

Advanced Placement Computer Science: Meet the Committee

David Reed
Creighton University, Omaha, Nebraska
davereed@creighton.edu

Ann Shen
Bishop Strachan School, Toronto, Ontario
AShen@bss.on.ca

Laurie White
Mercer University, Macon, Georgia
white_la@mercer.edu



APCS Development Committee



- Don Allen, Troy High School, California
- Scot Drysdale, Dartmouth College, New Hampshire (Chair)
- Cay Horstmann, San Jose State University, California
- Tracy Ishman, Plano West Senior High, Texas
- Ann Shen, Bishop Strachan School, Ontario
- Fran Trees, Drew University, New Jersey
- Laurie White, Mercer University, Georgia

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

Retired in 2006:

- Reg Hahne, Atholton High School, Maryland

Observations from Previous Exams



- drop in exam volume the last two years (is it Java?)
- performance on 2004 & 2005 A exams was comparable to pre-Java years
- performance on AB exam was considerably lower in 2004, improved slightly in 2005
- students did OK with OOP emphasis
- student familiarity with Collection classes improved in 2005
- A students had trouble with "design", AB did OK
- many A students still were not familiar with the case study

3

APCS Exam Count

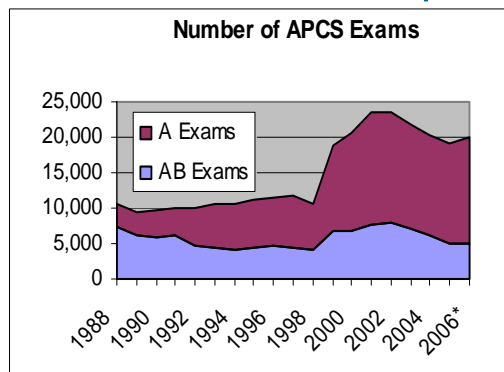


2006* (preliminary data):

14,814 A
5,058 AB
19,872 exams

increase from 2005:

13,924 A
5,097 AB
19,021 exams



Free Response questions were graded during 1 week in Clemson, SC

151 readers (3 Exam Leaders, 16 Question Leaders, 19 Table Leaders, 113 Readers)
53% college vs. 47% high school readers

4

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 5.0 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

```
Appointment nextAppt = apptList.get(i); instead of
Appointment nextAppt = (Appointment)apptList.get(i);
```

TEACHERS: ADVISE STUDENTS TO STAY WITHIN THE SUBSET!

7

2006 Free Response Questions



A1: Appointment Schedule

- maintain an ArrayList of Appointment objects, add a new appt. and remove conflicts

A2: Taxable Items (Design)

- utilize interface & abstract class in designing/implementing a class hierarchy

A3: Customer Lists

- implement comparator for Customer objects, merge lists of Customers

A4: Drop Game (MBS)

- utilize Environment to represent a game board, place pieces and search for pattern

AB1: Thesaurus

- add & remove entries in thesaurus (Map of words and corresponding synonyms sets)

AB2: Packs & Bundles (Design)

- utilize interface & polymorphism in designing/implementing a class hierarchy

AB3: Waiting List

- traverse, disconnect & reconnect parts of linked lists

AB4: Path Finder (MBS)

- recursively search for and store a path between Locations in an Environment

8

Comparison with Recent Exams



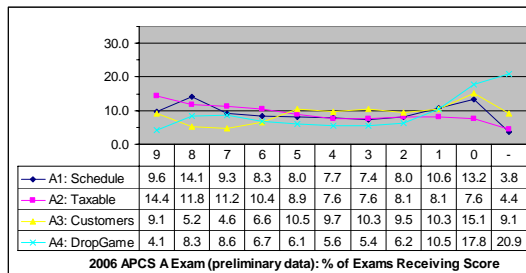
subjective view, based on statistical equators and reader experience:

- o A exam was comparable in difficulty to 2005; AB exam was easier
- o student performance was better at the top, otherwise comparable to 2005

Grade	APCS A Exams			APCS AB Exams		
	2004	2005	2006*	2004	2005	2006*
5 (Extremely well qualified)	18.6%	17.7%	21.8%	27.1%	31.0%	33.9%
4 (Well qualified)	23.6%	23.1%	22.1%	18.2%	19.6%	19.8%
3 (Qualified)	15.2%	15.0%	14.4%	17.6%	18.2%	17.0%
2 (Possibly qualified)	9.5%	10.0%	7.7%	12.1%	10.3%	8.8%
1 (No recommendation)	33.1%	34.2%	34.0%	25.0%	20.9%	20.5%

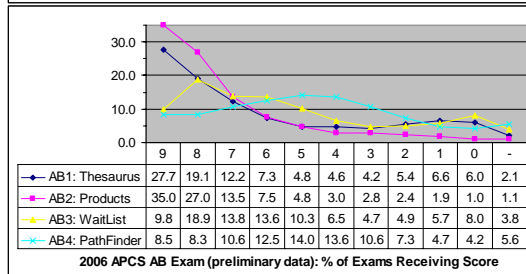
*2006 data is preliminary

Raw Scoring Data



	mean score*	% of 0/-	mean w/o 0/-
A1	4.33	17.0%	5.23
A2	4.85	12.0%	5.52
A3	3.46	24.2%	4.57
A4	2.94	38.7%	4.82

beautiful, even distributions!



	mean score*	% of 0/-	mean w/o 0/-
AB1	6.02	8.1%	6.57
AB2	7.20	2.2%	7.37
AB3	5.24	11.8%	5.94
AB4	4.68	9.8%	5.19

AB2 skewed too high, otherwise good distributions

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, AB2 utilized inheritance & interfaces

students did reasonably well, with some confusion on OOP concepts

- common error: not recognizing when inherited data/methods could be used
e.g., attempting to access private data from parent class instead of calling super
- common error: not knowing that private fields/methods are *class accessible*
e.g., didn't think could access private field of a class object passed as a parameter
- common error: attempting to reimplement existing functionality
e.g., implementing helper method in part A, then reimplementing code in part B

TEACHERS: CONTINUE TO EMPHASIZE OOP & ABSTRACTION!

