

Advanced Placement Computer Science: Meet the Committee

Judy Hromcik
Arlington High School, Texas
Member, AP Computer Science Development Committee
jhromcik@aisd.net

David Reed
Creighton University
Chief Reader of AP Computer Science
davereed@creighton.edu



APCS Development Committee



- Don Allen, Troy High School, California
- Scot Drysdale, Dartmouth College, New Hampshire (Chair)
- Reginald Hahne, Atholton High School, Maryland
- Cay Horstmann, San Jose State University, California
- Ann Shen, Bishop Strachan School, Toronto
- Laurie White, Mercer University, Georgia

- David Reed, Creighton University, Nebraska (Chief Reader)
- Frances Hunt and Dennis Ommert (ETS Consultants)

Retired in 2005:

- Judy Hromcik, Arlington High School, Texas
- Richard Kick, Hinsdale Central High School, Illinois
- Julie Zelenski, Stanford University, California

2005 Free Response Questions



A1: Hotel Reservation

- maintain an array of Reservation objects and an ArrayList serving as waiting list

A2: Ticket Sales (Design)

- design and implement classes in a hierarchy (Ticket → Advance → StudentAdvance)

A3: ZigZag Fish (MBCS)

- extend Fish class to yield zig-zag movement (maintaining direction state)

A4: Improving Grades

- traverse an array of grades, determine if non-decreasing and calculate average

AB1: Salmon (MBCS)

- extend Fish class to store home location, move toward home and breed when mature

AB2: Postal Codes (Design)

- design and analyze data structure for city→codes mapping, implement a traversal

AB3: Successor Nodes

- traverse and search a binary tree with parent links

AB4: Expanded Aliases

- perform iterative expansion of email aliases, merging sets and queues

3

Observations from 2004 Exams



- slight drop in exam volume (was it due to language switch?)

- students did OK with OOP emphasis
- students did well on design questions
- students were more familiar with case study
- students were not familiar enough with Collection classes

- performance on A exam was comparable to previous years
- performance on AB exam was considerably lower

4

APCS Exam Count



2005 (preliminary data):

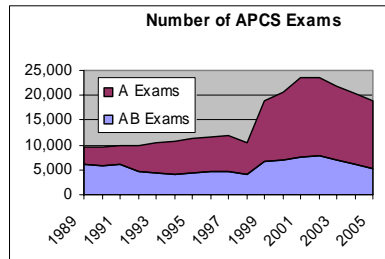
13,924 A
 5,097 AB

 19,021 exams

slight drop from 2004:

14,337 A
 6,077 AB

 20,414 exams



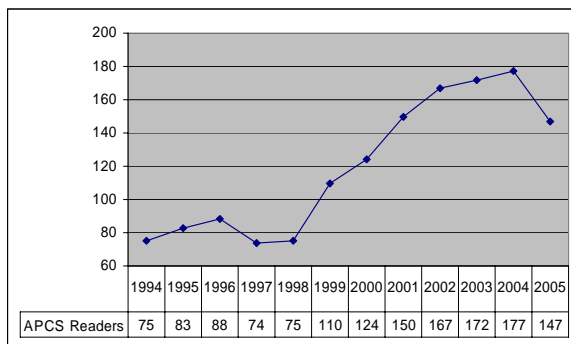
Note: not as dramatic a drop as most colleges are experiencing still, cause for some concern

Grading Process



2005 introduced table leaders to the reading hierarchy

- Chief Reader, 2 Exam Leaders
- 16 Question Leaders (1 college + 1 high school QL per question)
- 17 Table Leaders (1 TL for every 6-8 Readers, organized in pods)
- 111 Readers



in 2005, roughly a 45/55 college-to-high-school ratio

always looking for new readers, especially from colleges

apply online at:

<http://www.ets.org/reader/ap/requirements.html>

Grading issues with Java



all questions are designed with the APCS Java subset in mind

- however, solutions that utilize constructs/classes outside the subset are NOT penalized (unless the question specifically forbids it)
- likewise, code based on Java 5 is NOT penalized

as in previous years, some minor errors are ignored when grading

e.g., missing semicolons, = instead of == , case discrepancies

e.g., no penalty if fail to downcast when accessing a collection

```
String customer = waitlist.get(0); instead of  
String customer = (String)waitlist.get(0);
```

TEACHERS: ADVISE STUDENTS TO STAY WITHIN THE SUBSET!

