

# Impact of Alternative Introductory Courses on Programming Concept Understanding

Allison Elliott Tew  
W. Michael McCracken  
Mark Guzdial

College of Computing  
Georgia Institute of Technology

## Overview

- § Research Question
  - Do different approaches in introductory CS really matter?
- § Methodology
  - Two alternative intro courses
  - Pre- and post-test in common second course
- § Results
  - Significant differences on pre-test
  - No significant differences on post-test
- § Discussion
  - Rethinking the CS 1 debate

## Research Questions

- § What is the impact of alternative pedagogical approaches to introductory computing?
  - What computing concepts do students understand when beginning their second course?
  - How does the introductory approach influence the learning outcomes of the second course?

## Georgia Tech Introductory Courses

Course	Content	Audience	Languages
'Computing'	Objects Early Simple data structures & algorithms	CS Majors	Python
'Engineering'	Objects Early Traditional CS content in engineering context	Engineering Majors	MATLAB & Java
'2nd Course'	Object-oriented programming Advanced data structures	CS & Engineering Majors	Java

## Methodology

- § Identified common topics
  - 'Introductory'
    - § Conditionals, arrays, binary search trees, searching, sorting, recursion
  - 'Advanced'
    - § OO Class definitions, polymorphism, dynamic binding, linked lists, hash tables, GUIs
- § Quasi-experimental study design
  - Pre-Test/Post-Test in 2nd course
    - § Introductory and advanced topics on both tests
  - Multiple choice question format (Lister et al 2004)
    - § Tracing questions
    - § Code completion

## Sample Tracing Question

```
int[] array1 = { 4, 5, 3, 6, 2, 7, 1 };
int[] array2 = { 7, 4, 2, 1 };

array1[ 3 ] = array1[ 5 ];
array1[ 2 ] = array2[ 2 ];
array1[ 4 ] = array2[ 3 ] + 5;
array1[ 6 ] = array1[ 3 ];
if ( array1[ 1 ] > array2[ 1 ] )
    array1[ 1 ] += 2;
```

Array

What is the value of array1 after this code is executed?

- a) { 4, 4, 2, 7, 2, 2, 1 }
- b) { 4, 7, 3, 6, 7, 7, 7 }
- c) { 4, 7, 2, 7, 6, 7, 7 }
- d) { 4, 7, 2, 7, 2, 2, 1 }
- e) { 4, 7, 2, 6, 6, 7, 6 }

## Sample Code Completion Question

The following method "isSorted" should return true if the array "x" is sorted in ascending order. Otherwise, the method should return false:

```

public static boolean isSorted(int[] x)
{
    //missing code
}

```

Which of the following code fragments is the missing code?

a) `boolean b = true;`  
`for (int i = 0; i < x.length - 1; i++)`  
`{`  
 `if (x[i] > x[i + 1])`  
 `b = false;`  
 `else`  
 `b = true;`  
`}`  
`return b;`

b) `for (int i = 0; i < x.length - 1; i++)`  
`{`  
 `if (x[i] > x[i + 1])`  
 `return false;`  
 `return true;`  
`}`  
`...`

Sorting

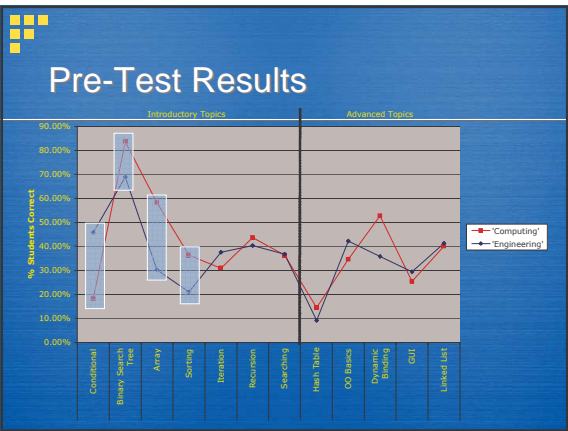
## Participant Information

§ 'Computing'

- Students who had completed the traditional intro CS course
- Pre-Test (n = 55)
- Post-Test (n = 31)
- Majors
  - § CS (55 - 75%)
  - § Electrical & Computer Engineering

