

Started on	Wednesday, 8 February 2023, 11:43 PM
State	Finished
Completed on	Wednesday, 8 February 2023, 11:59 PM
Time taken	16 mins 3 secs
Grade	600.00 out of 600.00 (100%)

Question **1**
Correct
Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Inheritance - Animal

Anda diminta untuk membuat kelas-kelas animal untuk memodelkan hewan di suatu kerajaan hewan. Hewan dimodelkan sebagai kelas **Animal** dengan berbagai jenis hewan merupakan turunan dari kelas **Animal**. Saat ini, kerajaan hewan hanya memiliki 1 jenis hewan yang merupakan raja para hewan yaitu **Gajah** yang dimodelkan sebagai kelas turunan **Gajah** dengan nama yang sama.

Pada soal ini, anda diminta untuk mengimplementasi kelas **Animal** yang merupakan sebuah *abstract class* yang memiliki 2 buah atribut:

- 1. **numberOfLegs**, menyimpan banyak kaki dari hewan dalam tipe data **int**
- 2. **children**, menyimpan banyak anak yang dimiliki hewan tersebut dengan tipe data **long**

dengan kelas **Animal** ini juga memiliki beberapa metode yang memiliki peran berikut:

- 1. **getNumberOfLegs**, getter dari atribut **numberOfLegs**
- 2. **getNumberOfChildren**, getter dari atribut **children**
- 3. **increaseChild**, meningkatkan nilai atribut **children** dengan nilai argumen **inc**
- 4. **toString**, mengembalikan informasi hewan dengan format string "Number of Legs: {numberOfLegs}, Child: {children}" (tanpa kutip)
- 5. **getAnimalPower**, metode abstrak dari kelas **Animal** yang mengembalikan kekuatan dari hewan tersebut

Lengkapi file [Animal.java](#)
Submit file **Animal.java**

Java 8

 [Animal.java](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	30	Accepted	0.07 sec, 26.16 MB
2	30	Accepted	0.07 sec, 28.42 MB
3	40	Accepted	0.07 sec, 29.14 MB

Time limit	1 s
Memory limit	64 MB

Inheritance - Animal Extended

Kerajaan hewan yang dimodelkan sebagai kelas `Animal` saat ini telah memiliki 3 jenis hewan yang berbeda-beda. Anda diminta untuk membuat kelas-kelas hewan yang memodelkan hewan-hewan tersebut. 3 hewan yang ada di kerajaan saat ini adalah `Gajah`, `Ayam`, dan `Serigala` yang dimodelkan sebagai kelas turunan `Animal` dengan nama yang sama. Spesifikasi bagi ketiga kelas tersebut adalah:

1. `Gajah`

- Merupakan turunan dari kelas `Animal`
- Spesifikasi Atribut:
 - Memiliki atribut `basePower`, yang menyimpan kekuatan dasar dari hewan tersebut.
 - Memiliki atribut `jumlahGading`, yang menyimpan berapa banyak gading yang dimiliki oleh gajah tersebut dalam tipe `long`
- Spesifikasi Konstruktor:
 - Konstruktor menerima `basePower` dan `jumlahGading` secara berturut-turut
 - Gajah memiliki 4 buah kaki (`numberOfLegs=4`)
 - Jumlah gading maksimal 2, semua `input` di atas itu akan secara otomatis di-set ke 2.
 - Jumlah gading minimal 0, semua `input` di bawah itu akan secara otomatis di-set ke 0.
- Spesifikasi Metode:
 - `getJumlahGading`, `getter` dari atribut `jumlahGading`
 - `getAnimalPower`, implementasi metode abstrak `getAnimalPower` dari kelas `parent Animal` yang mengembalikan kekuatan dari hewan ini. Gajah otomatis memiliki kekuatan yang berkali lipat karena ukurannya, namun dapat berubah tergantung dengan jumlah gading yang ia miliki. Formulanya adalah: $3 * basePower * (1 + jumlahGading)$

2. `Elang`

- Merupakan turunan dari kelas `Animal`
- Spesifikasi Atribut:
 - Memiliki atribut `basePower`, yang menyimpan kekuatan dasar dari hewan tersebut.
 - Memiliki atribut `jumlahTelur`, yang menyimpan berapa banyak telur yang dimiliki oleh Elang tersebut dalam tipe `int`
- Spesifikasi Konstruktor:
 - Konstruktor menerima `basePower`.
 - Elang memiliki 2 buah kaki (`numberOfLegs=2`)
 - Pada awalnya, elang tidak memiliki telur sama sekali.
- Spesifikasi Metode:
 - `getJumlahTelur`, `getter` dari atribut `jumlahTelur`
 - `bertelur`, menambahkan jumlah telur Elang dengan 1 telur lagi.
 - `getAnimalPower`, implementasi metode abstrak `getAnimalPower` dari kelas `parent Animal` yang mengembalikan kekuatan dari hewan ini. Elang harus melindungi telurnya namun dapat dibantu oleh anaknya. Formula kekuatannya adalah: $basePower * children - jumlahTelur$

