Introduction to CS111

Tuesday, Sep. 4, 2007

CS111 Computer Programming

Department of Computer Science
Wellesley College

Am I in the right class?

- CS111 introduces the fundamentals of programming and problem solving techniques using Java.
- More advanced concepts are taught in CS230, Data Structures.
- CS110, Computer Science and the Internet, teaches web design.
- CS112, Computation for the Sciences, teaches programming skills using MATLAB

Please register for lab section*

- Labs provide hands on experience with the ideas presented in lecture.
- Labs meet Wednesdays at 8:30 -- 10:20, 10:30 -- 12:20, and 2:15 -- 4:05.
- If you cannot get into the section you want, register for another and use Q&A to arrange a swap.

*This does **not** satisfy the Wellesley laboratory requirement.

You needn't go to the bookstore

- All course materials may be found at

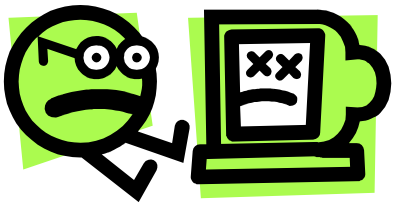
  http://cs.wellesley.edu/~cs111

- The CS file server, cs.wellesley.edu, is used to download and upload programming assignments.*

*Assignment 0, due by 4pm today, gets you an account. Lab 1 shows you how to use it.
Assignment 1, due on Tuesday, September 11

Help

- Q&A and announcements will be posted on FirstClass: Wellesley Conferences /Courses/CS/CS111-F07.
- Lecturers, lab instructor, and TAs have office/drop-in hours that are posted on the course web site.
- We'd like to get to know you. Come see us, even if you don't have a question.
- PLTC tutors are available to those who want them (at no charge to you).



Most importantly, talk to each other. But…

- All discussion should be in a high level language*. In particular, do not share code (we consider this a violation of the Honor Code).
- Do not consult materials from previous semesters (this also violates the Honor Code).

*Like English.

What is Computer Science?

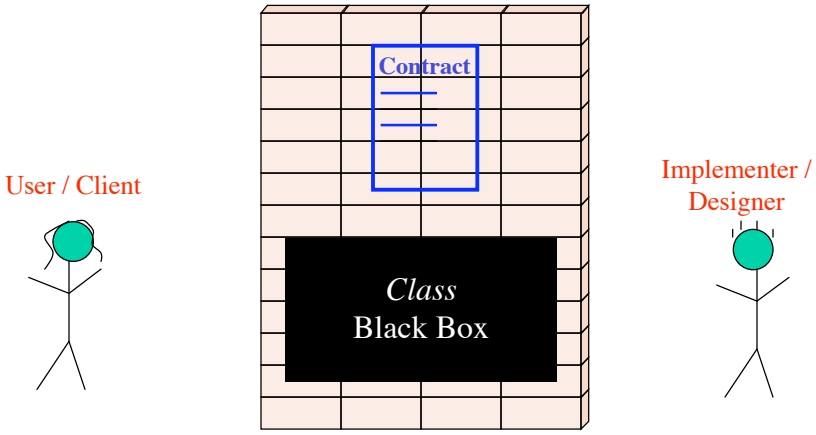
- It's not really about computers .
- It's not really a science.
- It's about imperative ("how to") knowledge as opposed to declarative ("what is") knowledge.
- Imperative knowledge is expressed via algorithms = computational recipes.
- "A computer language … is a novel formal medium for expressing ideas about methodology, not just a way to get a computer to perform operations. Programs are written for people to read, and only incidentally for machines to execute"

  -- Harold Abelson and Gerald J. Sussman

Four big ideas

- Four important concepts are at the core of this course:
  1. Abstraction;
  2. Modularity;
  3. Divide, Conquer and Glue;
  4. Models
- These interrelated ideas are important in almost every discipline, but they're at the core of CS.
- We will illustrate these ideas in several ways.
- Our goal is to help you think about problem solving in new ways.

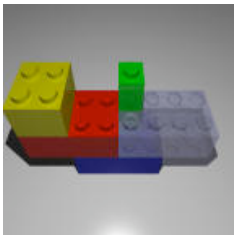
Big idea number 1: Abstraction



*Visit http://cs.wellesley.edu/~cs111/contracts for some useful Java contracts, which are known as Application Programming Interfaces (APIs).

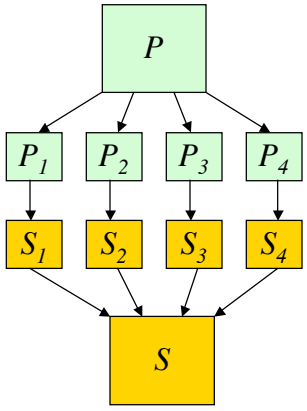
Big idea number 2: Modularity

- Large systems are built from components called modules.
- The interfaces between modules are designed so they can be put together in a mix-and-match way.
- In Java, goal is to design classes for maximum reusability.



