

Dasar-Dasar Pemrograman 2

Tutorial 3 Kelas B, D, & E

Array

Kamis, 28 Feb 2019 - 16:00 WIB



FAKULTAS
ILMU
KOMPUTER

Array

Array adalah struktur data paling sederhana yang disediakan oleh bahasa pemrograman Java. Secara informal, array adalah sebuah kumpulan N buah data dengan tipe data yang sama. Selanjutnya, dapat dikatakan bahwa array memiliki panjang N. Sederhana, bukan? Array dapat menyimpan objek-objek dengan tipe data primitive atau tipe data reference/object.

Array bersifat *fixed-size* yang berarti panjang sebuah array tidak dapat diubah setelah array tersebut diinisialisasi. Sebuah array dapat diidentifikasi dengan simbol “[]”. Elemen-elemen pada array dapat diakses berdasarkan index elemen tersebut. Sebagai contoh, `myArray[i]` akan mengembalikan elemen dengan index `i` pada array `myArray` (`i < N`), karena index suatu array dimulai dari 0.

Array juga dapat bersifat *multidimensional*, yang berarti tiap elemen array tersebut merupakan sebuah array juga.

Untuk mengetahui lebih lanjut tentang array bisa lihat di

<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html>

Contoh Penggunaan Array

Coba tebak output dari potongan program dibawah ini sebagai latihan (bukan Tutorial Lab) :

Soal 1

```
int[] numbers = new int[10]; // menginisiasi array of int dengan panjang 10
System.out.println(numbers[5]); // menampilkan ... , karena ...
```

Soal 2

```
String[] strList = new String[5]; // ...  
System.out.println(strList[0]); // menampilkan ... , karena ...
```

Soal 3

```
int[] primes = {2,3,5,7,11}; //menginisiasi array of int dengan initial values  
System.out.println(primes[2]); // menampilkan ...
```

Soal 4

```
String[] animals = {"cat","bird","tiger"};  
System.out.println(animals[1]); // menampilkan ...  
animals[0] = "fish"; // mengganti elemen index ke-0  
System.out.println(animals[1]); // menampilkan ...
```

Soal 5

```
String str = "a b c d";  
String[] splitstr = str.split(" "); // menginisiasi array of String dari method split  
System.out.println(splitstr[3]); // menampilkan ...  
splitstr[3] = "new";
```

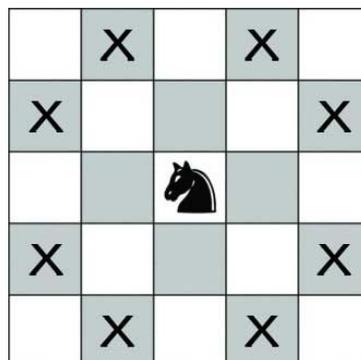
Kuda Kuda di Catur



Image source: sparkchess.com

“Terima kasih kak sudah membantuku mengerjakan tugas minggu lalu, kakak ada waktu luang? mau nonton final catur CGT bersamaku?”. Tentu saja kamu sangat senang dan langsung menerima permintaan Dek Depi untuk nonton bersama. Saat permainan dimulai, tiba-tiba Dek Depi bertanya kepada kamu, “Kak, kuda itu jalannya gimana sih? Sekarang bisa makan yang mana aja ya?”

Sebenarnya bisa saja kamu memberitahu Dek Depi secara langsung, namun karena kamu ingin terlihat pintar di depan Dek Depi, kamu merumuskan masalah tersebut berdasarkan gambar berikut dan membuatnya menjadi sebuah program.



Apa yang Harus Dilakukan?

Kamu diminta untuk **memproses input** yang berupa array 2 dimensi berisi state catur $N \times N$ dari tiap petak. Pada akhir program, kamu harus menghasilkan output jumlah bidak yang dapat dimakan oleh kuda berdasarkan input. Misal kuda berada di posisi (i, j) pada matriks (indeks dalam array), maka kuda akan dapat memakan maksimal 8 bidak, salah satu posisi yang dapat dimakan oleh kuda adalah $(i-2, j+1)$ (**cara kuda untuk makan tidak hanya ini!**) untuk sisanya kembali lagi ke peraturan di catur tentang pergerakan kuda. **Ingat!** representasi ini menggunakan array dan indeks array terbatas.

Input

Masukan pertama adalah sebuah bilangan bulat positif N , yang menyatakan ukuran papan catur. Selanjutnya, diberikan matriks berukuran $N \times N$ yang menggambarkan lokasi bidak pada papan catur. Bidak catur diwakili oleh karakter berikut: **P** (Pion), **H** (Kuda), dan **.** (Kosong) yang dipisahkan spasi antar kolom.

Dijamin kuda hanya ada satu dalam satu matriks.

