LP24 Object Oriented Programing
Final Exam - Spring 2017

Documents, Computer, Calculator unauthorized

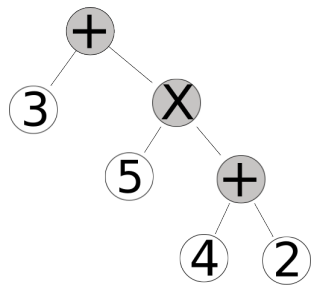
1. Theoretical questions (3 pts)

A. What are the pillars of OOP?

B. What is inheritance? Explain the differences between inheritance and composition. When should we prefer one or the other?

C. Explain the difference between a class and an object.

2. Exercise #1: Computations (8 pts)



The goal of this exercice is to develop a program aimed at evaluating arithmetical equations such as the one presented in the figure. For this purpose, we will use binary trees for representing an expression. Thus, the tree presented in the figure corresponds to the following expression: 3+5*(4+2)
A tree representing an expression is composed of nodes from two types: (1) constant value (2) binary operator.

The code which evaluates the value of the example can be written as follows:

```
class TestExprExample{
     public static void main(String [] arg){

     Constant c1 = new Constant (3) , c2 = new Constant (5) , c3 =
new Constant (4) , c4 = new Constant (2);

     Operator o1 = new Addition(), o2 = new Multiplication(), o3 =
new Addition();

     o1.setRL ( c1 ) ; o1.setLL ( o2 ) ; o2.setRL ( c2 ) ;
o2.setLL ( o3 ) ; o3.setRL ( c3 ) ; o3.setLL ( c4 ) ;

     System.out.println (o1.eval ( ) ) ;

     }

}
```

Question 1: Write the class `Node` which describes what a node is. Can the `Node` class be considered as a regular class? Why?

Question 2: Write the class that describes the constants

Question 3: Operators are nodes that can be evaluated depending on their type. However, they are sharing some common features. How to manage this in Java?

Question 4: Write the classes necessary for defining the 4 classical operators?

3. Exercise #2: The droid classification (9 pts)

We consider the Droid class which is aimed at representing the characteristics of a droid. A droid is defined by a name, a release date, a version, a manufacturer and a set of type numbers.
A droid can be from one or several types, each of them being associated to a set of possible actions such as firing for droid from type #4 or repairing ships for droid from type #2.
The types of droids are the following:

- The first type was made up of droids skilled in mathematical, physical, and medical sciences.
- Droids of the second type were skilled in engineering and technical sciences.
- Third type droids were skilled in social sciences and service functions.
- Droids of the fourth type were skilled in military and security functions
- Fifth type droids were programmed for menial labor and intensive jobs that did not require a high level of intelligence

Question 1: Write the enum `Type` which will represent the available types of droids. Why do we need an enum instead of using an integer for representing the number of the droid types.

Question 2: Write the `Droid` class. Is this class a regular class?

Question 3: How can we manage the 5 possible types of droids. Can we define a Java class for each droid type? What Java feature can we use then? Give all necessary the code which will allow to define droids from the proposed types.

Question 4: Looking for R2D2. Using your classes, you can now define the Astromech droid type which is from droid type #2. Write this class.

Question 5: All the droids have a location device. Considering that all the droids from the empire are store into an ArrayList named myDroids, write the lines of codes of a main program that will make a call to the `locate` function of all the imperial droids. What features of OOP is used for making this easily possible?