

## Bab I Pengenalan Bahasa

### Identitas

#### Kajian

Pengenalan Java

#### Topik

1. Review Java : array dan object

#### Referensi

1. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms Third Edition. Cambridge: The MIT Press.
2. Sedgewick, R. (2002). Algorithm in Java: parts 1-4 Third Edition. Boston: Pearson Education, Inc.
3. P. Deitel and H. Deitel, Java How To Program 9th Edition, Prentice Hall, 2011.
4. Poo, Danny, Derek Kiong, and Swarnalatha Ashok. Object-Oriented Programming and Java, 2nd Edition. Springer, 2010.
5. Wu, C. Thomas. An Introduction to Object-Oriented Programming with Java. McGraw-Hill, 2009.

#### Kompetensi Utama

1. Mahasiswa mampu menggunakan bahasa java untuk membangun aplikasi sederhana

#### Lama Kegiatan Kajian

1. Pertemuan Terbimbing : 2 x 500 menit
2. Kegiatan Mandiri : 1 x 70 menit

#### Parameter Penilaian

1. Jurnal 70%
2. Tugas Akhir 30%

#### Pengumpulan

Kumpulkan jawaban dari Jurnal tembimbing dan jurnal mandiri pada LMS yang telah disediakan



## Jurnal terbimbing

1. Buatlah sebuah aplikasi yang dapat digunakan untuk melakukan pendataan deret angka dengan memanfaatkan Array. Deret tersebut akan terurut secara ascending. Berikut kode tersebut.

```
import java.util.Scanner;
class KelolaArrayInteger{
    int n=10;
    int[] arrOfInt = new int[n];
    Scanner inp = new Scanner(System.in);

    public static void main(String[] args){
        KelolaArrayInteger kai = new KelolaArrayInteger();
        kai.inputAndSort();
        kai.traversal();
    } //eomain

    void inputAndSort (){
        int val = 0;
        int temp = -999;
        for (int a=0; a<n; a++){
            System.out.print("Val : ");
            val = inp.nextInt();
            if (a>0){
                for(int b=0; b<=a; b++){
                    if (val < arrOfInt[b]){
                        temp = arrOfInt[b];
                        arrOfInt[b] = val;
                        val = temp;
                    }
                }
                arrOfInt[a] = val;
            }
        } //eorunthis
        void traversal() {
            for (int a=0; a<n; a++){
                System.out.print(arrOfInt[a]+" ");
            }
        } //eotraversal
    } //eof
```



2. Modifikasi kode program diatas, sehingga data dalam array akan terurut secara Descending. Ubah dalam method `inputAndSort()` bukan didalam method traversal. Dilarang juga memanfaatkan fungsi sort dari kelas Array.
3. Di game para Programmer Legend, setiap programmer harus menghitung kekuatan serangan mereka berdasarkan atribut yang dimiliki. Terdapat tiga atribut penting yaitu `logicalPower`, `bugTracingPower`, dan `endurancePower`. Setiap programmer harus menghitung total kekuatan mereka berdasarkan rumus:

Total Kekuatan =  $(2 \times \text{logicalPower}) + (3 \times \text{bugTracerPower}) - (0.5 \times \text{endurancePower})$

Buatlah program dengan subprogram (method) untuk menghitung total kekuatan programmer tersebut. Program akan menerima beberapa masukan berupa jumlah untuk programmer, kemudian menghitung dan mencetak total kekuatan setiap programmer menggunakan subprogram.

Masukan

1. Input pertama adalah bilangan bulat N, jumlah petualang.
2. Untuk setiap petualang, input berupa tiga bilangan bulat yang mewakili serangan fisik, serangan sihir, dan pertahanan.

Keluaran berupa total kekuatan untuk masing-masing petualang.

Input :

2

10 30 40

15 7 8

Output:

Total coding power Programmer 1 : 70

Total coding power Programmer 2 : 47

Hint : Gunakan method untuk melakukan proses input, dan perhitungan

