

Panduan untuk mengerjakan soal Praktikum_02 - snake

Pada umumnya, kesulitan terjadi untuk menjawab pertanyaan “Bagaimana caranya memindahkan tubuh snake, terutama ketika harus berbelok?”

Perhatikan clue berikut ini:

	0	1	2	3	4	5	6	7	8	9
0
1
2
3	X	.	.	.
4
5	o	o	o	.	.
6
7
8
9

Pada gambar di atas, head (berwarna merah) dari snake berada pada koordinat (5,5) dan makanan pada koordinat (6,3). Dengan demikian, snake seharusnya bergerak ke atas sebanyak dua kali, dan ke kanan sebanyak satu kali untuk mencapai makanan tersebut.

Gunakanlah LinkedList untuk “membentuk” badan snake tersebut (gunakan LinkedList dari Java Collection). Secara lebih spesifik, LinkedList of Point (gunakan Point dari java.awt). Jadi, snake yang Anda miliki adalah LinkedList yang berisi point-point yang merepresentasikan koordinat dari setiap bagian snake tersebut. Pada contoh di atas, LinkedList-nya berisi: (5,5), (6,5), (7,5)

	0	1	2	3	4	5	6	7	8	9
0
1
2
3	X	.	.	.
4	o
5	o	o	.	.	.
6
7
8
9

- **Contoh langkah snake tanpa menemukan makanan**

Gambar di atas adalah posisi snake ketika snake berjalan satu kali ke atas. **Perhatikan bahwa Anda tidak perlu menggeser seluruh tubuh snake.**

Ketika snake bergerak ke atas, yang perlu Anda lakukan adalah meng-insert atau **add Point baru**, yaitu (5,4) **ke dalam LinkedList** yang Anda (gunakan method **addFirst()** dari LinkedList) DAN hapus Point terakhir dari snake Anda (7,5) (gunakan method **removeLast()** dari LinkedList). Dengan demikian, Anda tidak perlu menggeser seluruh Point yang membentuk snake Anda. Point yang membentuk snake Anda saat ini adalah (5,4),(5,5),(6,5)

Jika Anda melangkahkan snake Anda satu kali lagi ke atas, maka Anda perlu menginput Point (5,3) dan menghapus Point terakhir, yaitu (6,5)

	0	1	2	3	4	5	6	7	8	9
0
1
2
3	o	x	.	.	.
4	o
5	o
6
7
8
9

Berikut adalah contoh code untuk pergerakan ketika snake tidak memakan makanan yang tersedia:

```
public void moveNotEat(Point newPoint){
    snake.addFirst(newPoint);
    snake.removeLast();
}
```

- **Contoh langkah snake ketika menemukan makanan**

Ketika snake mendapatkan makanan, yang perlu Anda lakukan **hanya menambahkan Point baru** ke LinkedList snake Anda. Anda tidak perlu menghapus Point terakhir dari snake Anda, karena memang tubuh snake bertambah panjang satu Point.

	0	1	2	3	4	5	6	7	8	9
0
1
2
3	o	o	.	.	.
4	o
5	o
6	x	.	.	.
7
8
9

Pada contoh di atas, Anda hanya perlu menambahkan Point (6,3) ke dalam LinkedList Anda.

Contoh source code untuk kasus ini:

```
public void moveEat(Point newPoint){
    snake.addFirst(newPoint);
}
```

- **Mengetahui posisi head dari snake**

Untuk memeriksa apakah snake Anda menabrak dinding atau menemukan makanan, tentunya Anda perlu mengetahui isi dari head snake Anda, gunakan method `element()` dari LinkedList.

Silakan pelajari lebih jauh mengenai LinkedList dari javadoc yang tersedia.

Selanjutnya, silakan Anda berkreasi. Anda boleh mengerjakan tugas ini dengan cara lain (tidak menggunakan LinkedList) ☺.