Reaching Computer Clubs with Computing Concepts

using Scratch, Alice, Greenfoot and JavaFX

Daniel Green
Sun Microsystems
Kansas City Computer Club
http://pesced.ning.com

Wednesday, June 17th 2009
11:40am - 12:00 pm
Fitzpatrick Center, Duke University
Touch Points

- About Computer Clubs
- Programming Concepts and Tools Used
  - Scratch, Alice, Greenfoot and JavaFX
- Moving Concepts Between Tools
- Reaching Computer Clubs
Some Observations

* I grew up playing video games...
* I like playing video games, probably too much...
* My kids like playing video games...
* Their friends and their friends’ friends like playing video games...
* Video games are created by people writing software... but we know there’s a problem here...
An Inconvenient Truth?

Figure 1. Computer Science Listed as Probable Major Among Incoming Freshmen
Source: HERI at UCLA

NO VIDEO GAMES
An Inconvenient Reality

- Computer Science decline would also mean...
- No Amazon Kindle
- No Blu-ray Disc (Sorry, Neil Young)
- No Mobile Phones (JAVA in 80% of them)
- No Google Maps
- No LiveScribe Pen (JAVA ME in pen)
- No Ricoh Printers (JAVA ME CDC)
- No Mars Rover
About This Talk:

Students in the current K-12 environment often are exposed to computer science in this manner: “learn to type, learn Microsoft Word, learn Microsoft Powerpoint”. This approach teaches basic computer appreciation or computer operation and does not equip students with the powerful ideas that underly computer science. Computer Club is a series of local workshops open to the public, focusing on improving digital literacy of students by empowering them to create projects involving computer graphics, animation, video, sound, gaming, programming in a monthly instructor led setting. The target age for Computer Club workshops are 9-16 years old. The goal of computer club outreach is to volunteer and work with students on creating interesting projects that teach them the underlying computer science concepts, and equip them with tools they can continue to use at home, work, and school, and focus on free, open source and multi-platform software tools when available. Computer Club on a Stick is an approach to providing this environment on a USB memory stick so that students can continue building on the projects they create during class using tools provided on the memory stick.

This session will cover project approaches and ideas for integrating programming concepts using Alice, Java, JavaFX, Greenfoot and BlueJ. Programming project concepts include sequence, iteration, conditionals, variables, threads, synchronization, boolean logic, random numbers, event handling, user interface interaction, data structures, procedures and functions, recursion, inheritance and classes, and parameters. Resources for inclusion on memory sticks will be covered during the session.

---

Computer Club on a Memory Stick

Bring Java + Friends to Life

Presenter: Daniel Green

Presenter Bio:
Daniel Green is a Senior System Engineer for Sun Microsystems Inc, specializing in Java ME and mobile ecosystem solutions for wireless carriers. He is also one of several volunteer teachers and coaches in his area supporting Computer Club, FIRST Lego League and other student activities that encourage math, science and programming activities to engage students in learning.

Daniel Green has used Java, Alice, Greenfoot, BlueJ, Squeak, Scratch and other tools to teach K-12 students and college workshops on introduction to programming. His students regularly create projects with these tools as a way of learning basic programming concepts.

About Computer Club

Building Digital Literacy in the Clubhouse

Be Subversive

LET THEM HAVE FUN MAKING VIDEO GAMES

About Computer Club

Building Digital Literacy in the Clubhouse

Be Subversive

LET THEM HAVE FUN MAKING VIDEO GAMES
What is Computer Club?

* Teaching students 9 yrs - 16 yrs old
* Getting past “Computer Application Training”
* Create projects involving animation, sound, graphics, gaming, etc. - things kids like
  * `System.out.println("Hello World!");`
* Use free and multi-platform tools such as Scratch, Alice, Greenfoot...
* Project competition and prizes
Computer Club Examples

* MIT - LiFELONg KiNDERGARTEN
  * http://llk.media.mit.edu/

* Intel - COMPUTER CLUBHOUSE
  * http://www.computerclubhouse.org/

* Many others less formal

* Many online resources
  * http://www.dickbaldwin.com/
Where is Computer Club?

- After school, in the school computer lab
- On weekends, in university computer labs
- FIRST Lego League Teams - off season
- Anywhere volunteers can get access to enough computers to support the number of students interested...
“Although computer science is an established discipline at the collegiate and post-graduate levels, its integration into the K-12 curriculum has not kept pace in the U.S. As a result, a serious shortage of information technologists exists at all levels. (NO VIDEO GAMES!)

The second edition of the ACM Model Curriculum sets the context for computer science within K-12 education today and provides a framework for state departments of education and school districts to address the educational needs of young people and prepare them for personal and professional opportunities in the 21st century.”

http://www.csta.acm.org/Curriculum/sub/ACMK12CSModel.html

* wonderful resource...
* from us, by us...
* for typing and Word?
“Why on Earth would we want to do that, in an era of glossy animation-rendering engines, game-design ogres and sophisticated avatar worlds? Because if you want to give young students a grounding in how computers actually work, there's still nothing better than a little experience at line-by-line programming.

