# Domain Modeling with Umple

## 1 Required Software Engineering Tools

- Umple Online
  (http://cruise.eecs.uottawa.ca/umpleonline/)
- Umple Command-line based Compiler
  (http://cruise.eecs.uottawa.ca/umple/UmpleTools.html)

You may also use the following:

- Eclipse Neon Modeling Tools
  (http://www.eclipse.org/downloads/packages/eclipse-modeling-tools/neonr)
- Umple Eclipse plugin 1.22.0.5146 (or later)
  (http://cruise.eecs.uottawa.ca/umpleonline/download_eclipse_umple_plugin.shtml )

## 2 User Story for Fantasy Basketball Game

We are developing an online, web-based software for allowing users to play fantasy basketball competitions run in parallel with each real NBA season. Users of the fantasy basketball game assemble virtual teams with a unique name consisting of 12 real basketball players selected from arbitrary NBA teams. Players are identified by their license ID and we store their first name, last name, and salary. Users may select each player of their virtual team from a different real team, but they can use only a give budget to sign players for their team. Unlike in real basketball games, a real player can be signed by many virtual teams. Within a virtual team, 5 players are in the starting line-up and the remaining 7 players constitute the bench.

Virtual teams get scores based on the performance of their players in real basketball matches. A player's virtual score is calculated after each match as the sum of his points, assists and rebounds. If this player is selected in the starting line-up of a virtual team by a user, then this team receives double score from this player. The system shall provide live rankings of virtual teams refreshed after each match played.

## 3 Domain Model and Code Generation

Given the above user story, describe all concepts of the Fantasy Basketball Game System and their relationships in an Umple domain model. You may add extra attributes not mentioned the text above, but it is not compulsory. Then, generate Java and Php code from the domain model. Note that for the purpose of this assignment, you do *not* need to include a root class representing the application that contains all other classes.

# Submission

This assignment is to be done individually. You are required to hand in a ***single zip file*** containing

(i) the Umple program (a .ump file),
(ii) the class diagram generated by Umple Online (or by other modeling tool for UML Class diagrams if Umple Online is not available) either in an MSWord file or a PDF file,
(iii) Java code generated from the Umple program, and
(iv) (iv) Php code generated from the Umple program.

The assignment is due ***Wednesday, February 15, 2017 at 23:30***. If you realize that you need to make changes to your submission, do not resubmit only the file(s) that have changed, but rather resubmit another complete zip file. If you are using an application other than MSWord to hand in your class diagram, convert your report first to either a PDF file or a DOC(X) file.

# Marking Scheme

| ***Part of Assignment*** | ***Marks*** |
|---|---|
| Correct definition of classes, attributes, associations, containments and multiplicities in the Umple program | 70 |
| Class diagram consistent with the Umple program | 10 |
| Generated Java code | 10 |
| Generated Php code | 10 |
| | |
| Total Marks: | 100 |