CSE 142 Summer 2001

Programming as Communication

Programming as Communication

- With a program, we communicate with two important entities:
  - machines
  - people
- The first is obvious, the second may not be:
  - Programs that don't work (bugs)
  - Inherited code
  - Program evolution

Reading vs. Understanding

- People and machines are very different.
  - Machines are good are reading but bad at *understanding*
  - People are good at understanding but bad at *reading*
- Read this:

is our children learning?

- What does it ask?
- What does it mean?

Metaphor: Programs as instructions

- Operational world view: a program is a set of instructions that brings about some action.
- Imagine giving directions:
  - To another student
  - To a tourist
- The student operates at a higher level of *abstraction*
- Good instructions (at any level of abstraction) require precision.
- Programming languages let you build new abstractions: it's like adding new (more meaningful) words to your language

Metaphor: Programs as Math

- We can think of programs as *executable math*:
- Consider:

Area = PI x Radius^2

- We can employ such expressions in programs.
- Most of our intuitions and knowledge about mathematics apply to computers.

Metaphor: Programs as Simulations

- Programs *model* our world or imaginary worlds
- We can define things in our programs that model the things in our world. We call these things *objects*.
- We can suspend the laws of physics in programs
- Programs are *plastic*: they are easy to mold to our wishes
- The limit of plasticity: big programs become as hard to manipulate as real-world entities. Why?