Only, quietly and without fanfare, or even any comment or notice by software pundits, we have drifted into a situation where almost none of the millions of personal computers in America offers a line-programming language simple enough for kids to pick up fast. Not even the one that was a software lingua franca on nearly all machines, only a decade or so ago. It is a problem for our nation and civilization.” -- David Brin, salon.com, 9/14/2006
Engage with Java Technology Learning Path

**TEACHING CONCEPTS**

- Sequence
- Iteration,
- Conditional Logic
- Variables
- Data Structures (dynamic lists)
- Events Handling
- Parallel Execution
- Synchronization
- Random Numbers
- Boolean Logic
- Dynamic Interaction
- User Interface Design
- Publish projects as Java applets on scratch.mit.edu

**FEATURES**

- Simplest tool
  Ages 5-15
  - Drag and drop code blocks
  - 2D graphics frameworks interaction

- Simpler tool
  Ages 8-22
  - Drag and drop code blocks
  - 3D graphics frameworks interaction

- Less simple
  Ages 12-22
  - Interactive interpreter code typing
  - Media computation helper classes

- Less simple
  Ages 13-25
  - Type, compile, run, debug
  - 2D gaming framework interaction via 5 Java classes

- Less simple
  Ages 15-25
  - Type, compile, run, debug
  - No default graphics framework environment

- Complex tool
  Ages 15-25
  - Type, compile, run, debug
  - No default graphics framework environment

- Complex tool
  Ages 16+
  - Type, compile, run, debug
  - No default graphics framework environment

**TOOLS**

- Simplest tool
  Ages 5-15
  - Scratch
  - Imagine, program, share

- Simpler tool
  Ages 8-22
  - Alice
  - Dr. Java

- Less simple
  Ages 12-22
  - Greenfoot
  - Media Computation.org

- Less simple
  Ages 13-25
  - BlueJ

- Less simple
  Ages 15-25
  - NetBeans

- Complex tool
  Ages 15-25
  - Java

- Complex tool
  Ages 16+
  - No/Any tool

**All Scratch concepts plus..**

- Procedures and Functions
- Parameter Passing & Return Values
- Recursion
- Defining Classes of Objects
- Inheritance
- Text Input

- All previous Alice concepts, excluding user interface design
- Manipulation of audio, images, video through media computation helper classes – see mediacomputation.org
- All Java Programming Language Features Available
- Interpreted mode is great step from drag and drop code blocks (no errors possible)

- All previous Alice concepts plus...
- 5 Java classes encapsulate 2D gaming and simulation concepts
- All Java Programming Language Features Available
- Interpreted mode is great step from drag and drop code blocks (no errors possible)

- All previous Alice concepts plus...
- 5 Java classes encapsulate 2D gaming and simulation concepts
- All Java Programming Language Features Available
- Interpreted mode is great step from drag and drop code blocks (no errors possible)

- All previous Alice concepts plus...
- 5 Java classes encapsulate 2D gaming and simulation concepts
- All Java Programming Language Features Available
- Interpreted mode is great step from drag and drop code blocks (no errors possible)

- All previous Alice concepts plus...
- 5 Java classes encapsulate 2D gaming and simulation concepts
- All Java Programming Language Features Available
- Interpreted mode is great step from drag and drop code blocks (no errors possible)
# Computer Club Tools Used

<table>
<thead>
<tr>
<th>Tools</th>
<th>sequence</th>
<th>loop</th>
<th>conditionals</th>
<th>variables</th>
<th>threads</th>
<th>synchroniz</th>
<th>booleans</th>
<th>random nu</th>
<th>event handl</th>
<th>UI inter</th>
<th>data struct</th>
<th>proc funct</th>
<th>recusion</th>
<th>classes</th>
<th>param</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scratch</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Alice</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Greenfoot</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>BlueJ</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Dr. Java</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Squeak</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>SeaMonkey</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>JavaFX</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>
Tools Used - Scratch

* Drag and Drop 2D Programming
* “Low Floor, High Ceiling, Wide Walls”
* Math / Geometry / Algebra Projects
* Demo “Little Crab” scenario from Greenfoot
* Social network aspect of http://scratch.mit.edu is very motivating
**Tools Used - Alice**

- Drag and Drop 3D Programming
- Targeted at middle school girls
- Story / Movie Projects
- Alice 3 High Interest Points:
  - Electronic Arts SIMs assets
  - “Show me the code”
  - Demo “Plane” scenario from Alice
Tools Used - Greenfoot

* Type / Compile / Run / Debug 2D Programming
* Beginner tool for Java Programming
* 5 Java classes encapsulate Java 2D Gaming
  * Actor, World, Greenfoot, GreenfootImage, MouseInfo
* Micro-World, Games, Simulation Projects
* Demo "Little Crab", "Wave Lab", "Gravity", "Marbles Game", "Greeps Contest" scenarios
Tools Used - BlueJ

