Preparing Students for SE research and practice

What are we doing right? Wrong? Could we do better??

John Grundy

Professor of Software Engineering

HOD, Electrical and Computer Engineering

First a Quizz – who are the Software Engineers in the Audience...? ☺

(Thanks to http://www.funtrivia.com/playquiz/quiz2416201baa068.html)

Q1: Design is one step in the process of software development, in which the requirements for building the software are gathered and analyzed in order to create an architectural model.

✅ True

- False

Q2: Which of these are characteristics of a strong design?

- Low Coupling
- Modular
- High Cohesion
- ✅ All of these

Q3: The two levels of design for software architecture are:

- coding design and architectural design
- ✅ data design and architectural design
- pattern design and data design
- architectural design and XP design

Q4: In software engineering what does KDD stand for?

- Knowledge distortion of design
- ✅ Knowledge discovery in databases
- Knowledge development in design
- Knowledge detection of databases

Q5: What is the use of an Architectural Description Language (ADL)?

✅ provides syntax for describing a software architecture

- helps design the language in which the software should be built

- codes the architecture into desired programming language

- describes how to design the software in a specified language

Q6: If a software engineer wants to check the complexity of an architecture, he/she may use:

- ✅ dependencies
- architecture testing
- coupling
- beta testing

Q7: Which of these is an example of an archetype?

- Actors
- ✓ Node
- Architecture style
- All of these

Outline

- Motivation
  - ME(SE) feedback from graduates
  - CompSci Part 3 offerings re-gig
  - Long-time John G biases/experiences :-)
- Requirements
- Design
- Implementation
- Evaluation
- Reflection

ME(SE) Feedback

- As part of Graduating Year Review for ME(SoftwareEng) – JG interviewed 8 ME grads (graduating 2006-2008)
- All working or worked in industry
- Some feedback (all excellent BTW!) of key benefits:
  - Time management
  - Ability to work alone/in team
  - Team work skills
  - Presentation, technical writing skills
  - Critical thinking/comparison of technologies etc skills

Part III CompSci reg-gig

- CS/SoftEng co-teaching
  - CS345/SE350 (HCI)
  - CS351/SE351 (Databases)
  - CompSys304/SE363/CS313 (Computer Arch)
  - possibility of CompSci students doing SE325 (Software arch)
- Web + remote objects combined
- Graphics + image proc combined

Some reflections...

- Industry feedback (to JG + others) on CS & SE grads
  - Do we formally capture/review/act on/advertise...??
- Jobs available – nature of the work
- Shortage of graduates
- Difficulty attracting/retaining students to CS, SE, IT in general...
- Where do most graduates go?
  - Industry or research?
- Research students of JG
  - Strengths and gaps

Requirements 1: What do we need??

- SERG research students?
- Software industry?

SERG	Industry
#1:	#1:
#2:	#2:
#3:	#3:
#4:	#4:
...	...

Requirements 2: What do we get??

- HOW DO WE KNOW???
- Is there a tension between these "stakeholders"?
- What do we get:
  - Strengths:

  - Weaknesses:

Design: How to get what we want/need...

- BE(Software):
  - 4 year programme (~15-16 SE courses)
  - Mix of theory, practice, experience (e.g. 306, 401A&B)
  - Mix of staff expertise (CompSci, ECE, other Eng)
  - High GPA entry students (with physics, calc backgrounds)

- Issues:
  - Sufficient "research skills" background for ME/PhD?
  - Need to ensure Post-grad courses for direct to PhD?
  - Sufficient "theory"? Too much "practice"??

Design 2

- BSc (3 year):
  - More variable student background
  - More flexibility in programme
  - Mix 300/700; other Department/Faculty courses
  - Projects (380/780/789) incl substantive research and/or practice/writing

- Issues:
  - Sufficient "SoftEng" material for SERG projects?
  - Sufficient "SoftEng" for industry??
  - High attrition rate Part 1->2->3

Implementation

- Assignments
  - Learning by doing...
  - examples used
  - engagement of students
  - performance
  - representativeness of future research/work
  - workload
  - the scourge of plagiarism, cheating, ...

- Real-world vs research vs "toy" problems
- Assessment vs learning by doing
- Team work
- Communication skills
- Critical thinking

Pop Quiz time...

Q1: What was most important thing ME(SE) students thought they got from ME?

    time management

Q2: What are the course codes of the three Computer Architecture courses being taught together in 2009?

    CompSys 304, CompSci 311, SoftEng 363

Q3: What was the most important thing we identified for SERG post-graduates?

    ☺

Q4: In which course do students gain detailed experience with writing skills in BSc programme?

    380 project

Q5: In which 2 BE courses do students do a lot of "experience" work vs theory/practice?

    SE306 Team project, SE401A&B Part 4 Project

Evaluation

- How well do we actually evaluate our graduates for research (post-grad) & industry outcomes?
  - Graduate feedback?
  - Employer feedback?
  - Other research institution feedback?
  - Our own feedback?? :-)
    - When was the last time we talked about this stuff?
- What do others do?
- How do we respond to issues?
  - We know best… ☺
  - Industry/other research & teaching orgs

Reflections

- Rather topical issues:
  - http://www.cc.gatech.edu/conferences/fse16/sees.php
- Do we compromise SE and/or CS programmes by co-teaching courses/material?
- Educate for industry vs educate for research – tension or symbiotic?
- Software for its own sake vs software in other domains (e-Science, Engineering, businessm, entertainment, ...)
- Generation-Y "characteristics" important?? (self-focus vs employer focus; tech-savvy; ambitious; collaborative vs competitive...)

References/Items of Interest

- http://www.cccblog.org/2008/07/11/computer-science-enrollments-the-real-news/

- http://www.cc.gatech.edu/conferences/fse16/sees.php

- http://www.funtrivia.com/playquiz/quiz2416201baa068.html