3. `Serigala`

- Merupakan turunan dari kelas `Animal`
- Spesifikasi Atribut:
 - Memiliki atribut `basePower`, yang menyimpan kekuatan dasar dari hewan tersebut.
 - Memiliki atribut `kawanan`, yang menyimpan `state` apakah serigala tersebut memiliki kawanan atau tidak dalam tipe `boolean`
- Spesifikasi Konstruktor:
 - Konstruktor menerima `basePower` dan `kawanan` secara berturut-turut
 - Serigala memiliki 4 buah kaki (`numberOfLegs=4`)
- Spesifikasi Metode:
 - `hasKawanan`, `getter` dari atribut `kawanan`
 - `getAnimalPower`, implementasi metode abstrak `getAnimalPower` dari kelas `parent Animal` yang mengembalikan kekuatan dari hewan ini. Kekuatan dapat dihitung dengan formula:
 - Jika serigala memiliki `kawanan`, maka:
 - Jika ia tidak memiliki anak (`children=0`), maka kekuatannya hanya `basePower` saja.
 - Jika ia memiliki anak, maka kekuatannya akan jauh bertambah dengan nilai $basePower * 3 * children$
 - Jika serigala tidak memiliki `kawanan`, maka kekuatannya hanya `basePower` saja.

Lengkapi [Gajah.java](#), [Elang.java](#), dan [Serigala.java](#)

Submit file zip yang berisi file: `Gajah.java`, `Elang.java`, `Serigala.java`

Java 8

 [animal.zip](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	14	Accepted	0.07 sec, 28.31 MB
2	14	Accepted	0.07 sec, 28.98 MB
3	14	Accepted	0.07 sec, 28.01 MB
4	14	Accepted	0.07 sec, 28.01 MB
5	14	Accepted	0.07 sec, 26.20 MB
6	14	Accepted	0.07 sec, 27.78 MB
7	16	Accepted	0.07 sec, 27.98 MB

Question **3**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Inheritance - Kebun Binatang

Memanfaatkan kelas **Animal**, **Elang**, **Gajah**, dan **Serigala** yang telah didefinisikan sebelumnya, buatlah program utama **KebunBinatang.java**.

Suatu hari, Alice mengunjungi Kebun Binatang Bandung. Dia penasaran dengan kekuatan masing-masing hewan yang ada di kebun binatang tersebut. Bantulah Alice untuk mengetahui kekuatan hewan dari binatang berdasarkan karakteristik mereka.

Format Input

Baris pertama berisi sebuah bilangan bulat 1, 2, atau 3, yang menandakan jenis hewan:

- 1 untuk **Elang**
- 2 untuk **Gajah**
- 3 untuk **Serigala**

Beberapa baris berikutnya berisi informasi mengenai hewan, sesuai dengan jenis hewan:

- Untuk hewan **Elang**, memiliki **basePower** 5 dan menerima 2 baris berisi sebuah bilangan bulat berupa nilai atribut **children** dan atribut **jumlahTelur**,
- Untuk hewan **Gajah**, memiliki **basePower** 10 dan menerima 2 baris berisi bilangan bulat berupa nilai atribut **children** dan atribut **jumlahGading**,
- Untuk hewan **Serigala**, memiliki **basePower** 10 dan menerima 2 baris berisi sebuah bilangan bulat berupa nilai atribut **children** dan boolean berupa nilai atribut **kawanan**.

Format Output

Keluarkan 3 baris output,

Pada baris pertama, keluarkan informasi mengenai hewan yang dilihat oleh Alice.

Pada baris kedua, keluarkan informasi spesifik mengenai setiap hewan sebagai berikut:

- Untuk **Elang**, tunjukkan jumlah telur.
- Untuk **Gajah**, tunjukkan jumlah gading.
- Untuk **Serigala**, tunjukkan apakah memiliki kawanan atau tidak.

Pada baris ketiga, keluarkan informasi kekuatan hewan yang dilihat oleh Alice.

Contoh

Input 1

1
2
2

Output 1

Number of Legs: 2, Child: 2
Jumlah Telur: 2
Animal Power: 8

Input 2

2
2
2

Output 2

Number of Legs: 4, Child: 2
Jumlah Gading: 2
Animal Power: 90

Input 3

3
2
true

Output 3

Number of Legs: 4, Child: 2
Serigala berada di dalam kawanan
Animal Power: 60

Input 4

3
2
false

Output 4

Number of Legs: 4, Child: 2
Serigala tidak berada di dalam kawanan
Animal Power: 10

Perhatikan bahwa keluaran diakhiri dengan *newline*.

