakan struktur data dan method dari Java Collections API. Asisten akan memeriksa kode sumber Anda untuk penilaian White Box.

Format Masukan

Pada kuis ini, Anda tidak perlu membuat implementasi masukan dan keluaran.

Baris pertama masukan merupakan angka N ($1 \leq N \leq 5000$) yaitu jumlah perintah yang akan dilakukan. N baris selanjutnya merupakan perintah dan terbagi menjadi tiga yaitu:

1. "add [T]" dengan T merupakan bilangan integer untuk memasukkan nilai T ke awal list,
2. "deleteRange [X] [Y]" untuk menghapus list dari X – Y dengan nilai X dan Y bilangan integer ≥ 0 .

Format Keluaran

Untuk setiap perintah deleteRange cetak seluruh elemen dari list.

Perhatikan urutan case-case berikut:

- Jika sebelum delete list kosong, tidak perlu mencetak apa-apa. Namun, jika sesudah delete list kosong, cetak baris kosong.
- Saat $X > Y$, maka masukan tidak valid dan cetak "**x > y**" (tanpa tanda kutip).
- Saat X dan/atau Y berada di luar indeks list maka masukan tidak valid dan cetak "**x/y out of bounds**" (tanpa tanda kutip).

Jika masukan tidak valid, tidak ada perubahan pada list.

Contoh Masukan

```
10
add 12
add 9
add 6
add 3
add 7
deleteRange 2 7
deleteRange 3 1
deleteRange 2 2
add 5
deleteRange 0 3
```

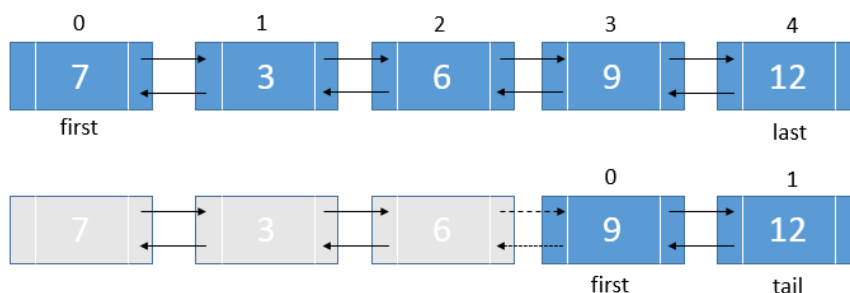
Contoh Keluaran

```
x/y out of bounds
x > y
7 3 9 12
12
```

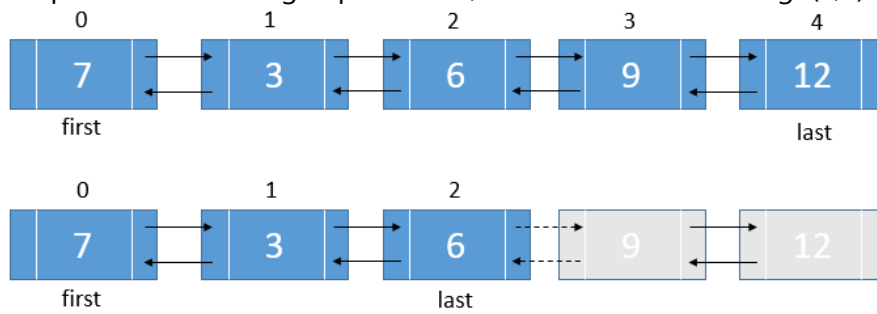
Hint

Perhatikan kasus-kasus berikut:

- Linked List kosong.
- Jika posisi X sama dengan posisi First, lihat contoh deleteRange(0,2) berikut:



3. Jika posisi Y sama dengan posisi Last, lihat contoh deleteRange(3,4) berikut:



4. Perhatikan juga saat menghapus dari first sampai last.