Reading Assignment

Object-Oriented Design Heuristics, Chapter three
Class invariants

“Class invariants are predicates of (statements about) a class that should always be true”


Examples are:

An instance variable is not nil
An instance variable is an ordered collection
An integer value has to be in a certain range
Example

Stack

Instance variables: elements, top

elements – Array containing the element of the stack
top – An integer pointing to element that is currently the top of the stack

Stack>>isEmpty
  ^top = 0

Stack>>isFull
  ^top = elements size

Stack>>pop
  self isEmpty ifTrue: [invoke your empty stack policy].
  topElement := elements at: top.
  top := top – 1.
  ^topElement

Stack>>push: anObject
  self isFull ifTrue: [invoke your full stack policy].
  elements at: (top := top + 1) put: anObject.
Class Invariants

Are to be true

- Once the object is constructed
- Before and after each message send to an object

In the middle of a message send to an object the class invariant may be false
WordStream

Parent class: ReadStream

Inherited instance variables:

• collection   <SequenceableCollection> elements to read
• position     <Integer> pointer to the current access position
• readLimit    <Integer> size of the collection
• writeLimit   <Integer> farthest that has been written into the collection
• policy       <StreamPolicy> policy for choosing the print format for various entities, such as Dates, Times, currencies, or other context-sensitive information

WordStream>>on: aCollection
    ^super on: (aCollection runsFailing: [:each | each isWordSeparator])