Lengkapi file [KebunBinatang.java](#).

Submit file **KebunBinatang.java**

Java 8

 [KebunBinatang.java](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.08 sec, 28.54 MB
2	20	Accepted	0.07 sec, 30.45 MB
3	20	Accepted	0.08 sec, 28.54 MB
4	20	Accepted	0.08 sec, 28.81 MB
5	20	Accepted	0.08 sec, 28.32 MB

Question **4**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Inheritance - Robot

Anda diminta untuk membuat kelas-kelas Robot untuk memodelkan robot di suatu dunia modern. Robot dimodelkan sebagai kelas **Robot** dengan berbagai jenis hewan merupakan turunan dari kelas **Robot**. Saat ini, dunia modern hanya memiliki 1 jenis robot yang merupakan induk robot berupa AI bernama **Sirik** yang dimodelkan sebagai kelas turunan **Sirik** dengan nama yang sama.

Pada soal ini, anda diminta untuk mengimplementasi kelas **Robot** yang merupakan sebuah *abstract class* yang memiliki 2 buah atribut:

- 1. **numberOfParts**, menyimpan banyak bagian dari robot dalam tipe data **int**
- 2. **isActive**, yang memberitahu apakah robot tersebut aktif/berfungsi dengan tipe data **boolean**

dengan kelas **Robot** ini juga memiliki beberapa metode yang memiliki peran berikut:

- 1. **getNumberOfParts**, getter dari atribut **numberOfParts**
- 2. **getIsActive**, getter dari atribut **isActive**
- 3. **setActive**, menetapkan nilai atribut **isActive** dengan nilai argumen **isActive**
- 4. **toString**, mengembalikan informasi robot dengan format string "Number of Parts: {numberOfParts}, IsActive: {isActive}" (tanpa kutip)
- 5. **getPrice**, metode abstrak dari kelas **Robot** yang mengembalikan harga dari robot tersebut

Lengkapi file [Robot.java](#)

Submit file **Robot.java**

Java 8

 [Robot.java](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.23 sec, 26.74 MB
2	25	Accepted	0.35 sec, 28.04 MB
3	25	Accepted	0.31 sec, 27.78 MB
4	25	Accepted	0.35 sec, 30.04 MB

Time limit	1 s
Memory limit	64 MB

Inheritance - Robot Extended

Robot yang ada pada dunia modern saat ini sudah bertambah menjadi 3 jenis. Anda diminta untuk membuat kelas-kelas robot yang memodelkan robot-robot tersebut. 3 robot yang ada di kerajaan saat ini adalah **Sirik**, **Walle**, dan **Bumblebee** yang dimodelkan sebagai kelas turunan **Robot** dengan nama yang sama. Spesifikasi bagi ketiga kelas tersebut adalah:

1. Sirik

- Merupakan turunan dari kelas **Robot**
- Spesifikasi Atribut:
 - Memiliki atribut **price** yang menyimpan harga dasar dari robot tersebut.
 - Memiliki atribut **totalModel** yang menyimpan banyak model untuk robot ini.
- Spesifikasi Konstruktor:
 - Konstruktor menerima parameter **price**.
 - Sirik tersusun atas 5 bagian dan awalnya berada dalam kondisi tidak aktif (**numberOfParts=5, isActive=false**)
 - Pada awalnya, hanya ada 1 jumlah model Sirik.
- Spesifikasi Metode:
 - **getTotalModel** *getter* dari atribut **totalModel**
 - **addModel** menambahkan **totalModel** sejumlah 1 buah.
 - **getPrice**, implementasi metode abstrak dari kelas **Robot** yang mengembalikan harga dari robot tersebut. **Sirik** memiliki harga yang bergantung pada jumlah model dan jumlah bagian yang ia miliki, dengan batasan jumlah model tidak boleh terlalu banyak. Formulanya adalah :
 - Jika **totalModel** kurang dari/sama dengan 2, maka formulanya adalah : **price + (500 * numberOfParts * totalModel)**
 - Jika **totalModel** lebih dari 2, maka harganya akan menurun dengan formula : **price + (500 * numberOfParts - (100 * totalModel))**

2. Walle

- Merupakan turunan dari kelas **Robot**
- Spesifikasi Atribut:
 - Memiliki atribut **mileage** yang menyimpan satuan jarak yang telah ditempuh oleh robot tersebut bertipe **long**.
- Spesifikasi Konstruktor:
 - Konstruktor menerima parameter **mileage** dan **isActive** secara berturut-turut.
 - Walle tersusun atas 10 bagian (**numberOfParts=10**)
- Spesifikasi Metode:
 - **getMileage** *getter* dari atribut **mileage**
 - **addMileage**, menambahkan jumlah mileage dengan nilai argumen **increment**.
 - **getPrice**, implementasi metode abstrak dari kelas **Robot** yang mengembalikan harga dari robot tersebut. Formula dasar harga Walle adalah **1000 * mileage** . Namun apabila ia tidak aktif, maka akan ditambah lagi harganya dengan **500**.