§ 'Engineering'

- Students who had completed the engineering context intro CS course
- Pre-Test (n = 109)
- Post-Test (n = 48)
- Majors
  - § Electrical & Computer Engineering
  - § Industrial Engineering



## Conditional

§ Question

Traced through a sequence of Boolean and mathematical nested conditional statements to Calculate math operations on x and y

§ Results

- 'Engineering' Group understood this concept better than the 'Computing' Group
- § 45.87% of 'Engineering' students answered correctly
- § 18.18% of 'Computing' students answered correctly

## Conditional (continued)

§ Results (continued)

- 'Engineering' and 'Computing' both chose the same distracter most frequently
  - § Misunderstood scope of nested conditional statements
- 'Computing' also often chose an additional distracter
  - § Miscalculating first Boolean logic statement

§ Interpretation based on Course Analysis

- Scoping syntax more explicit in Java than in Python
- Boolean logic covered more in 'Engineering' lecture
- Connections to other coursework/concepts in engineering majors
  - § Flow-chart notation
  - § Logic in digital circuit design

## Binary Search Tree

§ Question

Given a class definition and a figure of a BST  
Trace code for a pre-order traversal

§ Results

- 'Computing' Group understood this concept better than the 'Engineering' Group
- § 83.64% of 'Computing' students answered correctly
- § 68.81% of 'Engineering' students answered correctly
- Both groups chose the same distracter most frequently
  - § Listed nodes read logically from left to right, top to bottom

§ Interpretation based on Course Analysis

- BST presented not in engineering context
- Issue of motivation

## Array

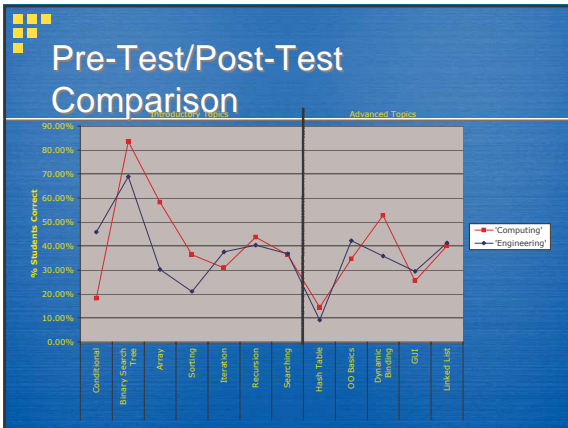
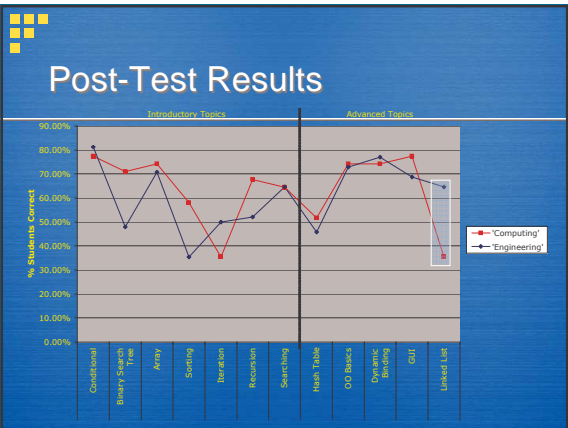
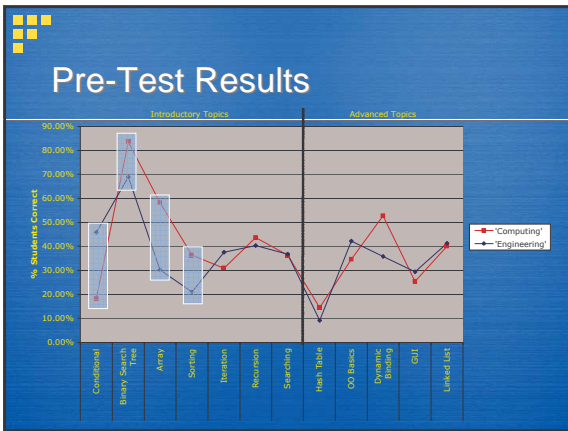
- § Question
  - Calculations and manipulations on two arrays
- § Results
  - 'Computing' Group understood this concept better than the 'Engineering' Group
    - § 58.18% of 'Computing' students answered correctly
    - § 30.28% of 'Engineering' students answered correctly
  - Majority of 'Engineering' students chose distracter
    - § Incorrectly indexed array beginning with 1
- § Interpretation based on Course Analysis
  - MATLAB vectors are indexed beginning with 1
  - Incorrect transfer from first language to Java

