Software Engineering Instruction and Education Theory: A Dialogue

Patricia A. Basili
Prince George’s Community College
And
Victor R. Basili
University of Maryland

Copyright 2006 Victor and Patricia Basili
My Personal Experience

- In 1975 I taught my first undergraduate software engineering course covering methods and techniques that were state of the art at the time and assigned team projects.

- What were my goals for the course?
  - Teach students to think as an engineer of software
  - Facilitate skill development in
    - problem solving
    - applying techniques
    - integrating techniques
    - communicating with others
    - asking the right questions
  - Provide a context for engineering-decision making
    - measurement and feedback
My Personal Experience

• As a way of addressing these goals, I assigned a project to be developed by teams of 3 people to build skills in *communicating within a problem space*

• They were assigned roles: manager, chief programmer and librarian, but all had the task of coding and collecting data (shrinking Mills’ Chief Programmer Team)

• They had to make two to three presentations to build skills in *organizing and communicating with outsiders*

• We used this study as an opportunity to run an experiment on the effects of methodology
My Personal Experience

• Why a project as part of the course?
  – Build skills with something larger than what a single programmer could do
  – Learn to communicate and integrate as a team
  – Gain some insights into what it would be like on a real software development
  – Allow us to study the effects of methods on students

• Why student presentations?
  – So they could see there were other approaches to solving the same problem
  – So they could learn to listen carefully to other options – noting they were bringing a similar experience to the table
  – So they could gain some skill in presenting their material
My Personal Experience

• Why collect data?
  – So *they* could have feedback on what was going on,
    • Observe themselves
    • Understand where they are spending their time
    • What kinds of problems they are having
    • Use data as a form of control
  – So that *we* could understand and see if we could evaluate the effects of techniques
    • How to teach better
    • What to teach
    • As a mechanism to study the feasibility, usefulness and effectiveness of various methods and techniques
My Personal Experience

• Over the years, the assignments changed
  – From an emphasis on coding to design to requirements to building systems from components
  – Maintenance rather than development assignments

• Over the years the techniques taught and applied in projects changed
  – Chief programmer teams, functional design, Cleanroom, Mills’ functional specifications and state machines, object oriented design, COTS based development
  – Note: This makes it hard to study and evolve effectiveness when the course changes almost every couple of years

  – What is the core of information that has stayed constant since 1975
    • Problem solving, interaction, communication, technique skill developments
My Personal Experience

• In 1975, I was criticized for giving group projects
  – Some form of cheating, hard to provide individual grades, …

• Others have
  – Used the whole class on one project
  – Created competing teams
  – ….

• Now most of our courses at UMD, not just software engineering,
  have projects associated with them, data bases, systems, etc.

• But we do not link them to software engineering, i.e., they do not
  necessarily apply good software engineering techniques on those
  projects
Three Issues

– Course content

– Making the most of the project

– Validating what has been learned

Copyright 2006 Victor and Patricia Basili
Issue: Course Content

– There is too much material to cover and it is not clear what methods and techniques should be covered
  • How do we cover the material necessary and provide the appropriate skill development?
  • Shouldn’t there be a whole course on testing, on design, on requirements elicitation?
  • Does a company want to hire someone who knows only one testing technique?
  • Other programs represented here have developed two course sequences or even whole programs

– This talk does not cover this issue

Copyright 2006 Victor and Patricia Basili
Issue:
Making the most of the project

– How do we get the maximum benefit out of the project assignments?
  • Can we take advantage of some of the concepts from educational theory to maximize the efforts spent on class projects?

– How do we conceptualize our goals to improve the software engineering class?

– How do we improve the class as a learning environment?

Copyright 2006 Victor and Patricia Basili
Issue: Validating what has been learned

– How do we identify the various learning expectations?

– How do we know we have achieved those expectations?

– How do we reconfigure course goals to meet the accountability requirements of accrediting agencies, e.g., courses outcomes for the middle states accreditation?