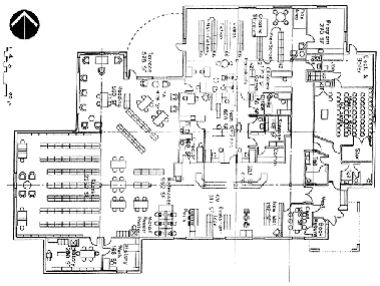
Big idea number 3: Divide, conquer & glue

Divide
problem P into subproblems.
Conquer
each of the subproblems, &
Glue (combine)
the solutions to the subproblems into a solution S for P.



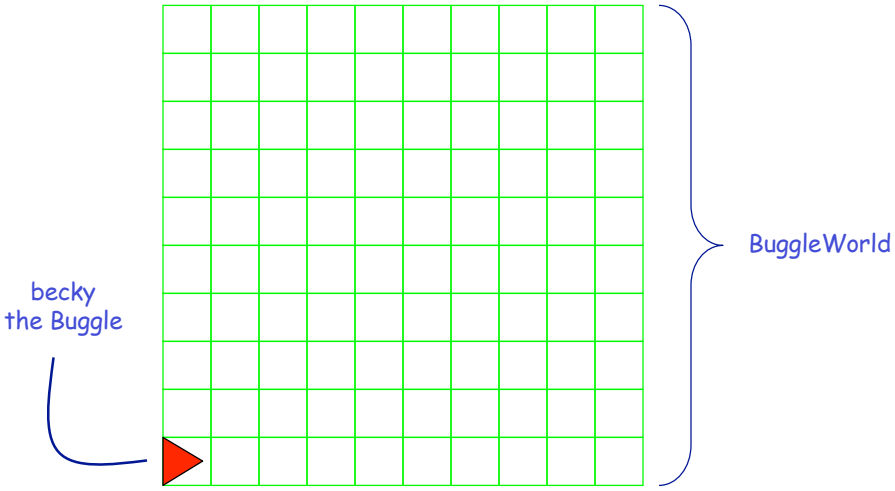
Big idea number 4: Models

- Need simple models to understand complex artifacts and behaviors.

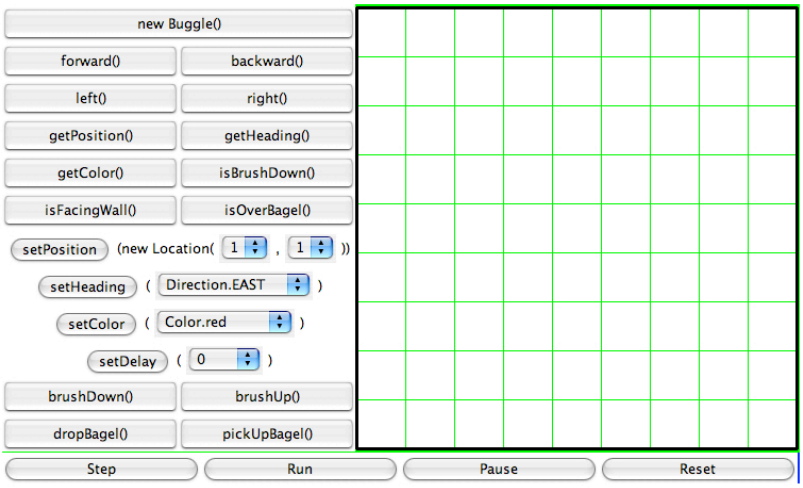


- Throughout this course, we will use a Java Execution Model (JEM) to explain what happens when Java code is executed.

