ISYS1083/1084 Object Oriented Software Design

TuteLab 4 - Design by Contract (DbC)

PART 1: Tutorial Questions

1. Briefly explain the following:

   (a) Pre-condition
   (b) Post-condition
   (c) Invariant
   (d) Specification
   (e) The 'old' keyword.
   (f) 'Result' variable.

2. Defensive Programming

   (a) What is defensive programming?
   (b) Why is this good or bad?
   (c) What defensive code can be removed when using DbC?
   (d) Discuss the differences between Design by Contract and defensive programming.

3. Blame

   (a) What is the concept of "Blame" in Design by Contract?
   (b) If a pre-condition is violated, who is to blame?
   (c) If a post-condition is violated, who is to blame?
   (d) If an invariant is violated, who is to blame?
   (e) Is this a good or bad thing?

4. Java.util.Set

Consider the `Set<E>` interface in the `java.util` package (from Java 5+ API). Identify the pre-conditions and post-conditions for the following operations (the API documentation can help you here, especially the throwable exceptions).

- `int size()` returns the number of elements in the set
- `boolean add(E e)` adds an element to the set
- `boolean remove(Object o)` removes an element e from the set (note that the argument is not parameterised by type E)
- `boolean contains(Object o)` returns true if the object is contained in the set (again the argument is of type Object not E).

Why are the last two arguments typed as Object and not E?

Do contracts exist in all APIs or just ones with pre/post-conditions?

Part 2 – Lab Exercises

5. Sample Exercise (from the lecture)

There is an example exercise at the end of the Topic 4 lecture notes. Maybe you thought we had forgotten about this one? Well now that you have completed the first four questions now is a good time to have a go at that example :)