– How do we define outcomes for the software engineering class?
Transition
Education is not a discipline per se- it is a field of study that utilizes research methods of
- Psychology
- Sociology
- Anthropology
- Biology
Paradigm Shift in late 70’s
From:
Behaviorist View of the Learning Process
To:
Constructivist View
Behaviorism

• Stimulus → Response
• Teaching Strategy → Student Behavior
• Teaching → Learning
• Cause → Effect
Students Absorb knowledge
Constructivism

Originates from Cognitive Psychology

Examines: Stimulus $\rightarrow$ Response

↑

What happens during learning?
Constructivism

• To learn anything each learner must construct his/her own understanding by tying new information to prior experiences.
Implications of Constructivism

• Knowledge is idiosyncratic (cannot assume that the what the instructor has said – is what has been learned)

• Starting point of instruction is eliciting what they know – accessing the mental schema where related information is stored. (questioning, asking for a prediction, pre-testing, reviewing, making explicit what has been done, what we are going to do today, and where this fits in the course.)

Copyright 2006 Victor and Patricia Basili
Implications of Constructivism

• Students are “active” builders of knowledge not passive recipients.

• What is going on in their heads- as you lecture- is an important factor in whether the material is learned correctly, and whether it is remembered. Meaning must be made.
Implications of Constructivism - Formative Assessments

- Questions
- Think-Pair-Share
- Minute Paper
- Graphic Organizers
Group Work

• Vygotsky – a developmental theorist- stated that the intellectual development of children required social interaction.

• With adults – group allows one to test out their thinking – and to learn problem solving methods- be more efficient problem solvers than when working alone. Active learning technique.

• Models “real” world of work.
Group Work

Roger T. and David W. Johnson
University of Minnesota
“Active Learning Cooperation in the College Classroom

Robert E. Slavin, Johns Hopkins
“Cooperative Learning: Theory, Research, Practice”

S. Sharan –”Handbook of Cooperative Learning Methods”

Copyright 2006 Victor and Patricia Basili
Issues of Alignment/Accountability

• Content

• Class Activity

• Assessment
Bloom’s Taxonomy

• Hierarchy of cognitive tasks.
• Different level of tasks require different methods of instruction and different types of assessment.
• Lowest Level “knowledge”
  Factual information
To be remembered – must be connected to something – and meaning must be made.
Knowledge Level

• If “isolated fact” – remembering only accomplished through rehearsal.
• Mnemonic Devices
• Teaching is telling.

Testing – is “recall” What is, Define, List
Taxonomy Levels

- Knowledge
- Comprehension
- Application
- Analysis
- Synthesis
- Evaluation
Hands-on Activity

Arrange the cards:
Column 1 – Bloom’s Cognitive Levels
Column 2– Definition of level
Column 3 – What the instructor does
Column 4 – What the student does
Column 5– What kind of test item may be appropriate.
What are my goals for the course?

• Expose students to the current techniques

• Facilitate skill development in
  • problem solving
  • applying techniques
  • integrating techniques
  • communicating with others
  • asking the right questions

• Provide a context for engineering-decision making
  – measurement and feedback
  – What works best when
My goals for the course revisited

• Successful students should be able to
  – Know at least a couple of methods and techniques that are available for various phases of the life cycle
  – Comprehend what the methods/techniques are supposed to do, i.e. what does a test technique do?
  – Apply a subset of techniques effectively
  – Analyze the problem so they can identify the main issues that need to be addressed by the techniques
  – Synthesize a solution using an integrated set of techniques to solve a problem
  – Evaluate how well their solution worked

Copyright 2006 Victor and Patricia Basili
Assessable Outcomes

• Think in terms of what you want students to know and be able to do

• Try not to use words like “understand” but rather – what do they have to do to demonstrate understanding.
Dialogue

• Let’s align an outcome – an activity – and an assessment item.