3. Bumblebee

- Merupakan turunan dari kelas **Robot**
- Spesifikasi Atribut:
 - Memiliki atribut **totalTransformation** yang menyimpan berapa banyak ia bisa berubah.
- Spesifikasi Konstruktor:
 - Konstruktor menerima parameter **numberOfParts** dan **isActive**.
 - Pada awalnya, **totalTransformation** yang tersedia hanya 1.
- Spesifikasi Metode:
 - **getTotalTransformation** *getter* dari atribut **totalTransformation**
 - **addTransformation**, menambahkan **totalTransformation** dengan 1.
 - **getPrice**, implementasi metode abstrak dari kelas **Robot** yang mengembalikan harga dari robot tersebut. Harga **Bumblebee** sangat bergantung dari apakah ia sudah aktif atau tidak. Formulanya adalah:
 - Dalam kondisi tidak aktif, maka harga robot adalah **numberOfParts * 1000**
 - Namun apabila ia sudah aktif, maka harganya adalah **numberOfParts * 1000 * totalTransformation**

Lengkapi [Sirik.java](#), [Walle.java](#), dan [Bumblebee.java](#)

Submit file zip yang berisi: **Sirik.java**, **Walle.java**, **Bumblebee.java**

Java 8

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	14	Accepted	0.07 sec, 29.78 MB
2	14	Accepted	0.07 sec, 28.18 MB
3	14	Accepted	0.08 sec, 27.81 MB
4	14	Accepted	0.07 sec, 26.91 MB
5	14	Accepted	0.07 sec, 28.12 MB
6	14	Accepted	0.07 sec, 26.93 MB
7	16	Accepted	0.07 sec, 28.65 MB

Question **6**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Inheritance - Dunia Robot

Memanfaatkan kelas `Robot`, `Bumblebee`, `Sirik`, dan `Walle` yang telah didefinisikan sebelumnya, buatlah program utama `DuniaRobot.java`.

Suatu hari, Cello mengunjungi Dunia Robot. Dia ingin membeli robot yang paling tepat buat diri sendiri. Bantulah Cello untuk mengetahui harga dan spesifikasi masing-masing model robot.

Format Input

Baris pertama berisi sebuah bilangan bulat `1`, `2`, atau `3`, yang menandakan jenis robot:

- `1` untuk `Bumblebee`
- `2` untuk `Sirik`
- `3` untuk `Walle`

Beberapa baris berikutnya berisi informasi mengenai robot, sesuai dengan jenis robot:

- untuk `Bumblebee`, menerima 3 baris berisi sebuah bilangan bulat berupa nilai atribut `numberOfParts` dan `totalTransformation` beserta boolean berupa nilai atribut `isActive`,
- untuk `Sirik`, menerima 2 baris berisi bilangan bulat berupa nilai atribut `price` dan atribut `totalModel`,
- untuk `Walle`, dan menerima 2 baris berisi sebuah bilangan bulat berupa nilai atribut `mileage` dan boolean berupa nilai atribut `isActive`.

Format Output

Keluarkan 3 baris output,

Pada baris pertama, keluarkan informasi mengenai robot yang dilihat Cello.

Pada baris kedua, keluarkan informasi spesifik mengenai setiap hewan sebagai berikut:

- Untuk `Bumblebee`, tunjukkan jumlah transformasi.
- Untuk `Sirik`, tunjukkan jumlah model.
- Untuk `Walle`, tunjukkan mileage.

Pada baris ketiga, keluarkan harga robot yang dilihat Cello.

Contoh

Input 1

```
1
5
5
true
```

Output 1

```
Number of Parts: 5, IsActive: true
Jumlah transformasi: 6
Harga robot: 30000
```

Input 2

```
2
1000
10
```

Output 2

```
Number of Parts: 5, IsActive: false
Jumlah model: 11
Harga robot: 2400
```

Input 3

```
3
5
false
```

Output 3

```
Number of Parts: 10, IsActive: false
Jumlah mileage: 5
Harga robot: 5500
```

Perhatikan bahwa keluaran diakhiri dengan *newline*.

Lengkapi file [DuniaRobot.java](#)
Submit file `DuniaRobot.java`

Java 8

 [DuniaRobot.java](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.08 sec, 28.66 MB
2	20	Accepted	0.08 sec, 27.99 MB
3	20	Accepted	0.07 sec, 28.52 MB
4	20	Accepted	0.07 sec, 27.91 MB
5	20	Accepted	0.07 sec, 27.89 MB

[◀ Feedback Praktikum 2](#)

Jump to...