Buggles, bagels & breakfast

Getting started

Tuesday, Sep. 5, 2006

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 Thursdays 1:30 -- 3:20.
- 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.

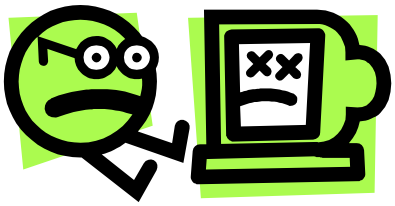
You needn't go to the bookstore

- 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 the end of the day, gets you an account. Lab 1 shows you how to use it.

Help

- Q&A and announcements will be posted on FirstClass:
  Wellesley Conferences /Courses/CS/CS111-F06.
- 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).

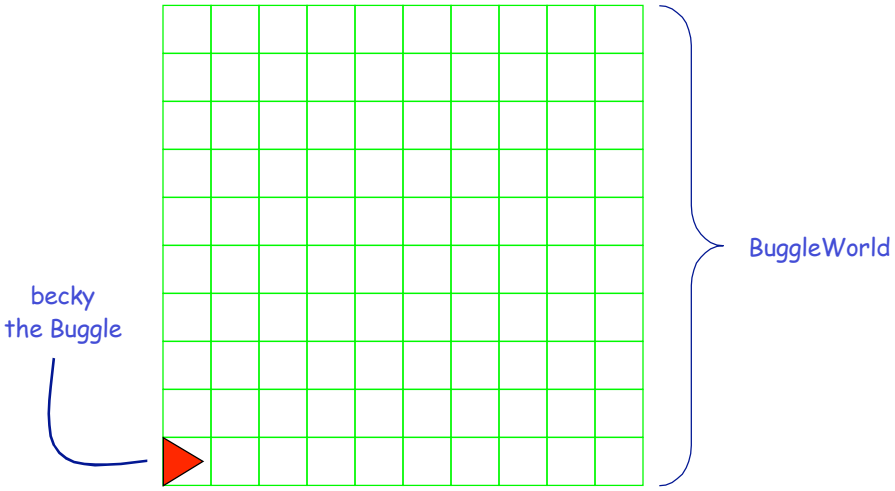


Most importantly, talk to each other. But…

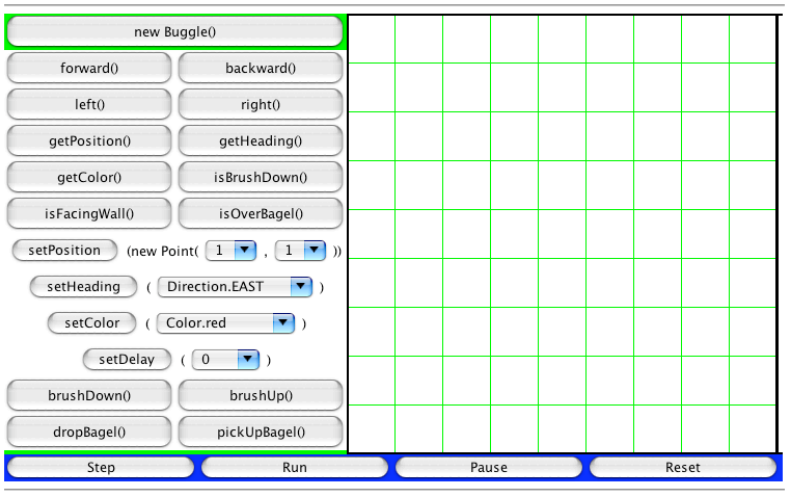
- All discussion should be in a high level language*. In particular, do not share code (we consider this a violation of the Honor Code).
- Do not consult materials from previous semesters (this also violates the Honor Code).

*Like English.

