Read ALL of ch1, ALL of ch2, begin ch3 for project, ALL of both

4. An old throughput into classes

3. assert() macro, proper and improper use

2: Pre and Post conditions.


2. Assignment experience with DDD.

process requirements (MULTI-MODULE SOFTWARE) semantics

The focus of the lab: Our lab's development environment and

1. Review: Variable, semantics of assignment

(handouts available)

handout, with lab sessions schedule. (Lab 1 begins TODAY, extra

0. Lab, project, and electronic submission policy and instructions

CSI 310: Lecture 2
The assignment operator means \texttt{COPY} a value into a variable.

\texttt{MORE CHEAPLY} than with past technology.

network technology enable copies of information to be made MUCH

copyright issue changes that we face because current computer and

\texttt{NEW YORK TIMES MAGAZINE} of January 25, 2003 article about

\texttt{somebody the thing itself?}

\texttt{COPY} of something (like a Jaguar sports car) and \texttt{COPY}

\texttt{What is the main difference between giving somebody else a}
Usually, RAM: Random Accessible Memory

which costs money.

Each variable is implemented by a separate piece of hardware

scalable: cell, box, storage facility

syntactic: object, instance, memory location, location,

**variable**

Data is more important than executable code.
Values in variables can be copied into variables or used as stores.

A variable is a data structure. Its value is the data it currently stores.

The variable is not the same as its value.

```java
{ 
    The value of N is 8.  
    N + 5;  
    Now the value of N is 3.  
    N = 3;  
    Assignment stmt.  
    The value of N is unpredictable.  
    if (N} 
    main()
```
value does change.

created. That identity is unchanged even though the variable's
created. Each variable retains the identity it gets (WHAT IT IS) when it is
contents (what it contains)

SYNONYMS FOR "VALUE IN A VARIABLE": THE VARIABLE'S STATE, ITS
SYNONYMS FOR "VALUE IN A VARIABLE": OBJECT, INSTANCE, MEMORY LOCATION.

changes the value it contains.

STORING INTO A VARIABLE DOESN'T CHANGE THE VARIABLE; STORING
in the variable.

COPYING OR USING A VARIABLE'S VALUE DOES NOT CHANGE THE VALUE

\[ N = 2 \times N; \] \n\[ \text{N's value is now } 6. \text{ N's value changed. N did NOT.} \]

\[ N = 3; \] \n\[ \text{N's value is now 3. N is the same.} \]
\[ \text{int N; } \] \n\[ \text{N is created. We cannot predict its value.} \]
mathematically speaking, useless invariants are still called
useful invariants are about states/values of program variables, but

must be TRUE.

What the invariant says
statement after the invariant’s spot, and is just ABOUT TO execute the
before the invariant’s spot, and is just finished executing the statement

Whenever the computer has just finished executing the statement

An invariant is correct, or true, or is not a lie means:

program.

An invariant must have a PARTICULAR SPOT in a computer

and a predicate (e.g., “is now 3”).

and a predicate (e.g., “is now 3”).

and a predicate (e.g., “is now 3”).

An invariant is kind of expression of a thought (in your mind) by a

An invariant is NOT a kind of programming language statement.

called invariants

called invariants

Sentences like “N’s value is now 3” and “N’s value is now 6” are
form of useful pre and postconditions about the function, interfaces by writing comments into the header file that are in the

One process requirement in this course is to DOCUMENT function

Instead, study the CONTRACT on page 7.

Stay tuned whoops.

Super power-point slides on pre and post CONDITIONS.

And now for something completely different—Main and Savage’s

postconditions.

interfaces by writing kinds of invariants called preconditions and

Main and Savage chapter 1 covers how to document function