

Dasar-Dasar Pemrograman 2

Lab 04

Multidimensional Arrays



FAKULTAS
ILMU
KOMPUTER

Arrays of Arrays

Sebelumnya kalian telah mempelajari bahwa Arrays dapat menyimpan *datatype* maupun *object* sebagai elemennya. Nah, satu hal yang perlu kalian ketahui juga adalah Arrays dapat bersifat multidimensional, yang berarti tiap elemen Arrays tersebut merupakan sebuah Arrays juga. Dengan kata lain, Arrays juga dapat menyimpan sebuah Arrays lainnya. Keren, bukan?!

Inisiasi Multidimensional Arrays

Hampir sama dengan inisiasi Arrays, untuk menginisiasi Multidimensional Arrays, kalian dapat menggunakan format berikut:

```
tipeData[dim1][dim2]...[dimN] namaArrays = new  
tipeData[panjangArrays1][panjangArrays2]...[panjangArraysN];
```

Atau

```
tipeData[dim1][dim2]...[dimN] namaArrays =  
{<Anggota Arrays1>},  
{<Anggota Arrays2>},  
...,  
{<Anggota ArraysN>}};
```

Agar lebih mudah, kita dapat melihat Multidimensional Arrays sebagai suatu matriks, sebagai berikut:

```
AnggotaArrays1
```

AnggotaArrays2
...
AnggotaArraysN

Sebagai contoh, misalkan kalian ingin membuat sebuah Arrays 2 dimensi bernama 'kelompokAsdos' yang merupakan gabungan dari 3 kelompok asdos yang sebelumnya telah terbentuk. Kalian dapat mentranslasikan kasus ini ke dalam Multidimensional Arrays, dengan cara:

```
String[][] kelompokAsdos = {{ "Alvin", "Lika", "Ryaas"}, {"Lita", "Litha", "Naufal"}, {"Dek", "De", "Pe"}};
```

Accessing Multidimensional Arrays

Untuk mengakses elemen yang ada dalam Multidimensional Arrays, kalian dapat menggunakan indeks layaknya Arrays pada umumnya. Sebagai contoh, misalkan kalian memiliki Arrays 2 dimensi yang berisi integer, sebagai berikut:

```
int[][] matrix = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};
```

Kita dapat melihat 2-D Arrays itu seperti ini

	Column 0	Column 1	Column 2
Row 0	1 (matrix[0][0])	2 (matrix[0][1])	3 (matrix[0][2])
Row 1	4 (matrix[1][0])	5 (matrix[1][1])	6 (matrix[1][2])
Row 2	7 (matrix[2][0])	8 (matrix[2][1])	9 (matrix[2][2])

Sebagai contoh, jika kalian ingin mengakses setiap indeks pada Arrays 'matrix' dan mengeluarkannya sebagai output satu-persatu, maka kalian dapat melakukannya dengan

```
int[][] matrix = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};

for(int i=0; i<matrix.length; i++){
    for(int j=0; j<matrix[i].length; j++){
        System.out.println(matrix[i][j]);
    }
}
```

Kalian juga dapat melakukan berbagai manipulasi terhadap Arrays, seperti melakukan *swap* untuk spesifik elemen pada Arrays. Perhatikan contoh potongan kode berikut

```
int[][] matrix = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};

int[] temp = new int[]; //menyimpan elemen sementara

temp[0] = matrix[0][2];
matrix[0][2] = matrix[2][0]; //melakukan swap
matrix[2][0] = temp[0];
```

perubahan:

```
matrix = {{1, 2, 7}, {4, 5, 6}, {3, 8, 9}}
```

Selamat mengerjakan Lab 04!

Soal Lab 04

Nona Manis Putarlah ke Kiri~



Angewomon berhasil mendapat keuntungan besar dari bisnis jasa ekspedisinya. Menikmati kekayaan barunya, Angewomon menghabiskan waktu dengan menonton Dig*mon Adv*nture. Tetapi karena keasyikan, Angewomon tidak sadar seluruh hartanya telah dicuri!

Saat memeriksa tempat penyimpanan hartanya, Angewomon menemukan secarik kertas yang secara iseng ditinggalkan oleh si pencuri. Tertulis di sana, “HAHAHAHAHAHA! Makanya punya duit bagi-bagi. Di masa pandemi begini—” Angewomon memutuskan untuk berhenti membaca. Ini jelas-jelas tulisan si Etemon.

