TME4013 Formal Methods, Sem 1 2020/2021

**Lab Data Definition Exercise**

1. Define class Book
2. Define data type for Title as of type seq of char.
3. Define instance variable for bookTitle as of type Title, and numberOfPages as of type natural.
4. Define constant value for book Price as RM20.50
5. Define invariant for numberOfPages more or equal to ONE but less than or equal to 100 pages.

Submit your Lab Data Definition Exercise to eLEAP before 8 Nov 2020, 5m.

-------------------------

-- Lab 1 - data definition

-------------------------

class Book

types

public Title = seq of char;

instance variables

private bookTitle: Title;

private numberOfPages: nat1;

inv numberOfPages >= 1 and numberOfPages <= 100;

values

private Price: real = 20.50;

------------------------------

-- Lab 2 - operation definition

-- explicit & implicit

------------------------------

operations

expOperationSignature: nat1 \* real ==> real

expOperationSignature (a, b) ==

(

a := a \* b/a + b;

return a;

)

-- explicit operation

-- 1. define the explicit operation definition signature

-- to calculate the total price of book (e.g., CalculateTotalPrice()).

-- 2. Define the operation body (algorithm) that calculate the total price.

-- This is done by multiplying number of book with price of each book

-- (e.g., totalPrice = noofBook \* price).

-- 3. Return the value of totalPrice to operation CalculateTotalPrice().

operations

impOperationSignature(b:real) c:real

pre b > c and b <> 1.123

post c = b \* b/c

-- implicit operation

-- 1. define the implicit operation definition signature

-- to calculate the total price of book (e.g., CalculateTotalPrice()).

-- 2. Define the pre-conditions, that the number of book and the price of book

-- must greater than zero.

-- 3. Define the post-conditions that the total price of the book is defined

-- by the number of book mutiply by price of each book

-- (e.g., totalPrice = noofBook \* price).

-- Submit your Lab2 to eLEAP.

end Book

**----------------------------------**

**-- Lab 3 Use Mapping and Composite type**

**----------------------------------**

class Book

types

public Title = seq of char;

**public Category = <Software> | <Database> | <Network>;**

* 1. **Define type ‘BookStore’ as of type record consists of bookStoreName and bookStoreAddress.**
  2. **Define type mapping (e.g., CategoryToBookStore) that model the mapping of type ‘Category’ to type ‘BookStore’**

instance variables

private bookTitle: Title;

private numberOfPages: nat1;

inv numberOfPages >= 1 and numberOfPages <= 100;

* 1. **Define the instance variable of type ‘CategoryToBookStore’**
  2. **Define the instance variable of type ‘BookStore’**

values

private Price: real = 20.50;

operations

**searchBookStore: Category ==> CategoryToBookStore**

**SeacrhBookStore (category) ==**

**(**

* 1. **find book category e.g., if category = <software>**
  2. **then map that category to instance variable name in 3.4, and assign it to variable in 3.3.**
  3. **else assign with empty mapping, e.g., {|->}**
  4. **finally return the mapping value to return type ‘CategoryToBookStore’.**

)

end Book