ENGAGING STUDENTS IN THINKING LIKE COMPUTER SCIENTISTS: PEER INSTRUCTION

Beth Simon, CSE Univ of California, San Diego
The Big Concern: Creating Computer Scientists

Doing Vs. Understanding or even Being
What do we ask computing students to do?

Revised Bloom’s Taxonomy Levels

- Remembering
- Understanding
- Applying
- Analyzing
- Evaluating
- Creating
What do we ask computing students to do?

Revised Bloom’s Taxonomy Levels

In Introductory Programming
What do we ask computing students to do?

Revised Bloom’s Taxonomy Levels

In Introductory Programming

Creating
Evaluating
Analyzing
Applying
Understanding
Remembering
What do we ASSESS computing students at doing?

Revised Bloom's Taxonomy Levels

In Introductory Programming

```cpp
int findgcd (int x, int y)
{
    while (x>0 && y>0)
    {
        if (x>y)
            x = x % y;
        else
            y = y % x;
    }
    if (x == 0)
        return y;
    else return x;
}
```

Revised Bloom’s Taxonomy Levels
What do we ASSESS computing students at doing?

In Other Courses?

What do you KNOW a student understands from an Exam?
Submitted program?

Remembering

Revised Bloom’s Taxonomy Levels
We Have Them Doing

But is it enough to be sure they are Understanding Like Computer Scientists Being Like Computer Scientists
You say...

- Beth, be reasonable!
- This is an academic setting!
- We have limitations!
You say...

- Beth, be reasonable!
- This is an academic setting!
- We have limitations!
Which happens A LOT (most?) in lecture?

A. Remembering
B. Understanding
C. Applying
D. Analyzing
E. Evaluating
Who is involved the MOST in those activities?

A. Those same 5-10 students
B. ½ the students
C. All of the students
D. Instructor
My view of lecture:

Lecture → Textbook → Homework → Exam

First Exposure → Read Hard Stuff → See if You Know Hard Stuff → Show Knowledge Mastery

No opportunity for expert feedback!
My view of lecture:

Lecture → Textbook → Homework → Exam

First Exposure → Read Hard Stuff → See if You Know Hard Stuff → Show Knowledge Mastery

Textbook (or...) → Lecture → Homework/Lab → Exam

Read Easy(er) Stuff → Learn Hard Stuff: With teacher and discussion → Practice Knowledge Mastery → Show Knowledge Mastery

Expert guides in applying, analyzing, evaluating
How? Peer Instruction!
How? Peer Instruction!
How? Peer Instruction!
How? Peer Instruction!

Design crafted questions to engage students in applying, analyzing, and evaluating
The Peer Instruction Algorithm

Pre-Class preparation
Quiz/Incentive/Feedback

Textbook (or...)

Read Easy(er) Stuff
The Peer Instruction Algorithm

Pre-Class preparation
Quiz/Incentive/Feedback

1) Individual Thinking, Vote
2) Group Discussion (with 2 other students)
3) Group Vote

Class-wide discussion
Student-led/Instructor Modeling/Mini-lecture

Learn Hard Stuff:
With teacher and discussion

Textbook (or...)

Read Easy(er) Stuff
What does this code do?

A. Makes the eskimo girl say Hello, then jump up and down
B. Makes the eskimo girl say Hello WHILE jumping up and down
C. Makes the eskimo girl say Hello
D. None of the above
Good Peer Instruction Questions

- Are designed based on the DISCUSSION they will generate
  - Not (as much or at all) to “test” knowledge/comprehension

- Should be examples of:
  - What YOU would look at/pay attention to
  - What YOU would analyze (trade-offs, etc.)
  - What YOU know students often struggle with
  - What YOU know is critical to really, really understand deeply
Yes, you need clickers
Yes, you need to give marks (and candy)
The Big Concern: Creating Computer Scientists

Doing Vs. Understanding or even Being
If you want to spend your expert time with students...

- Guiding them in applying
  - Challenging situations
- Focusing them on analyzing
  - What to pay attention to
  - What questions to ask
  - How to test
- Guiding them in evaluating
  - Appropriate tradeoffs
  - Common mistakes
How do I learn more about Peer Instruction?

- Carl Wieman Science Education Initiative
  - http://cwsei.ubc.ca
  - Resources-> Clickers
  - Has MANY resources
  - Start with Video Page
- http://peerinstruction4cs.org
Thank you!
Best Practice Quick Sheet

- Voting in class must be part of the grade (3-10%)
- Give reading quizzes before each lecture (or with clickers at beginning of each lecture)
- Foster group discussion
  - Assign groups (and keep the same ones all term)
  - Require them to reach consensus and vote “the same”
  - (Rotate ownership of discussion within groups)
- Ask challenging questions, about things you really care about
  - That make for meaningful discussion
  - 35-75% solo -> >75% group
- Respond to student experience
  - Ask them to provide explanations first (then you can re-word)
  - If the group vote correctness is low
    - Take time to review, explain, clarify
- Frequently (once a week, then often) remind students of the value of Peer Instruction