

MID EXAM
School of Creative Industry
2017-2018 ODD

Department : Information and Multimedia Technology	Day/Date : Thursday / 28 Sep 2017
Code : IMT1011 – A IMT0012 -A	Time : 07:30 - 10:00
Subject : Object Oriented Programming	Smt/Room : 3 / Lab. Computer (A316)
Lecturer : Caecilia Citra Lestari S.Kom., M.Kom.	Sifat Ujian : Open Book

Pada komputer masing-masing terdapat folder Project UTSOOP_XXXXX yang merupakan tempat anda mengerjakan UTS ini.

Setelah **selesai**, pada **Finder / Windows Explorer** gantilah nama folder dari UTSOOP_XXXXX menjadi UTSOOP_<5 digit terakhir NIM anda>

Terdapat Critter yang memiliki atribut:

- - color : Color untuk menyimpan warna binatang
- # isHungry : Boolean untuk menyimpan status apakah binatang lapar/tidak
- # moveCounter : int untuk menyimpan berapa kali binatang bergerak
- # lastMove: String untuk menyimpan arah gerak terakhir binatang
- - x : int untuk menyimpan posisi x binatang
- - y : int untuk menyimpan posisi y binatang
- - health : int untuk menyimpan nilai kesehatannya
- - isDie : boolean untuk menyimpan status binatang masih hidup atau tidak.

dan method:

- fight(String opponent): boolean untuk menyerang binatang lain
- getColor(): Color untuk mengetahui warna dari binatang tsb
- getMove(): String untuk mengetahui arah gerak dan menggerakkan binatang
- getLastMove: String untuk mengetahui arah gerak terakhir binatang tanpa menggerakkan binatang itu.
- isHungry(): boolean untuk mengetahui apakah binatang lapar/tidak
- getHealth(): int untuk mengetahui nilai Health
- setHealth() untuk mengatur nilai Health binatang, ketika Health = 0 isDie bernilai true.
- setPosition(int x, int y) untuk mengatur posisi x,y binatang
- getX(): int untuk mengetahui posisi X binatang
- getY(): int untuk mengetahui posisi Y binatang
- isDie(): boolean untuk mengetahui apakah binatang mati/hidup
- eat() untuk memberi makan binatang;
- toString() untuk mengetahui binatang apakah itu.

Terdapat class Main yang memiliki method main() (psvm). Anda hanya boleh menjalankan program dari class ini. Tidak diperkenankan mengubah kode di dalam psvm.

1. (15 poin) Buatlah class Fox yang adalah Critter dengan beberapa *behaviour* berikut:
 - Nilai awal Health adalah 10
 - Fox selalu kalah bertarung. Ketika kalah bertarung dengan Cougar health -5, ketika kalah bertarung dengan Annaconda ia akan mati.