11

Java Collections



Collection classes are used extensively

- A1 : ArrayList
- A3 : array
- AB1 : Set, Map
- AB3 : linked list (ListNode)
- A4: MBS classes, ArrayList
- AB2: ArrayList
- AB4: MBS classes, List, recursion

students seemed fairly comfortable with Collection classes

- common error on A: confused access on arrays and ArrayLists
- common errors on AB: attempting to create an instance of the Set interface using LinkedList methods on a List object

not surprisingly, the difficulty was in applying complex algorithms to data

- e.g., merging sorted lists in A3, recursive path search in AB4

TEACHERS: EMPHASIZE THE USE OF COLLECTIONS IN SOLVING PROBLEMS, AS WELL AS BASIC METHODS FOR ACCESS/UPDATE

12

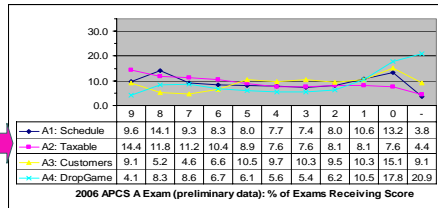
Design Questions



each exam included a question involving the design of a class hierarchy

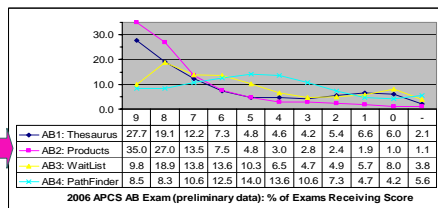
- A2 utilized interface & abstract class; AB2 utilized interface & polymorphism

student performance was excellent – highest mean on each exam



	mean score*	% of 0/-	mean w/o 0/-
A1	4.33	17.0%	5.23
A2	4.85	12.0%	5.52
A3	3.46	24.2%	4.57
A4	2.94	38.7%	4.82

TEACHERS:
CONTINUE TO EMPHASIZE "DESIGN" SKILLS, BOTH CLASS & DATA STRUCTURE DESIGN



	mean score*	% of 0/-	mean w/o 0/-
AB1	6.02	8.1%	6.57
AB2	7.20	2.2%	7.37
AB3	5.24	11.8%	5.94
AB4	4.68	9.8%	5.19

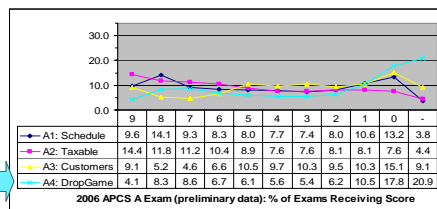
13

MBS Case Study



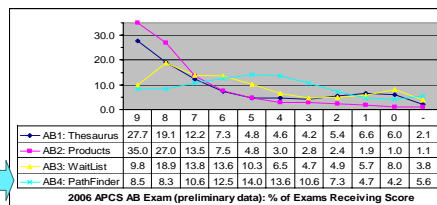
student performance on MBS questions was good overall

- A4 had MANY blanks, but reasonable mean if throw these out
- AB4 had lowest mean, but still high (especially for a recursion question)



	mean score*	% of 0/-	mean w/o 0/-
A1	4.33	17.0%	5.23
A2	4.85	12.0%	5.52
A3	3.46	24.2%	4.57
A4	2.94	38.7%	4.82

A EXAM TEACHERS: BE SURE YOUR STUDENTS KNOW THE MBS! (JUST ONE MORE YEAR)



	mean score*	% of 0/-	mean w/o 0/-
AB1	6.02	8.1%	6.57
AB2	7.20	2.2%	7.37
AB3	5.24	11.8%	5.94
AB4	4.68	9.8%	5.19

14

2006 Exam in Summary



- exam count was up
- student performance was very good, especially at the top
- OOP concepts: good
- Collections: good, but some trouble applying complex algorithms to data
- "Design" skills: excellent
- MBS case study: still unfamiliar to many A students

15

Presenter Information



Ann Shen

- Bishop Strachan School
- High school teacher
- Private girls school in Toronto, Ontario, Canada
- 10 years teaching (9 years AP)
- A level in Grade 11 and AB level in Grade 12
- AP Computer Science Reader since 2001

16

Presenter Information



Laurie White

- University Professor
- Mercer University, Macon, Georgia
- CAC accredited program
- 24 years teaching, with an emphasis on programming and programming languages
- AP Computer Science Reader since 1991
- Committee member for 1 year

17

FYI: Online Resources



- o <http://apcentral.collegeboard.com>
AP Central: AP info, course descriptions, reference materials, ...
- o <http://www.collegeboard.com>
College Board: general info about the organization, AP program
- o <http://www.dave-reed.com/APCS>
unofficial APCS site, includes this and other AP-related talks
- o <http://cs.colgate.edu/APCS>
unofficial APCS site by Chris Nevison (former Chief Reader)

18