

Administrivia

Slide 1

- One purpose of the syllabus is to spell out policies, especially about:
 - Course requirements and grading.
 - Exam dates (can only be changed if you all agree). “Please plan accordingly” means “don’t schedule something else for these dates”.
 - Late work.
 - Academic integrity.
- Most other information will be on the Web, either on my home page (office hours) or the course Web page.

More Administrivia

Slide 2

- Part of my job is to answer your questions outside class, so if you need help, please ask! in person or by e-mail or phone.

Slide 3

What Is This Course About?

- Improve programming skills.
- Understand “object-oriented” paradigm.
- Learn (more) basic concepts — data structures, etc.
- Along the way — learn Java, use IDE.

Slide 4

A Few Words About Computer Use in Class

- Checking your e-mail when you first get here is okay.
- Taking notes online is okay.
- Surfing the Web or playing games during lecture is not okay.
- Remember that I can lock all screens, project what’s on one student’s screen, etc. — and I will if need be. But I’d rather you’d all be responsible enough to resist this distraction!

“Object Orientation”?

Slide 5

- A “programming paradigm” — contrast with procedural programming, functional programming, etc.
- No accepted-by-all definition, but most definitions mention encapsulation:
 - Data and functionality grouped together into “objects”.
 - Some data/functionality is hidden.
- Origins in simulation/modeling, where the goal is to model complex systems consisting of many (real-world) objects.
- (More about this next week.)

What’s An Object?

Slide 6

- Object — set of data (attributes) and associated functions (methods, behaviors, operations) that can act on data.
- Objects interact by calling each other’s methods, or by sending each other messages.
- Often makes sense to have many similar objects — hence “classes”.
- (More about this next week.)

What's a Class?

- Can be thought of as a blueprint for objects of a given type; individual objects are "instances" of the class.
- Defines attributes and methods each object will have (instance variables/methods), attributes and methods shared by all objects of a class (class variables/methods).
- Public interface — attributes and methods visible from outside the class.
- (More about this next week.)

Slide 7

The Course Programming Project

- Write an arcade-style game.
- Build on "game infrastructure" (a.k.a. "Lewis Magic Cloud").

Slide 8

Game Basics

- “Player” — human-controlled moving entity.
- “Screens” — two-dimensional grids, make up playing field, side view or top view, can be linked together.
- “Blocks” — components of “screen” grids.
- “Game entities” — program-controlled entities, stationary or moving.

Slide 9

Minute Essay

- Tell me about your background:
If you took CSCI 1320 at Trinity, when and with what professor?
If not, what programming classes have you taken (high school or other), and what language(s) did you use?
Have you had any exposure to Java?
- What kind of computer do you think you will probably use for your homework (lab machine, home machine, etc. — if a home machine, what operating system)?
- What are your goals for this course?

Slide 10