CSE3311 SOFTWARE DESIGN – ASSIGNMENT 4
SYSTEM DESIGN
VER. 1.0

PRZEMYSLAW PAWLUK

Due: Wednesday, July 19, 2013

Where: Electronic

Weight: 10%

1. Main Points

Be sure to read and follow all the guidelines from the links on reports and academic honesty from the WWW home page for the course. The specification is the union of this document plus the program text you are given.

1.1. Learning objectives.

- Reading and writing assertions
- Reading and understanding contracts
- Verifying the correctness of algorithms

1.2. **To hand in.** Hand in, in class, a report containing the following items as a package in the given order.

(1) Cover page printed from the course web pages
(2) Your report consisting of your solutions to the tasks in Section 2. For reports done in pairs, include an appendix describing the contributions of the two team members. (electronic document)
(3) Electronic submission: **There is only electronic submission for this report.**

2. Tasks

2.1. **Cars Design.** You will design and implement classes for cars. There are three types of cars: Sedan, Compact, and Sports. Each car contains an integer amount of gas. All cars respond to the gas feature, which increments the amount of gas by 1 up to a maximum of 50, and prints out "Gas!" to the standard output. Except the Compact car, which after printing "Gas!" also prints "Well, I have to work hard again".

All cars respond to the accelerate feature, which decreases the amount of the gas by 1 and increases the speed of a car by 1. A car prints out "Faster!" when it receives the accelerate feature. However, Sports cars get so excited that they prints "It really feels good!" which costs the Sports car two more units of gas, but increases the speed by another one. All the cars respond to the brake feature, which decreases the amount of the gas, as well as speed, by 1. The speed is represented by an integer between 0 and 200, except for a Sports car, which can reach 300. When the speed is greater than $(3 * gas - 50)$, a car responds by printing "Speeding!" A Sedan car will further complain "Why hurry?" The exertion of printing the extra words costs Sedan an additional unit of gas

Draw a class hierarchy in BON, which specifies the relationships among all classes you designed (no interface details). Also, implement your classes in Eiffel with suitable contracts and invariants. Your design should be general enough to easily add new types of cars.

2.2. **BON: Classes and Relationships.** For each of the following requirements, identify candidate classes (with appropriate names) and define their relationships using BON. Associations and aggregations must be labeled. If no operations are specified, then do not provide any class interface details and use the short ellipse notation for classes. If operations are specified, then give appropriate interfaces and class invariants but do not give any contracts. Briefly explain each diagram in English.

2.2.1. *(Family) A father or a mother may have a son or daughter. A son or daughter may have a brother or a sister.*

2.2.2. *(Folders) A folder in a personal computer can contain folders, text files, binary files, and pictures. Folders, files and pictures can be opened and closed. Folders, files and pictures can be inserted and removed from a folder.*

2.2.3. *(Dental Office) A dental office has a number of patients. A patient is either male or female. Each patient has a set of teeth. The office would like to keep track of each patient's information including address and date of birth as well as information regarding his/her complete set of teeth. For each tooth a history is kept which is simply a note indicating the condition of that tooth. In addition, the office would like to keep a record of all operations performed on a tooth together with the date of the operation. An operation could be a root canal or an extraction. If a tooth is extracted then no other operation may be performed on that tooth. A tooth may have at most one root canal operation. Your system must be able to query the system for the following information.*

- A list of patients with a root canal operation.

- The number of teeth extractions performed in a month.
- All operations performed on female patients within certain date.

The design should be at a high level of abstraction, using deferred classes such as SET and LIST. You may assume that lower level classes such as DATE and ADDRESS exist. For the purpose of this question, we are only interested in overall architecture rather than implementation and efficiency. Identify constraints that are class invariants and indicate the classes that contain those invariants.

3. Grading scheme

The grade for the report is partitioned into the following parts.

(1) Overall presentation – 10%
(2) Cars Design – 30%
(3) BON: Classes and Relationships (total 60%)
    - Family – 10%
    - Folders – 20%
    - Dental Office – 30%