TIPS, Cara menerima input Integer:

```
// input String!
Scanner input = new Scanner(System.in);
String iniStr = input.nextLine();

// input Integer!
int iniInt = Integer.parseInt(input.nextLine()); //wah mejik ya
```

Kenapa tidak menggunakan `input.nextInt()` ? Jika menggunakan `input.nextInt()`, java hanya akan membaca integernya saja tapi tidak membaca whitespace setelah integer tersebut, sehingga whitespace-nya akan tertinggal di *buffer*.

Referensi bacaan :

http://pages.cs.wisc.edu/~deppeler/cs302/resources/handouts/handout_scanner.html

Output

Output jumlah bidak yang dapat dimakan oleh kuda.

Contoh Input/Output 1:

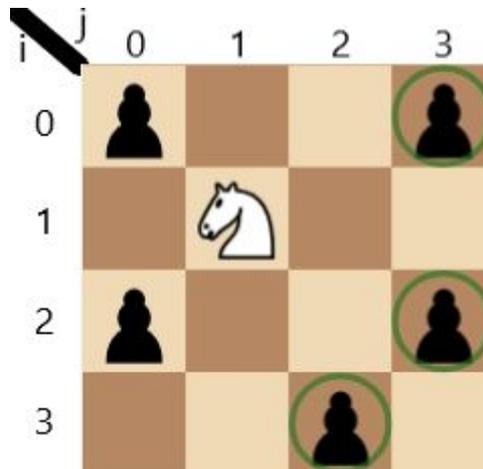
```

Masukkan ukuran catur:
4
Masukkan papan catur:
P . . P
. H . .
P . . P
. . P .
Banyaknya bidak catur yang dapat dimakan kuda:
3

```

Notes : karakter bercetak *Merah* merupakan input dari user, *Hijau* merupakan output dari program.

Penjelasan Contoh 1 :



Orientasi posisi (i,j)

Keterangan : Kuda berada pada posisi (1, 1), sehingga bidak yang dapat dimakan kuda berjumlah 3, yaitu bidak yang berada di posisi (0, 3), (2, 3), dan (3, 2). Lingkaran hijau adalah bidak yang dapat dimakan oleh kuda.

Contoh Input/Output 2:

Feel free to use, reuse, and share this work: the more we share, the more we have!



Masukkan ukuran catur:

6

Masukkan papan catur:

```
. P . P . .
. . . . P .
P . H . . .
. . . . . .
. P . P . P
. . . . P .
```

Banyaknya bidak catur yang dapat dimakan kuda:

5

Penjelasan Contoh 2 :



Orientasi posisi (i,j)

Keterangan : Kuda berada pada posisi (2, 2), sehingga bidak yang dapat dimakan kuda berjumlah 5, yaitu bidak yang berada di posisi (4, 1), (4, 3), (1, 4), (0, 3) dan (0, 1). Lingkaran hijau adalah bidak yang dapat dimakan oleh kuda.

Contoh Input/Output 3:

Masukkan ukuran catur:

3

Masukkan papan catur:

P . P

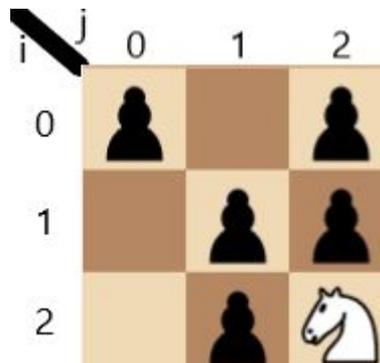
. P P

. P H

Banyaknya bidak catur yang dapat dimakan kuda:

0

Penjelasan Contoh 3 :



Orientasi posisi (i,j)

Keterangan : Kuda berada pada posisi (2, 2). Kuda tidak dapat memakan bidak manapun sehingga jumlah bidak yang dapat dimakan kuda adalah 0.

Komponen Penilaian :

Komponen	Penjelasan	Bobot
Input Output	Interaksi program sesuai dengan permintaan soal	20 %
Implementasi 2D Array	Menggunakan Array dua dimensi untuk menyimpan bidak catur.	30 %
Struktur Data	Menggunakan struktur data yang tepat untuk mengetahui letak bidak mana yang dapat dimakan oleh kuda	15%
Ketepatan	Ketepatan program sesuai permintaan soal	25 %
Kerapian	Penulisan kode mengikuti konvensi penulisan yang baik dan benar (nama variabel, styling, indentation, penyusunan, dll)	10 %

Deadline :

Kamis, 28 Februari 2019

Pukul **17:20 WIB**

Feel free to use, reuse, and share this work: the more we share, the more we have!



Format Pengumpulan :

Kumpulkan di slot pengumpulan yang telah disediakan di SCellE dengan format :

[Kode Asdos]_[Nama]_[Kelas]_[NPM]_Lab[X].zip yang berisi file **Kuda.java** Anda

Contoh : AFA_ArdhoFauzanJereTio_E_Lab3.zip yang berisi file Kuda.java di dalamnya

Acknowledged Lecturers :

-