
Programming as Modeling

CSE 142, Summer 2003
Computer Programming 1

<http://www.cs.washington.edu/education/courses/142/03su/>

Readings and References

- Reading
 - » Chapter 2, *Intro to Programming and Object-Oriented Design Using Java*, Niño and Hosch

What is Computer Science?

- What do you think it is?

Metaphor: Programs as Math

- We can think of programs as *executable math*
 - » a program calculates some result for us
- Consider:
$$Area = \pi \cdot Radius^2$$
- We can employ such expressions in programs.
- Most of our intuitions and knowledge about mathematics apply to computers.

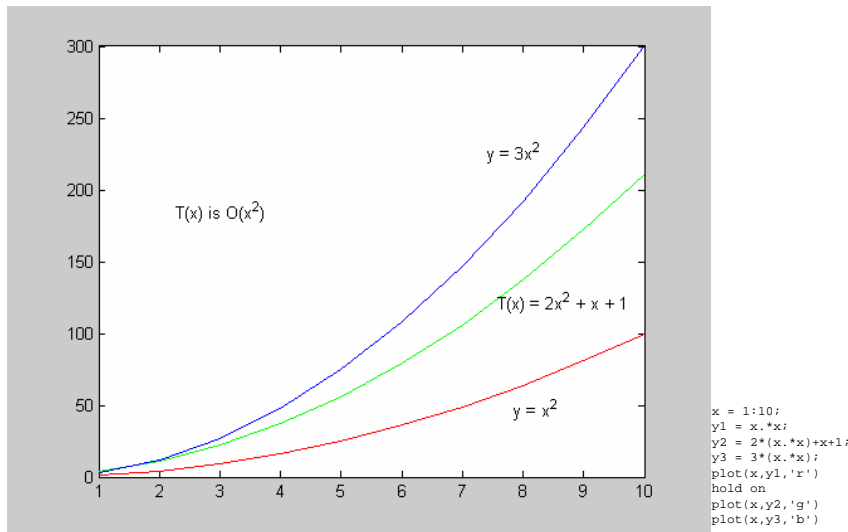
Metaphor: Programs as Directions

- Imagine giving campus directions:
 - » To another student
 - » To a tourist
 - » To a robot
- The student operates at a higher level of *abstraction* with a richer *vocabulary* of *shorthands* than the others
- An *algorithm* is a plan for how to accomplish a task
 - » A *program* is a software implementation of an algorithm
- Good algorithms (at any level of abstraction) require precision

23-June-2003

cse142-02-modeling © 2003 University of Washington

6



Using the program Matlab to calculate and plot function values

Metaphor: Programs as Simulations

- We also can think of programming as creating or simulating both *real* and *virtual* worlds.
- We can define things in our programs that *model* the things in our world. We call these things *objects*.
- Programs are *plastic*: they are easy to mold to our wishes
 - » Can be free of the constraints of real life!
- The limit of plasticity: big programs become as hard to work with as real-world entities

23-June-2003

cse142-02-modeling © 2003 University of Washington

7

What is Computer Science?

- Computation?
- Abstraction?
- Software Development?
- A useful viewpoint: think of it as *modeling*
 - » the model may be of something specific like a car or a weather system
 - » the model may be of something abstract like social interactions or a logical system of equations

23-June-2003

cse142-02-modeling © 2003 University of Washington

8

Overview of CSE 142

- We'll learn how to model...
 - » Objects
 - » Relationships
 - » Data
 - » Computation
- While learning how to create software systems using the Java programming language

Modeling: what? why?

- What are some examples of models?
- Why do we use models of objects?

Modeling: how?

- How do we model objects?

World of Objects

- Learn to think about *properties* and *responsibilities* of objects
 - » Properties: information relevant to the object
 - » Responsibilities: tasks an object performs
- Learn to think about how objects relate to each other in a system

Student Example: Properties

- Let's model a student object in the context of a course registration system
- Name examples of relevant properties:

Student Example: Responsibilities

- What are some responsibilities (tasks) for the student in a course registration system?

Retail Store Example

- Let's model a system to store inventory of a retail store that sells men's and women's shoes
- Give some examples of objects in the system

Retail Store Components

Object Properties Responsibilities

Summary

- Modeling the world with objects in software is called object-oriented programming.
- Objects have properties and responsibilities
- Modeling includes developing abstractions for objects and creating relationships among objects