7

OOP emphasis



with Java, object-oriented techniques are emphasized

- all problems utilized class design and/or implementation
- most problems utilized Java collections, class/method use
- A2, A3, AB1 utilized inheritance

students did reasonably well, with some confusion on OOP concepts

- common error: not recognizing when inherited data/methods could be used
e.g., overriding parent class instance variables & methods
e.g., attempting to access private data from parent class instead of calling super
- common error: attempting to reimplement existing functionality
e.g., given `minNode` method on tree, but try to implement from scratch

TEACHERS: CONTINUE TO EMPHASIZE OOP & ABSTRACTION!

8

Design Questions



each exam included a question involving class/data structure design

- A2 involved designing & implementing classes in a hierarchy
- AB2 involved designing/analyzing/using a data structure

student performance was mixed

- A2 was lowest mean, despite similarity to 2004 design question
- AB2 absolute mean was reasonable, especially given its open-endedness

	mean score*	% of 0/-	mean w/o 0/-		mean score*	% of 0/-	mean w/o 0/-
A1	4.54	18.6%	5.58	AB1	6.39	3.3%	6.61
A2	4.04	20.1%	5.05	AB2	5.05	12.9%	5.80
A3	4.30	24.5%	5.69	AB3	3.13	16.0%	3.73
A4	5.08	17.7%	6.18	AB4	5.54	13.4%	6.40

*2005 exam data is preliminary

TEACHERS: BE AWARE OF "DESIGN" IN ITS VARIOUS FORMS! 9

Java Collections



Java Collections are used extensively

- A1 : array, ArrayList
- A4 : array
- AB2 : Set, Map
- AB3 : binary search tree, recursion
- AB4 : Set, Map, Queue

	mean score*	% of 0/-	mean w/o 0/-
AB1	6.39	3.3%	6.61
AB2	5.05	12.9%	5.80
AB3	3.13	16.0%	3.73
AB4	5.54	13.4%	6.40

students knew Collection classes much better

- common error on A: confused access on arrays and ArrayLists
- common error on AB: not knowing access efficiency (HashSet vs. TreeSet)

but AB students had MAJOR troubles with AB3

- typically, tree traversal and recursion are difficult topics on the exam
- this year was especially difficult – extra links introduced, nontrivial algorithm

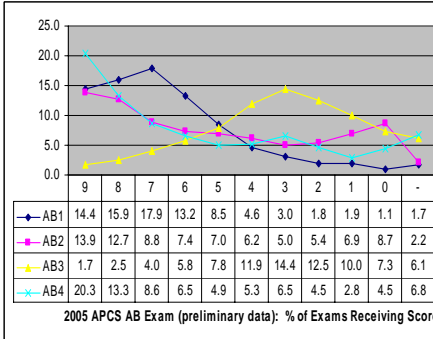
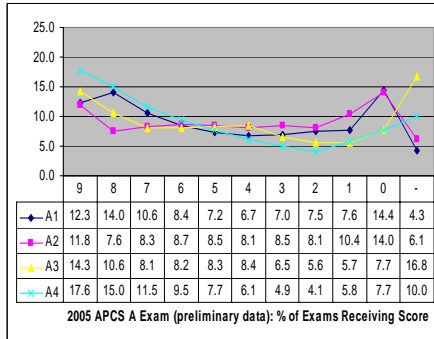
'AB' TEACHERS: DON'T FORGET LINKED STRUCTURES & RECURSION! 10

Java Marine Biology Case Study



student performance on MBCS questions was excellent overall

- A3 had MANY blanks, but 2nd highest mean if throw these out
 - suggests many A students did not know MBCS at all, but those that did were fine
 - knowledgeable students recognized similarity to DarterFish in the Appendix
- AB1 had highest mean



'A' TEACHERS: BE SURE YOUR STUDENTS KNOW THE MBCS! 11

Comparison with Recent Exams



- o A exam performance was comparable to 2004 & previous exams
- o AB exam performance was better, closer to previous exams