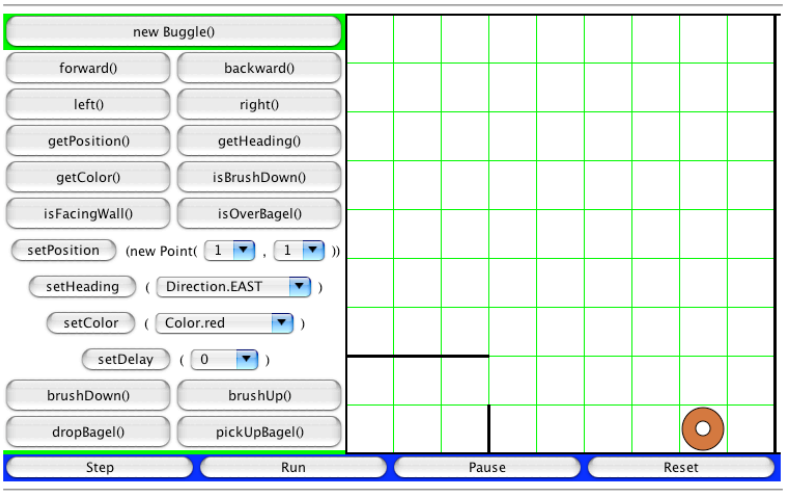
A world of problems: BuggleWorld



BuggleWorld

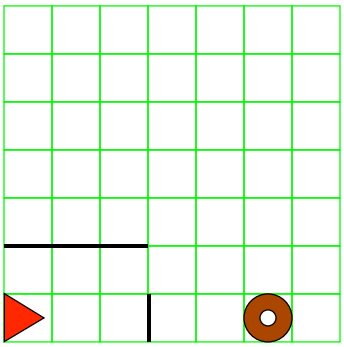


Bagels for Breakfast



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

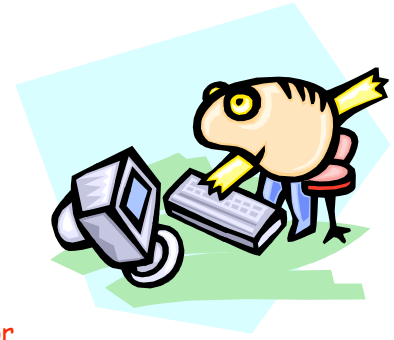
- The "native" language of a computer is a low-level language. E.g.,

```
# 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.

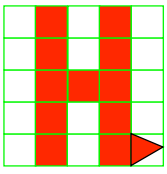


Interpreters

```
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

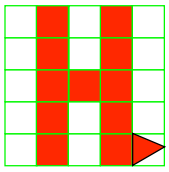
```
public void run ()
{
     Buggle ben = new Buggle();
     ben.left();
     ben.forward(4);
     ben.backward(2);
     ben.right();
     ben.forward(2);
     ben.left();
     ben.forward(2);
     ben.backward(4);
     ben.right();
     ben.forward();
}
```

Source code

Compiler

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

Object code



Results

Object code interpreter
(e.g., a Pentium chip)

Java does both

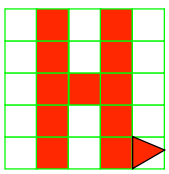
```
public void run () {
     Buggle ben = new Buggle();
     ben.left();
     ben.forward(4);
     ben.backward(2);
     ben.right();
     ben.forward(2);
     ben.left();
     ben.forward(2);
     ben.backward(4);
     ben.right();
     ben.forward();
}
```

Source code (.java)

Compiler

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

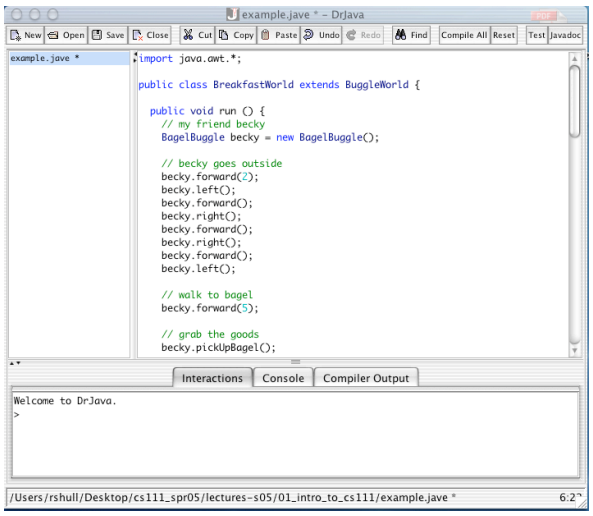
Java byte codes (.class)



Results

Interpreter
(JVM = Java Virtual Machine)

helps you both edit and compile



Applets vs. Applications

BreakfastWorld is an example of an applet: a Java program that can be executed by a web browser. It's easy to invoke applets from web pages. E.g.,

```
<html>
  <head>
    <title>Six a.m. at Turtle Lane</title>
  </head>
  <body>
    <applet code="BreakfastWorld.class" width=750 height=450></applet>
  </body>
</html>
```

- An application is a stand-alone Java program that can be executed without a web Browser. E.g., it can be executed directly in Dr. Java.
- We will start the semester with applets, but you will write some applications later in the course.

