What is DBC?

- Classes of a system communicate with one another on the basis of precisely defined benefits and obligations.

[Bertrand Meyer, CACM, Vol. 36, No 9, 1992]
What is DBC? (cont.)

- **Preconditions of methods**
  A boolean expression which is assumed true when the method gets called

- **Postconditions of methods**
  A boolean expression which the caller can assume to be true when the method returns

- **Class invariants**
  consistency conditions of objects must hold for all instances
Preconditions, Postconditions and Class Invariants (cont.)


e.g., inserting a certain element into a dictionary (a table where each element is identified by a certain character string used as key) of bounded capacity.

\[
\text{put } (x: \text{ELEMENT}; \text{key: STRING}) \text{ is}
\]
\[
\quad -- \text{Insert } x \text{ so that it will be retrievable through key.}
\]
\[
\text{require}
\]
\[
\quad \text{count} \leq \text{capacity}
\]
\[
\quad \text{not } \text{key}.\text{empty}
\]
\[
\text{do}
\]
\[
\quad ... \text{Some insertion algorithm} ...
\]
\[
\text{ensure}
\]
\[
\quad \text{has} (x)
\]
\[
\quad \text{item} (\text{key}) = x
\]
\[
\quad \text{count} = \text{old} \text{count} + 1
\]
\[
\text{end}
\]
The Notion of Contract

- **Obligations**
  (Must ensure precondition)
  
  Make sure table is not full and key is a non-empty string.

- **Benefits**
  (May benefit from postcondition)
  
  Get updated table where the given element now appears, associated with the given key.

- **Supplier**
  (Must ensure postcondition)
  
  Record given element in table, associated with given key.

- **Client**
  (May assume precondition)
  
  No need to do anything if table is full, or key is empty string.
The Notion of Contract (cont.)

- Obligations
  - Satisfy preconditions
  - Satisfy postconditions

- Benefits
  - No need to check output values
  - Result guaranteed to comply to postcondition
  - No need to check input values
  - Input guaranteed to comply to precondition

Current Existing Tools (cont.)

- JDK1.4: assert
- Eiffel
  - http://archive.eiffel.com
- iContract
  - by Reto Kramer, Cambridge Technology Partners
- JASS
  - http://semantik.informatik.uni-oldenburg.de/~jass/
- JMSAssert
  - by man machine systems
- Jtest and Jcontract
  - by parasoft
  - http://www.parasoft.com