Grade	APCS A Exams				APCS AB Exams			
	2002	2003	2004	2005*	2002	2003	2004	2005*
5 (Extremely well qualified)	19.3%	17.0%	18.6%	17.7%	34.2%	37.6%	27.1%	31.0%
4 (Well qualified)	25.3%	24.3%	23.6%	23.1%	12.5%	13.8%	18.2%	19.6%
3 (Qualified)	18.3%	19.8%	15.2%	15.0%	26.5%	24.5%	17.6%	18.2%
2 (Possibly qualified)	8.8%	9.3%	9.5%	10.0%	10.6%	10.1%	12.1%	10.3%
1 (No recommendation)	28.4%	29.8%	33.1%	34.2%	16.2%	14.0%	25.0%	20.9%

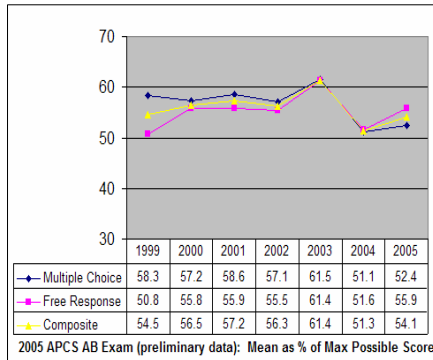
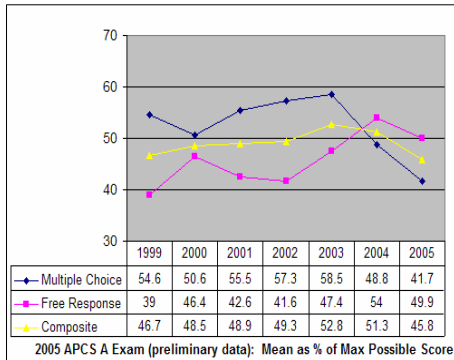
*2005 data is preliminary 12

Comparison: MC vs. FR



goal: means of multiple choice and free response to be 50% of max

- both exams were well within expectations
- as in 2004, free response actually higher percentage on both exams



13

Observations from the AP CS Classroom



- Object Oriented Programming has added more abstraction to the AP Computer Science curriculum
- Object Oriented Programming has added a layer of complexity to the AP Computer Science curriculum
- Most of my students need concrete examples to help them understand the abstract concepts found in the AP Computer Science curriculum

14

Teaching Tips for AP Computer Science





- Model algorithms before you show code
 - Give students a stack of unsorted vocabulary words on index cards. Have them write down the steps they take to sort the words in alphabetical order.
 - Give students 7 circles of different sizes (pancakes). Have them place the circles on their desk (griddle). Give them a paper plate and have them stack the pancakes (one at a time) from largest to smallest with the largest pancake on the bottom. Have them write down the steps they take to stack the pancakes on the plate.

15

Teaching Tips for AP Computer Science



- Use concrete models for abstract concepts
 - Bottle top array: 
 - 8 bottle caps (20 oz soft drinks work best)
 - piece of cardboard
 - Hot glue the caps on the cardboard
 - Cut strips of card stock or index cards for the data
 - Towers of Hanoi toys for stacks
 - Beads on a string for queues (Toddler's pop beads for teacher demo) 
 - Environment grid on card stock with colored goldfish for JMBS
Students run through code with grid and goldfish

16

Teaching Tips for AP Computer Science



- Use concrete models for abstract concepts
 - Barrel of Monkeys – linked lists
 - Playdoh and cookie cutters for class-object relationship
 - Laminated card stock and business card magnets for ListNode and TreeNode practice on magnetic boards



- Laminated index cards in different colors to illustrate object-reference relationships.



17

Teaching Tips for AP Computer Science



- Use concrete models for abstract concepts
 - Create a “map” of students to crayon colors
 - 4 boxes of 8-color crayons
 - paper
 - Hand each student a crayon.
 - Build a map (on paper) of student names to crayon colors
 - Some sample questions to ask:
 - How can you find which crayon Jack has?
 - How can you find all the students who have a specific color?

18

Teaching Tips for AP Computer Science



- Use concrete models for abstract concepts
 - Create a “map” of student (key) to a cup of Mike and Ike candies (value)
 - 2 boxes of Mike and Ike candies (4 or 5 colors)
 - small paper cups
 - paper
 - Model giving each student candy to put in their candy jar (paper cup)
 - **Algorithm:**
 - if the student does not have a paper cup*
 - teacher puts a paper cup on the student's desk*
 - teacher asks for the paper cup and adds a piece of candy to the cup*
 - **Questions to ask:**
 - How do you find out how many pieces of candy Morgan has?
 - How do you find out the colors of each piece of Morgan's candy?