* Type / Compile / Run / Debug 2D Programming
* Simple Interface
* Good visualization of classes / objects
* BlueJ plugin for NetBeans

“for students who wouldn’t be interested in making the jump otherwise...”
Tools Used - Dr. Java

* Lightweight Java IDE
* Referenced in “Media Computation” book
* Interpreted mode is easy for students
* free and multiplatform at http://drjava.org
Tools To Be Used - JavaFX

- Common student request:
  - “Let’s make Flash games”
  - Flash != $0
  - Flash programming less than desirable
- JavaFX focuses on RIA
  - graphics, sound, video, media
- You likely haven’t heard much about it yet
RIA Evolution to date: Convergence of Rich Client and Web

Initial Deployment

Netscape
Flash
Air
Silverlight
Chrome

Rich Client
Browser


RIA Experience

Sun Confidential: Internal Only
Rich Internet Application Frameworks

- **Air + Flash**
- **Silverlight**
- **Opera**
- **JavaFX**

**Text, Graphics**

**Screens**

**Expressiveness**

**RIA Features**
JavaFX + Java Architecture

JavaFX Applications and Services

JavaFX App Framework

JavaFX Mobile Runtime
JavaFX Desktop Runtime
JavaFX TV Runtime

JavaFX Mobile Ext
JavaFX Desktop Ext
JavaFX TV Ext

JavaFX Common Platform

Java Platform (Java Plug In)

Authoring Tools

Developer Tools
(IDE Plug ins, RAD, Media Factory)

Designer Tools
(Authoring, Publishing, Media Encoding)
Developer Tool Chain

Media Assets Created By:
- Adobe Illustrator
- Adobe Photoshop
- On2 Flix Encoders (JavaFX File Format, VP6 and MP3)
- Adobe CS3 (Flash and Flash Video (JavaFX 1.5))

Assets Transformed By:

Integrated into IDEs:
- Sun JavaFX Plug-in for IDEs
- Sun JavaFX Compiler

Emulated by (if reqd):
- Sun JavaFX Mobile Emulator
-.eclipse (JavaFX 1.5)

3rd Party RAD Tool
# 2008-09 Major Milestones

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Platform</strong></td>
<td><strong>Tools</strong></td>
<td><strong>TV Deployment</strong></td>
<td><strong>Launch</strong></td>
<td></td>
</tr>
<tr>
<td>1.0 Launch</td>
<td>Mobile Readiness</td>
<td>Mobile Announcements</td>
<td>Designer Tool</td>
<td>TV Readiness</td>
</tr>
<tr>
<td>1.0 (10/31)</td>
<td>2.0</td>
<td>2.5</td>
<td>3.0</td>
<td></td>
</tr>
<tr>
<td>2D, Media, Animation</td>
<td>3D, Full HD Video</td>
<td>Physics Engine</td>
<td>Immersive 3D</td>
<td></td>
</tr>
<tr>
<td>SDK NetBeans Plugin Media Factory Media Encoder (On2)</td>
<td>RAD Tool</td>
<td>Designer Tool (Preview)</td>
<td>Designer Tool</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Launch</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11/10 JavaFX.com</td>
<td>MWC '09</td>
<td>JavaOne '09</td>
<td>MAX '09</td>
<td>CES</td>
</tr>
</tbody>
</table>
Resources Used

* Scratch
  * http://scratch.mit.edu
  * http://learnscratch.org

* Alice
  * http://alice.org

* “Programming With Alice” book

* Greenfoot.org / BlueJ.org

* New Greenfoot book, existing BlueJ book
# Computer Club Tools Needed

<table>
<thead>
<tr>
<th></th>
<th>sequence</th>
<th>loop</th>
<th>conditionals</th>
<th>variables</th>
<th>threads</th>
<th>synchroniz</th>
<th>booleans</th>
<th>random number</th>
<th>event handling</th>
<th>UI interface</th>
<th>data structures</th>
<th>functions</th>
<th>recursion</th>
<th>classes</th>
<th>parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scratch</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Alice</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Greenfoot</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>BlueJ</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Dr. Java</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Squeak</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>SeaMonkey</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>JavaFX</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>
Welcome to Programming with Duke & Friends,

Here are a few things you can do right now...

Recommended Links!

- Engage with Java Learning Path Image
- Change Your World – Engage with Java
- Scratch
- Dr Java
- Greenfoot
- Greenfoot Gallery
- BlueJ
- NetBeans
- New to Java Programming Center – Young Developers

Thread: Obtaining computers
Started by Ben Schafer in General Discussion, Last reply by DanG Mar 17.
Reaching Computer Clubs with Computing Concepts

using Scratch, Alice, Greenfoot and JavaFX

Daniel Green
Sun Microsystems
Kansas City Computer Club
http://pesced.ning.com

Wednesday, June 17th 2009
11:40am - 12:00 pm
Fitzpatrick Center, Duke University