

# ENGI 9874

## Software Design and Specification

### Assignment 1

Due Friday, October 5th, 2018 by 4:00 PM

## Instructions

Hand this in on paper, in lab or under my office door by 4 PM. I recommend using Visual Paradigm to create each diagram, but your printed submission should be well-organized to clearly communicate your answers.

This is an individual assignment. Complete it by yourself, alone. (Although if you are trying to find a feature in Visual Paradigm, you can ask for help with that.)

## Question 1: Domain Analysis [25 marks]

For this question, you must analyze a problem and model the relationships expressed in that problem. This is not a software design question; rather, you are modeling the real things described and their interactions. You must create a class diagram based on the following information.

Customers of an airline can join their rewards program and earn points that can be used for benefits with the airline. Initially, members earn 1 point per dollar spent with the airline. After earning 10000 points, they gain “Elite” status with even more benefits and earn 3 points per dollar spent.

Customers, regular members, and elite members and purchase the following from the airline: tickets, advance seat selection, checked bags, in-flight entertainment, and in-flight food and drink.

Regular members and elite members can use points to pay for advance seat selection, checked bags, in-flight entertainment, in-flight food and drink, and day-of-travel seat upgrades.

Elite members can also purchase tickets using points, and get priority boarding before other customers.

Members are identified by a member number. Using points reduces the amount you have, but spent points are still counted toward the 10000-point threshold for Elite status.

## Question 2: Software Design [35 marks]

A basic electric circuit consists of resistors, inductors, and capacitors which have behaviour that can be defined by known mathematical equations. Each of these devices has 2 terminals, which we can call the input terminal and output terminal (though it often doesn't matter which is which).

These components are wired together with power supplies and switches, etc. to build basic circuits. We can model the wiring with the concept of a “net”, short for “network of wires connecting terminals” in this instance. A “net” is simply a connection point between 2 or more devices.

- (a) Draw a UML class diagram to represent software designed to model and simulate basic electric circuits (i.e. supporting resistors, inductors, capacitors, a single DC power supply, and switches). You will need a class hierarchy for components and nets, a class to represent the overall circuit, and perhaps more. Show appropriate associations with navigability, multiplicity, and annotations indications bags, ordered sets, etc. [12 marks]
- (b) Elaborate the class diagram to add attributes and operations as needed. It should be possible for for the circuit object to have component and nets added to it, and specific terminals of components can be added to specific nets, and the circuit simulate command should simulate all the components for a timestep starting with the power supply. [11 marks]
- (c) Show with a sequence diagram how a client can create a voltage divider circuit (i.e. a power supply and two resistors in series, connected by appropriate nets). [12 marks]

## Question 3: Chess Game Design [40 marks]

- (a) Create a class diagram for the following set of classes representing a chess game. Include all operations. [14 marks]
  - There are different pieces in the game: Pawn, Rook, Knight, Bishop, King, and Queen. These are all specializations of a general Piece. Each piece knows how it can move but must send a message to the board asking to be moved to a specific location (given in x,y coordinates) and can `draw()` itself on the screen.
  - There is a Board that knows where all the pieces are. After each `move(piece, x, y)`, it must `resolve()` any changes to the board state (i.e. captured pieces, etc.).
  - There is a Referee object that runs the game, allowing players to move in turn.
  - There is a GameView object that handles the user interface and display of the board. It can request the location of pieces from the Board, draw the board (and then each piece on the board), and accept input from users to trigger moves.

So after user input triggers a move request, the board resolves it, lets the view and referee know, and the view redraws the screen.

- (b) Make an object diagram showing the Board when there are 2 Kings (Black and White) and a Queen (Black) left on the board. (You may need to look up some reference on object diagrams, but Visual Paradigm draws them.) **[12 marks]**
- (c) Show with a sequence diagram (referring to the board setup from part b) the process of the user moving the Black Queen to capture the White King piece. **[14 marks]**