19

Teaching Tips for AP Computer Science



- Concrete to Code
 - **Now code the Algorithm:**
 - if the student does not have a paper cup*
 - teacher puts a paper cup on the student's desk*
 - teacher asks for the paper cup and adds a piece of candy to the cup*
- ```
if(!theMap.containsKey(student))
 theMap.put(student, new ArrayList());
ArrayList candyJar = (ArrayList)theMap.get(student);
candyJar.add(candy);
```

20

## Teaching Tips for AP Computer Science



- Use stories to introduce or reinforce concepts
  - To introduce arrays:  
"Too Many Daves" from The Sneetches and other stories by Dr. Seuss
  - Linked Lists:  
"King Looie Katz" from I Can Lick 30 Tigers Today! by Dr. Seuss
  - Stacks:  
"Yertle the Turtle" from Yertle the Turtle and other stories by Dr. Seuss
  - Recursion:
    - The Cat in the Hat Comes Back by Dr. Seuss
    - "Martin and the Dragon Stories" from LISP: A Gentle Introduction to the Art of Symbolic Computation by David Touretzky

21

## Teaching Tips for AP Computer Science



- Use logic problems to foreshadow upcoming topics
  - Great problems can be found in the following books:
    - The Puzzling Adventures of Doctor Ecco by Dennis Shasha  
ISBN: 0-486-29615-6
    - Problem Solving Strategies: Crossing the River with Dogs and Other Mathematical Adventures by Herr & Johnson  
Key Curriculum Press ISBN 1559530685

22

## Teaching Tips for AP Computer Science



- Software

- Jeliot 3– Algorithm Theater - <http://www.cs.joensuu.fi/jeliot/index.php>  
*Jeliot 3 is a Program Visualization application. It visualizes how a Java program is interpreted by displaying method calls, variables, and operations, allowing the student to follow step by step the execution of a program.*
- Alice – [www.alice.org](http://www.alice.org) Early OOP programming concepts without the overhead of a big language
- KarelJ Robot - <http://csis.pace.edu/~bergin/KarelJava2ed/Karel++JavaEdition.html>  
Karel the Robot in Java – OOP programming without a lot of language overhead
- BlueJ – Java IDE designed for teaching OOP early  
<http://www.bluej.org>

23

## Teaching Tips for AP Computer Science



- Be Creative!
- When you come up with a good idea, share it on the listserv or send to Fran Trees to be posted at AP Central: [ftrees@drew.edu](mailto:ftrees@drew.edu)  
Subject Line: Strategy Suggestion

24

## FYI: Online Resources



- <http://www.dave-reed.com/Talks/APNC2005.pdf>  
these slides, available online
- <http://apcentral.collegeboard.com>  
AP Central: AP info, course descriptions, reference materials, ...
- <http://www.collegeboard.com>  
College Board: general info about the organization, AP program
- <http://cs.colgate.edu/APCS>  
Unofficial APCS site, by Chris Nevison (former Chief Reader)

25

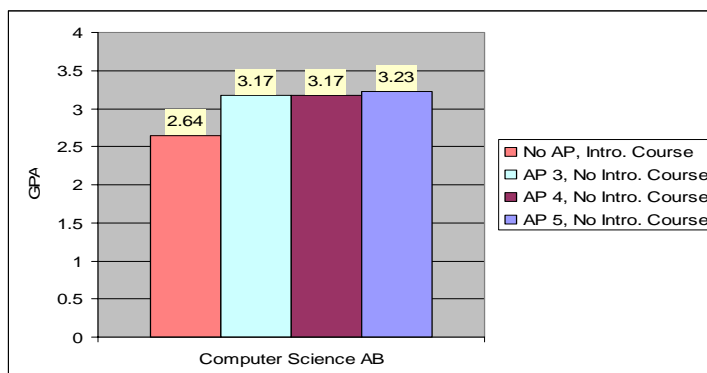
## FYI: AP Research on Performance



AP Research conducted a study from 1996-2001

- 20 different colleges and universities, over 72,000 student
- comparison of next-level college courses for AP and non-AP students

**FINDING:** students with AP credit performed better than non-AP students



26

# FYI: % of Schools with APCS Credit