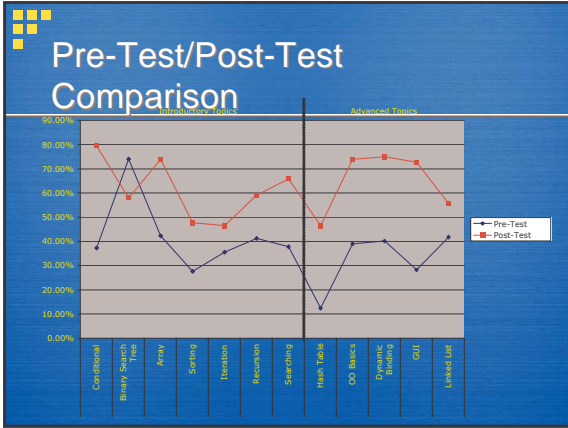
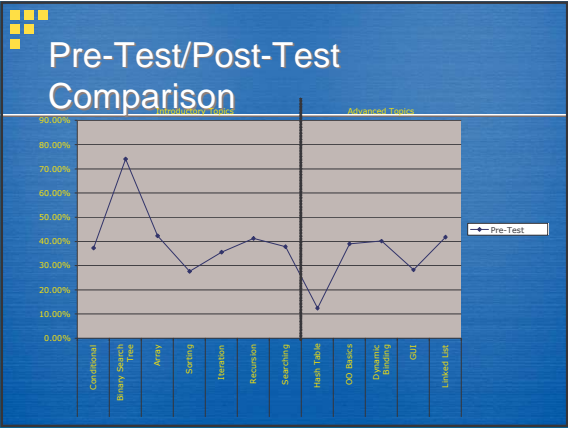
## Sorting

- § Question
  - Comparing value of elements in array for sorting algorithm
  - Select code to evaluate whether an array is sorted in ascending order
  - Replicated from Lister ITCSE study
- § Results
  - 'Computing' Group understood this concept better than the 'Engineering' Group
    - § 36.36% of 'Computing' students answered correctly
    - § 21.10% of 'Engineering' students answered correctly

## Sorting (continued)

- § Results (continued)
  - 'Computing' and 'Engineering' both chose the same distracter most frequently
    - § Correctly evaluated element comparison, but failed to aggregate evaluations
  - 'Engineering' also often chose an additional distracter
    - § Sorted in descending order
- § Interpretation based on Course Analysis
  - Findings are consistent with Lister et al
  - Sorting not connected to engineering problem solving context
  - Issue of motivation





### Discussion

- § After the introductory course, students have different levels of understanding of introductory concepts
- § Yet, these differences tend to disappear after a second common course
- § Conclusions
  - Students are continuing to learn introductory concepts in their second course
  - Other conceptual/procedural differences may appear later in the curriculum
  - Reconsider the debate about "what" to teach in CS1

### Future Work

- § Repeating pre/post-test study Fall Semester 2005

Course	Content	Audience	Languages
'Computing'	Objects Early Simple data structures & algorithms	CS Majors	Python
'Engineering'	Objects Early Traditional CS content in engineering context	Engineering Majors	MATLAB & Java
'Media Computation'	Objects Early Most traditional CS topics in media context	Liberal Arts Majors	Python

### Future Work (continued)

- § Repeating pre/post-test study Fall Semester 2005 (continued)
  - Gathering data on reliability
- § Additional qualitative component
  - Observing students taking the pre-test
  - Using think aloud protocol
- § Studying the 2nd course

### Acknowledgements

- § Undergraduate Research Student
  - Boris Goykhman
- § Faculty
  - Yannis Smaragdakis
  - John Stasko
  - Ashwin Ram
- § National Science Foundation
  - CCLI-ASA Program
  - CISE EI Program



Questions?