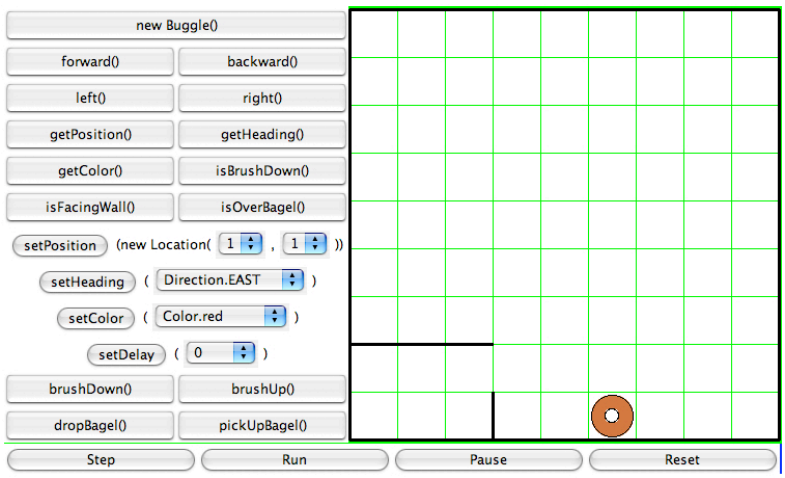
A world of problems: BuggleWorld



BuggleWorld

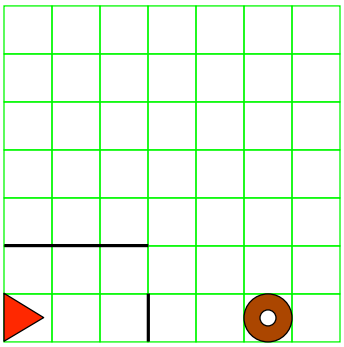


Bagels for Breakfast



Becky buys a bagel

```
public class BreakfastWorld extends BuggleWorld
{
  public void run ()
  {
        Buggle becky = new Buggle();
        // becky goes outside
        becky.forward(2);
        becky.left();
        becky.forward();
        becky.right();
        becky.forward();
        becky.right();
        becky.forward();
        becky.left();
        // walks to the bagel
        becky.forward(2);
        // and chows down
        becky.pickUpBagel();
  }
}
```



Taking the high road

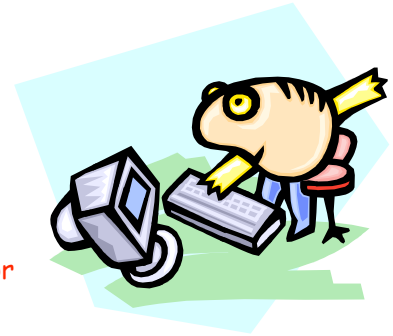
- The "native" language of a computer is a low-level language. E.g.,

```
# Store the sum of A and B in C
load r4, A
load r5, B
add r4, r5
store r4, C
```

- Java is a high-level language designed for people. E.g.,

```
// Store the sum of A and B in C
C = A + B;
```

- To get from high to low a translator is needed.

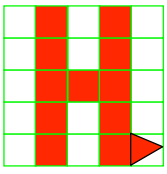


Interpreters

```
public void run ()
{
     Buggle ben = new Buggle();
     ben.left();
     ben.forward(4);
     ben.backward(2);
     ben.right();
     ben.forward(2);
     ben.left();
     ben.forward(2);
     ben.backward(4);
     ben.right();
     ben.forward();
}
```

Source code

Interpreter



Results

Compilers

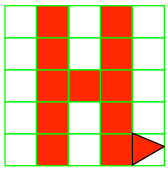
```
public void run ()
{
     Buggle ben = new Buggle();
     ben.left();
     ben.forward(4);
     ben.backward(2);
     ben.right();
     ben.forward(2);
     ben.left();
     ben.forward(2);
     ben.backward(4);
     ben.right();
     ben.forward();
}
```

Source code

Compiler

```
01110100101
01001111010
00100011110
11010001011
00010001100
00100111100
00000111111
01111110011
```

Object code



Results

Object code interpreter

Java does both

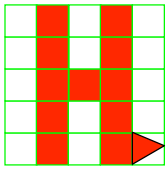
```
public void run ()
{
     Buggle ben = new Buggle();
     ben.left();
     ben.forward(4);
     ben.backward(2);
     ben.right();
     ben.forward(2);
     ben.left();
     ben.forward(2);
     ben.backward(4);
     ben.right();
     ben.forward();
}
```

Source code (.java)

Compiler

```
jtklew0er#2
^720[l8scWq
kls;wkjjh3?
nnmsllw7y0*
y%#*&jk23=}
(*kd1*8,<vV
p+}ke56&8)6
kls;wghjh3?
```

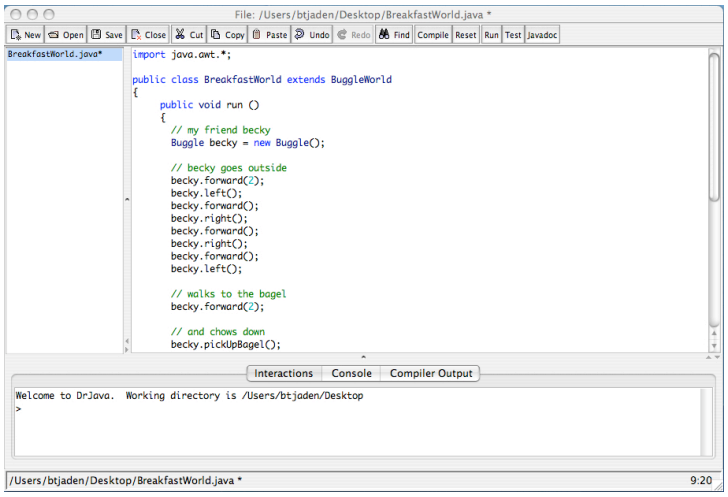
Java byte codes (.class)



Results

Interpreter
(JVM = Java Virtual Machine)

DrJava helps you both edit and compile



Beyond Buggles

We will use several "microworlds" other than BuggleWorld for problem-solving activies. You can see examples of these at:

http://cs.wellesley.edu/~cs111/museum.html

These worlds include:

- PictureWorld
- TurtleWorld
- Java Graphics
- AnimationWorld