MythSmashers
Episode 1: Science labs

- All science labs are created equal...er, are run in the same way.
Which of these things is not like the others?

- Traditional CS
- Paired CS
- Chemistry
- Biology
All science labs are created equal... er, are run in the same way.
Which major will this student choose?
• All science labs are created equal . . . er, are run in the same way.
Episode 2: Half the Students Learn

• Half the students won’t learn anything. These students will fall on their faces when they have to work alone.

“In the first course, students need some time to absorb the ideas themselves.”

“My inclination is to allow more group work starting in the second course.”

“We want to be sure that each student writes enough code him/herself to learn the introductory concepts.”

“I am against pair-programming in introductory courses, where students need to develop strong programming skills themselves.”
Pair Programming in Intro Course

• North Carolina State University
  • Fall 2001, Spring 2002, and Fall 2002 → AB-AB-AB
  • 660 engineering students
  • Southeastern US, very large public university
  • Large lecture sections
  • Closed lab
  • Pairs assigned, pair rotation

• University of California Santa Cruz
  • Fall 2000, Winter 2001, and Spring 2001 → A-B-A
  • 555 engineering students
  • Western US, large public university
  • Large lecture sections
  • Open lab
  • Pairs by student choice, same partner all semester
Exam Scores

Students who work in pairs will earn exam scores equal to or higher than solo programming students.

<table>
<thead>
<tr>
<th></th>
<th>Pair</th>
<th></th>
<th>Solo</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std. Dev.</td>
<td>N</td>
</tr>
<tr>
<td>NCSU F01</td>
<td>74.1</td>
<td>16.5</td>
<td>44</td>
</tr>
<tr>
<td>NCSU S02a</td>
<td>70.6</td>
<td>28.8</td>
<td>82</td>
</tr>
<tr>
<td>NCSU S02b</td>
<td>71.9</td>
<td>26.7</td>
<td>198</td>
</tr>
<tr>
<td>NCSU-F02</td>
<td>75.1</td>
<td>15.7</td>
<td>55</td>
</tr>
<tr>
<td>UCSC</td>
<td>75.2</td>
<td>18.9</td>
<td>367</td>
</tr>
</tbody>
</table>
Future Success

The use of pair programming in an introductory computer science course does not hamper student performance in future solo programming courses.

<table>
<thead>
<tr>
<th>Semester</th>
<th>Paired (%)</th>
<th>Solo (%)</th>
<th>Statistical Sign</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1: Spring 2002 – CS2: Fall 2002</td>
<td>26.37 (24/91)</td>
<td>29.50 (18/61)</td>
<td>No. $\chi^2$=0.179, $p&lt;0.672$</td>
</tr>
</tbody>
</table>

NCSU: % of students whose grades dropped by more than 1/3 of a grade

<table>
<thead>
<tr>
<th>Attempt Rates</th>
<th>Pass Rates (on 1st attempt) of Attempters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pair</td>
<td>76.7%</td>
</tr>
<tr>
<td>Solo</td>
<td>62.2%</td>
</tr>
</tbody>
</table>

UCSC: Attempt and Pass Rates for Second CS Class
• Half the students won’t learn anything. These students will fall on their faces when they have to work alone.
Episode 3: Student Retention

• We can attract and retain more students in computer science if we let them collaborate [through pair programming].
Success Rate

An equal or higher percentage of students in paired labs will complete the class with a grade of C or better compared to solo programmers.

<table>
<thead>
<tr>
<th></th>
<th># Paired</th>
<th>% Pair passing</th>
<th># Solo</th>
<th>% Solo Passing</th>
<th>Stat. Sign.</th>
</tr>
</thead>
<tbody>
<tr>
<td>NCSU</td>
<td>171</td>
<td>70.76</td>
<td>255</td>
<td>60.00</td>
<td>Yes. $\chi^2=5.61$, $p&lt;0.023$</td>
</tr>
<tr>
<td>UCSC</td>
<td>404</td>
<td>72.30</td>
<td>148</td>
<td>62.80</td>
<td>Yes. $\chi^2=4.57$, $p&lt;0.05$</td>
</tr>
</tbody>
</table>
Persistence in Computer Science

Students participating in pair programming will be significantly more likely than solo programmers to pursue computer science-related majors one year later.