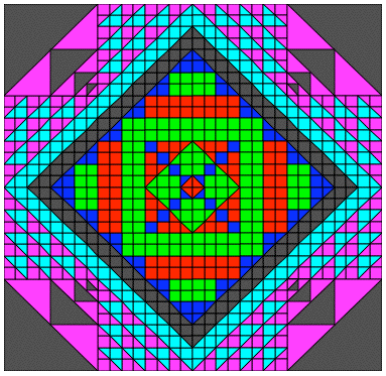
Beyond Buggles

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



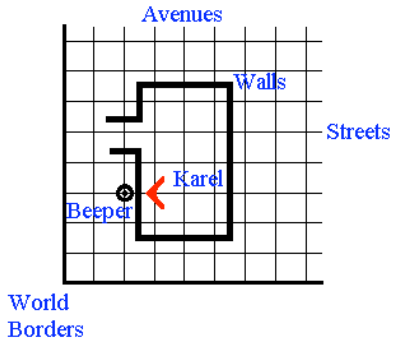
Buggle Ancestry

- "Floor turtles" used to teach children problem solving in late 1960s. Controlled by *LOGO* programming language created by Wally Feurzeig (BBN), Daniel Bobrow (BBN), and Seymour Papert (MIT).

- Logo-based turtles introduced around 1971 by Papert's MIT Logo Laboratory.
- Turtles play a key role in "constructionist learning" philosophy espoused by Papert in *Mindstorms* (1980).

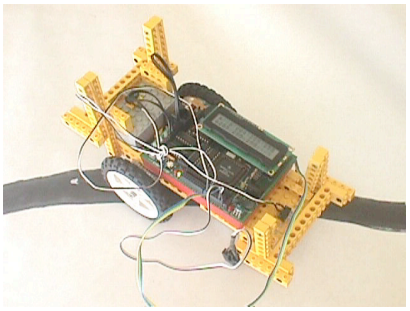
Buggle Ancestry (cont'd)

- Richard Pattis's Karel the Robot (1981) teaches problem-solving using Pascal robots that manipulate beepers in a grid world.
- *Turtle Geometry* book by Andrea diSessa and Hal Abelson (1986).
- LEGO/Logo project at MIT (Mitchel Resnick and Steve Ocko, 1988); evolves into Handyboards (Fred Martin and Brian Silverman), Crickets (Robbie Berg @ Wellesley), and LEGO Mindstorms
- StarLogo – programming with thousands of turtles in Resnick's *Turtles, Termites, and Traffic Jams* (1997).

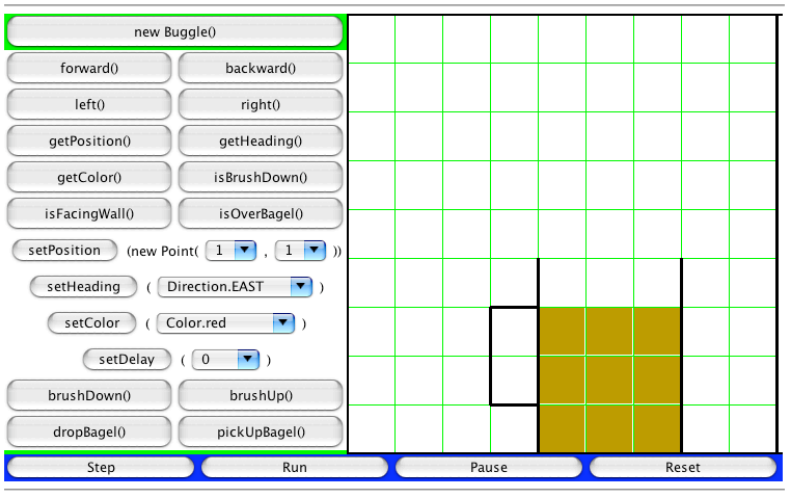


Turtles, Buggles, & Friends At Wellesley

- In mid-1980s, Eric Roberts teaches programming using software-based turtles.
- In 1996, Robbie Berg and Lyn Turbak start teaching Robotic Design Studio with Sciborgs.
- In 1996, Randy Shull and Takis Metaxas use turtles to teach problem solving in CS110.
- In 1997, BuggleWorld introduced by Lyn Turbak when CS111 switches from Pascal to Java.
- In 2006, Robbie Berg and others introduce PICO Crickets: http://www.picocricket.com



Cup of Joe



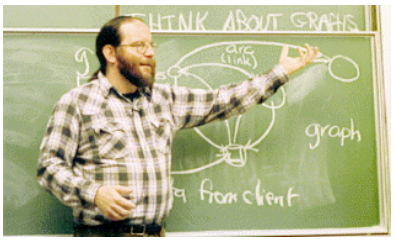
Long ago in a place not far from here …

There was a little turtle
Who lived in a box.
He swam in the water,
He climbed on the rocks.

He snapped at a mosquito.
He snapped at a flea.
He snapped at a minnow
And he snapped at me.

He caught the mosquito.
He caught the flea.
He caught the minnow.
But he didn't catch me.

Little Turtle
Vachel Lindsay



Fast forward

- Buggles are Java objects that live in BuggleWorld.
- They are part of a program written by Franklyn Turbak to teach problem solving and Java programming.

*Alas, we have no poem dedicated to the Buggle.