*University of Sheffield, Department of Computer Science*

# COM2001: Advanced Programming Techniques

# Semester 1: Functional Programming

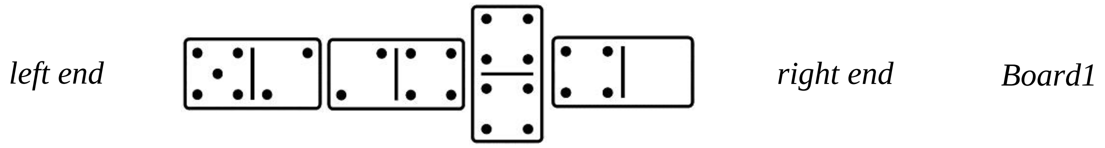
# Assessment 2017-18

# Assignment 1: Dominoes

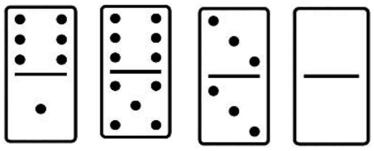
*This exercise counts for 12.5% of the assessment for the COM2001 module*

## 1. Introduction

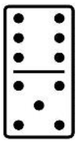
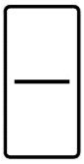
You are probably familiar with the game of dominoes. If not, see
*http://en.wikipedia.org/wiki/Dominoes*
Here is an example of a dominoes **board** part-way through a game

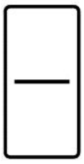
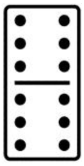


If the next player has in her **hand** the following:



*Hand1*

Then she may play  at the left **end**  or at the right end. The other dominoes may not be played.[1]

A standard set of dominoes contains all the permutations from  to,  28 in all.

This assignment involves representing and processing dominoes, domino boards and domino hands in Haskell.

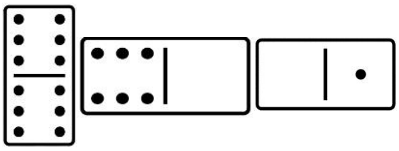
## 2. What you must do

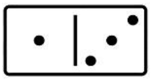
1. **Datatypes** : create Haskell data structures to represent
   a. a **Domino**
   b. a **Hand**
   c. a **Board**

[1] Some dominoes games allow plays up and down as well as left and right from a double, but we aren't considering this.

   d. an **End**

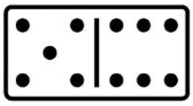
2. **Algorithms:** write and test the following functions in Haskell:
   a. **goesP:** predicate returning **True** if a given domino can be played at a given end of a given board.
   b. **knockingP:** predicate given a **Hand** and a **Board** returning **True** if there is no domino in the hand which can be played on the given board (i.e. the player is ‘knocking’).
   c. **playedP:** predicate returning **True** if a given **Domino** has already been played on a given **Board**.
   d. **possPlays:** given a **Hand** and a **Board**, return all the **Dominoes** which may be played at the left **End** and all those that may be played at the right **End**. The return type should be a pair (§13 of the lecture notes).
   e. **playDom:** given a **Domino**, a **Board** and an **End**, play the domino at the given end if it will go, returning the new **Board**. The return type should be a **Maybe** (see §14.3 of the notes).
   f. **scoreBoard:** in the UK, the dominoes game which is played seriously is **5s-and-3s** (http://www.pagat.com/tile/wdom/fives_and_threes.html) . In this game players score points by making the ends of the board add up to multiples of 3 and/or multiples of 5. So *Board1* scores 1 (5+0=5, =1*5). This board would score 0:



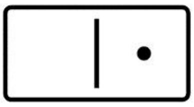
but playing  at the right would score 8 (5*3+3*5). Note that a double n adds 2*n to the 5s-and-3s count, making it 15 in this case.

**scoreBoard** takes a board and returns its 5s-and-3s score.

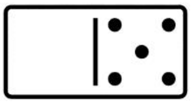
   g. **scoreN**: given a **Board** and an **Int** n, return all the **Dominoes** not already played which would give a 5s-and-3s score of n and the **End** at which to play each one.For example, **scoreN board1 2** should return



at the left



or



at the right

## 3. Marking Scheme

| | Credit |
|---|---|
| **Data Structures** | **15** |
| **goesP** | **10** |
| **knockingP** | **10** |
| **playedP** | **10** |

| | |
|---|---|
| **possPlays** | **15** |
| **playDom** | **10** |
| **scoreBoard** | **15** |
| **scoreN** | **15** |

***Assessment Criteria****: In function definitions, 60% of the credit is for coding, 20% for testing and 20% for documentation. A function which is working but poorly coded will be awarded about ½ of the credit for coding (i.e. 30% of the total credit for the function). The remaining coding credit is for good use of Haskell and the quality of the code.*

***Late hand-in penalties: see***

http://www.dcs.shef.ac.uk/intranet/teaching/public/assessment/latehandin.html

***Plagiarism penalties: see***

http://www.dcs.shef.ac.uk/intranet/teaching/public/assessment/plagiarism.html

## 4. What to hand in

Hand in 2 documents in a single zip archive:

1. Your commented code as a .hs file, ready to run
2. Your test results

## 5. How to hand in

*Hand in by MOLE*

## *6.* DEADLINE: Midnight Monday 23rd October (week 5)