Angewomon membalik kertas itu. Ternyata di baliknya ada sebuah matriks persegi berisi angka-angka. Angewomon menyadari kalau angka-angka itu melambangkan hartanya yang disembunyikan di lokasi tertentu. Etemon pasti meninggalkan peta ini di sini supaya Angewomon harus keluar rumah dan mengambil hartanya sendiri.

Tetapi, mengenal Etemon, ini tidak akan semudah itu. Angewomon memutar-mutar peta, tapi tidak yakin bagaimana orientasi yang benar. Angewomon meminta bantuanmu untuk memecahkan teka-teki peta Etemon.

Bantulah Angewomon mengambil hartanya kembali!

Requirements

Program kamu memproses input berupa array **2 dimensi berukuran $N \times N$** , serta **urutan arah rotasi** yang harus diterapkan. Output yang diharapkan adalah array 2 dimensi yang telah dirotasi sesuai urutan.


Input:

- Masukan pertama adalah sebuah bilangan **N** sebagai ukuran matriks.
- **N** baris berikutnya adalah elemen matriks, setiap barisnya berisi **N integer** yang dipisahkan oleh spasi
- Masukan terakhir berupa String yang terdiri dari huruf L (*Left*) dan R (*Right*), setiap hurufnya menunjukkan arah rotasi yang harus diterapkan. (Lihat contoh input/output)

Rotasi ke Kanan (R)

Contoh rotasi matriks 3x3 ke kanan satu kali:


1	2	3
4	5	6
7	8	9



7	4	1
8	5	2
9	6	3

Contoh rotasi matriks 4x4 ke kanan satu kali:

0	12	4	11
55	20	120	2000
47	1	22	15
10	12	2	1



10	47	55	0
12	1	20	12
2	22	120	4
1	15	2000	11

Rotasi ke Kiri (L)

Contoh rotasi matriks 3x3 ke kiri satu kali:

1	2	3
4	5	6
7	8	9

→

3	6	9
2	5	8
1	4	7

Contoh rotasi matriks 4x4 ke kiri satu kali:

0	12	4	11
55	20	120	2000
47	1	22	15
10	12	2	1

→

11	2000	15	1
4	120	22	2
12	20	1	12
0	55	47	10

Format Input

Keterangan: Teks berwarna **biru** adalah input yang dimasukkan oleh *user*.

Contoh masukan 1

Ukuran kolom dan baris: **3**
Peta matrix:
1 2 3
4 5 6
7 8 9
Putaran: **RR**

Contoh keluaran 1

Output:
9 8 7
6 5 4
3 2 1

Penjelasan Contoh 1

- Menjalankan perintah R pertama, akan memutar matrix input menjadi:

7	4	1
8	5	2
9	6	3

- Menjalankan perintah R kedua, akan memutar matrix input menjadi:

9	8	7
6	5	4
3	2	1

Contoh masukan 2

Ukuran kolom dan baris: 4

Peta matrix:

1 2 3 4

5 6 7 8

1 2 3 4

5 6 7 8

Putaran: RLL

Contoh keluaran 2

Output:

4 8 4 8

3 7 3 7

2 6 2 6

1 5 1 5

Penjelasan Contoh 2

- Menjalankan perintah R, akan memutar matrix input menjadi:

5	1	5	1
6	2	6	2
7	3	7	3
8	4	8	4

- Menjalankan perintah L Pertama, akan memutar matrix input menjadi:

1	2	3	4
5	6	7	8
1	2	3	4
5	6	7	8

- Menjalankan perintah L Kedua, akan memutar matrix input menjadi:

4	8	4	8
3	7	3	7
2	6	2	6
1	5	1	5

Batasan & Ketentuan

- Bilangan $N > 1$.
- Ukuran matriks dipastikan persegi.
- Matriks dipastikan berisi integer nonnegatif.
- Urutan rotasi dipastikan huruf kapital dan hanya berisi huruf L atau R.
- Wajib menggunakan struktur data Array. **Penggunaan ArrayList akan membuat pekerjaanmu tidak dinilai.**
- Gunakan template untuk mempermudah pekerjaanmu.

Komponen Penilaian

- **50%** Kebenaran kode.
- **40%** Metode manipulasi array yang benar.
- **10%** Dokumentasi dan kerapian kode

Kumpulkan berkas .java yang telah di-zip dengan format penamaan seperti berikut.

Lab04_[Kelas]_[KodeAsdos]_[NPM]_[NamaLengkap].zip

Contoh:

Lab04_A_LN_1234567890_DekDepe.zip