<table>
<thead>
<tr>
<th></th>
<th>Paired</th>
<th>Solo</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>UCSC</td>
<td>56.9%</td>
<td>33.8%</td>
<td>$\chi^2(1) = 12.18, p &lt; .001$</td>
</tr>
<tr>
<td>NCSU</td>
<td>25.6%</td>
<td>10.5%</td>
<td>$\chi^2(1) = 7.434, p &lt; .006$</td>
</tr>
</tbody>
</table>

Percentage of students declaring a Computer Science major 1 year after CS1
• We can attract and retain more students in computer science if we let them collaborate [through pair programming].
Episode 4: Teacher Intervention

- Teachers who allow their students to work in pairs spend a bunch of time dealing with students not getting along with their partner.
Collaboration Management: Compatibility Study Design

• CS1 (Freshman, 804 students)
  – Closed lab
  – Four projects
  – Assigned a new partner every 2-3 weeks.

• Software Engineering (3rd/4th year, 434 students)
  – Closed lab plus additional collaboration.
  – Four projects
  – Assigned a new partner each project

• Object-Oriented (Graduate, 112 pairing students)
  – No closed lab
  – Pairing optional
  – TA assigned partner
Select the evaluation number: 

Select your partner be evaluated: Marie Boucher

| Has the student attended your group meetings? | never |
| Has the student notified a teammate if he/she would not be able to attend a meeting or fulfill a responsibility? | never |
| Has the student made a serious effort at assigned work before the group meetings? | never |
| Does the student attempt to make contributions in group meetings when he/she can? | never |
| Does the student cooperate with the group effort? | never |
| Assess the technical competency of your partner relative to yourself. | Better than me |
| **Assess how compatible you and your partner were** | Very Compatible |

### Overall rating

- **Excellent**: Consistently went above and beyond -- tutored teammates, carried more than his/her fair share of the load.
- **Very Good**: Consistently did what he/she was supposed to do, very well prepared and cooperative.
- **Satisfactory**: Usually did what he/she was supposed to do, acceptable prepared and cooperative.
- **Ordinary**: Often did what he/she was supposed to do, minimally prepared and cooperative.
- **Marginal**: Sometimes failed to show up or complete assignments, rarely prepared.
- **Deficient**: Often failed to show up or complete assignments, rarely prepared.
- **Unsatisfactory**: Consistently failed to show up or complete assignments, unprepared.
- **Superficial**: Practically no participation.
- **No show**: No participation at all.

**Comments:** no more than 255 characters.

- Peer eval instrument developed by Rich Felder at NCSU.
## Overall Compatibility Results

<table>
<thead>
<tr>
<th>Class</th>
<th>Very compatible</th>
<th>OK</th>
<th>Not compatible</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1</td>
<td>64%</td>
<td>32%</td>
<td>4%</td>
</tr>
<tr>
<td>SE-P1</td>
<td>60%</td>
<td>33%</td>
<td>7%</td>
</tr>
<tr>
<td>SE-P2</td>
<td>56%</td>
<td>35%</td>
<td>9%</td>
</tr>
<tr>
<td>OO</td>
<td>76%</td>
<td>15%</td>
<td>9%</td>
</tr>
<tr>
<td>Total</td>
<td>60%</td>
<td>33%</td>
<td>7%</td>
</tr>
</tbody>
</table>
• Teachers who allow their students to work in pairs spend a bunch of time dealing with students not getting along with their partner.
Episode 5: Saving Time

- Teachers who allow their students to work in pairs save time.
Grading: You do the math

- 20 assignments, 15 minutes each = 5 hours
- 10 assignments, 20 minutes each = 3.33 hours

5 > 3.33
## Questions in Lab

<table>
<thead>
<tr>
<th></th>
<th>Number of Questions</th>
<th>Questions Per Person</th>
<th>Answer Latency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solo Lab Observation 1</td>
<td>111</td>
<td>1-13</td>
<td>0-10 minutes</td>
</tr>
<tr>
<td>Solo Lab Observation 2</td>
<td>81</td>
<td>1-9</td>
<td>0-10 minutes</td>
</tr>
<tr>
<td>Paired Lab Observation 1</td>
<td>43</td>
<td>0-7</td>
<td>0-3 minutes</td>
</tr>
<tr>
<td>Paired Lab Observation 2</td>
<td>55</td>
<td>0-7</td>
<td>0-2.5 minutes</td>
</tr>
</tbody>
</table>
• Teachers who allow their students to work in pairs save time.
Episode 6: Student Preference

- Students *always* prefer to work in pairs.
• Students **always** prefer to work in pairs.

• Stronger students

• Logistical issues

• Scheduling issues
Results Summary

• Pair programming students tend to:
  • Make it through the first class
  • Perform comparably or better on exams and projects
  • Perform just fine in future solo programming
  • Stick with computer science

• Also . . .
  • Teaching staff prefer pair programming classes
  • Helps teachers respond to spectrum of learning preferences, personality types, and to the current generation.

• These benefits outweigh the costs . . .
  • Some dysfunctional pair management
  • Need for pair evaluation