Koç University
COMP202
Data Structures and Algorithms
Assignment 1

Instructor: Barış Akgün
Due Date: February 11 2020, 23:59
Submission Through: Blackboard

This programming assignment will test your knowledge and your implementation abilities of what you have learned in the Java and OOP Concepts parts of the class.

This homework must be completed individually. Discussion about algorithms and data structures are allowed but group work is not. **Coming up with the same algorithm and talking about the ways of implementation leads to very similar code which is treated as plagiarism!** Discuss carefully. Any academic dishonesty, which includes taking someone else's code or having another person doing the assignment for you will not be tolerated. **By submitting your assignment, you agree to abide by the Koç University codes of conduct.**

Description

This assignment is designed to assess your understanding of java essentials; abstractions, interfaces, encapsulation, polymorphism and access control of java methods. The project is about animals. There are many species in animal kingdom with many common behaviors as well as their different properties. In this assignment you are required to implement some of the parts of the provided files to complete it. The code has comments about the assigment. Majority of what you need to do is actually in these comments so make sure you read them carefully.

The descriptions for the provided files are below. Bold ones are the ones you need to modify and they are placed under the *code* folder, with the rest under *given*.

- *Animal.java*: This file includes the definition of the `Animal` abstract class. You are going to need to look at this file however you do not need to modify anything in it.
- *iHabitat.java*: This file includes the definition of the `iHabitat` interface which is implemented by the `Animal` class. You are going to need to look at this file however you do not need to modify anything in it.
- ***iBehavior.java***: You need to implement the `iBehavior` interface in this file. Follow the instructions given in the code.
- ***Cat.java***: You are provided with the basic structure of the `Cat` class in the file. Follow the instructions carefully to complete the sections where you are required to write code.
- ***Angora.java*** and ***Caracal.java***: You are required to implement the `Angora` and the `Caracal` classes by following the instructions given in the files.
- *Main.java*: Use this file to print outputs for Angora, Caracal and the Cat classes.
- *Util.java*: This file includes utility functions. Do not worry about it, you will not need to use anything from this file in this assignment.

Grading

Your assignment will be graded through an autograder. Make sure to implement the code as instructed, use the same variable and method names. We will not only check the outputs but your whole implementation will be assessed such as access modifiers of methods and variables, input/output parameters of methods and class constructors.

A version of the autograder is also released to you. Its files are under the *autograder* folder. Our version will more or less be similar, potentially including more tests. Run the main program to get the autograder output and your grade.

In case the autograder fails or gives you 0 when you think you should get more credit, do not panic. Let us know. We can go over everything even after your submission.

This assignment is very easy. It will be worth 3% of your total grade. It will also most likely lead to a lot of unintentionally similar submissions. As a result, we are not going to do a plagiarism check.

Submission

You are going to submit a compressed archive through the blackboard site. The file should extract to a folder with your student ID without the leading zeros. This folder should only contain files that were in **boldface** in the previous section. Other files, which you should not have modified anyways, will be deleted. Other files will be deleted and/or overwritten.

Important: Make sure to download your submission to make sure it is not corrupted and it has your latest code. You are only going to be graded by your blackboard submission.

Submission Instructions

- You are going to submit a compressed archive through the blackboard site. The file can have *zip*, *tar*, *rar*, *tar.gz* or *7z* format.
- This compressed file should extract to a folder with your student identification number with the two leading zeros removed which should have 5 digits. Multiple folders (apart from operating system ones such as MACOSX or DS Store) greatly slows us down and as such will result in penalties
- Code that does not compile will be penalized and may receive no credits.
- Do not trust the way that your operating system extracts your file. They will mostly put the contents inside a folder with the same name as the compressed file. We are going to call a program (based on your file extension) from the command line. The safest way is to put everyting inside a folder with your ID, then compress the folder and give it your ID as its name.
- One advice is after creating the compressed file, move it to your desktop and extract it. Then check if all the above criteria is met.
- Once you are sure about your assignment and the compressed file, submit it through Blackboard.
- After you submit your code, download it and check if it the one you intended to submit.
- **DO NOT SUBMIT CODE THAT DOES NOT TERMINATE OR THAT BLOWS UP THE MEMORY.**

Let us know if you need any help with setting up your compressed file. This is very important. We will put all of your compressed files into a folder and run multiple scripts to extract, cleanup and grade your codes. If you do not follow the above instructions, then scripts might fail. This will lead you to get a lower grade than what the autograder suggests.