- Fox bergerak dengan pola N-E-S-W, sehingga pada move 1 ia bergerak ke N (method `getMove()` mengembalikan nilai 'N'), move 2 ia bergerak ke E, move 3 ke S, move 4 ke W, move 5 ke N, move 6 ke E, move 7 ke S, move 8 ke W, dst. Setiap selesai bergerak 4 kali, ia akan lapar
 - method `toString()` akan mengembalikan nilai "Fox"
2. (15 poin) Buatlah class Cougar yang adalah Critter dengan *behavior* berikut:
- Nilai awal Health adalah 15
 - Cougar menang bertarung dengan Fox namun kekurangan Health 2. Ia kalah dengan Annaconda dengan kekurangan Health 10.
 - Cougar bergerak dengan pola S-S-N-E-E-W. Ia akan lapar setiap selesai bergerak 2 kali.
 - Cougar dapat berlari ke arah yang ditentukan oleh parameter sejauh 4 kali jika ia tidak lapar, dan 2 kali jika ia lapar. Setelah berlari ia akan jadi lapar. Tambahkan method `run(String direction)` ke dalam class Cougar.
 - method `toString()` akan mengembalikan nilai "Cougar"
3. (15 poin) Buatlah class Annaconda yang adalah Critter dengan *behavior* berikut:
- Nilai awal Health adalah 20
 - Annaconda selalu menang bertarung. Namun ketika melawan Cougar Health berkurang -1.
 - Annaconda bergerak ke barat (W) dan timur (E) dengan pola yang semakin lama semakin lebar. Move-1 ia bergerak ke W, move-2 ke E, move-3 ke W, move-4 ke W, move-5 ke E, move-6 ke E, move-7 ke W, move-8 ke W, move-9 ke W, move-10 ke E, move-11 ke E, dst. Setiap sekali bergerak ia akan lapar.
 - method `toString()` akan mengembalikan nilai "Annaconda"
4. (5 poin) Buatlah class Food yang mempunyai atribut:
- `x : int`
 - `y : int`
- Buatlah method `get` dan `set` dari kedua atribut tersebut.
5. (10 poin) Tambahkan pada class `CritterFight` hal-hal berikut:
- Terdapat atribut `maxX` dan `maxY` untuk menyimpan maksimum posisi dari arena permainan
 - Terdapat atribut `fighters` yang merupakan `ArrayList` of `Critter`. Atribut ini berfungsi untuk menampung binatang-binatang yang bertarung.
 - Terdapat atribut `foods` yang merupakan `ArrayList` of `Food`.
 - Terdapat constructor yang memiliki parameter yang menerima jumlah Fox, Cougar dan Annaconda, serta nilai `maxX` dan `maxY` Di dalam constructor ini dibuatlah objek dari binatang-binatang itu sejumlah parameter. Semua binatang ditampung dalam `fighters`.
 - Gunakan `Math.random() * maxX` dan `Math.random * maxY` untuk menentukan posisi awal masing-masing binatang. Posisi binatang boleh sama.
6. (10 poin) Buatlah method `spreadFood()` yang membuat 10 objek `Food` dan meletakkannya di posisi *random*, namun tidak boleh ada 2 objek `food` yang memiliki posisi yang sama. Masukkan semua objek `Food` ke dalam `ArrayList foods`. Panggilan method ini di constructor
7. Tambahkan method `play()` pada class `CritterFight` yang akan melakukan hal-hal berikut:

- (10 poin) Setiap binatang bergerak 1 kali dengan memanggil method `getMove()` yang mengembalikan nilai N, W, E, atau S. Nilai N berarti binatang akan naik ke atas ($Y-1$), S berarti turun ($Y+1$), E berarti ke kanan ($X+1$), dan W berarti ke kiri ($X-1$).
 - (10 poin) Ketika seekor binatang berada pada posisi sama dengan sebuah Food dan jika ia lapar, maka ia akan memakannya. Panggilah method `eat()` dan hilangkan objek Food yang sesuai dari `ArrayList foods`.
 - (10) Ketika seekor binatang berada pada posisi sama dengan binatang lain, mereka akan bertarung (memanggil method `figh()` dari kedua objek), kecuali objek Cougar. Jika bertemu dengan Annaconda dan memiliki Health antara 5-10, objek Cougar akan berlari ke arah sebaliknya dari `lastMove`-nya, yaitu jika `lastMove = N` maka method `run()` mendapat parameter S, jika `lastMove = W` maka ke E, dan sebaliknya.
 - (5 poin) Lakukan pengecekan pada dua binatang yang bertarung. Jika binatang mati (`isDie()` mengembalikan nilai True), hilangkan dari `fighters`.
 - (10 poin) Ketika seekor binatang berada pada posisi sama dengan binatang sejenis, maka mereka akan beranak. Program akan membuat sebuah objek baru dari class yang sama memberikan posisi X dan Y random pada objek baru tersebut, dan memasukkannya ke `fighters`.
 - (5 poin) Lakukan pengecekan persediaan makanan. Jika objek Food di dalam `foods` kurang dari 3, maka panggil method `spreadFood()`;
 - Lakukan semua hal di atas hingga hanya ada 1 binatang yang tersisa.
8. (10 poin) Buatlah *report message* untuk setiap posisi dan aktivitas dari binatang-binatang tersebut seperti contoh berikut:
- <objek Critter> bergerak ke <arah>, sehingga posisi menjadi <x>,<y>
 - <objek Critter> memakan buah.
 - <objek Critter> <kalah/menang> bertarung dengan <objek Critter>
 - <objek Critter> mati
 - <objek Critter> beranak pinak.

Ganti <objek Critter> dengan jenis Critter (Fox, Cougar, atau Annaconda) dan nomor index pada *fighters*.