

CSC 300

Professional Responsibilities

- Instructor: Clark Savage Turner
- Office: 14-222, Phone: 756 6133
- Office Hours (tentative):
 - Monday 12:10 - 1 pm
 - Tuesday 2:10 - 5 pm
 - Wednesday 5:10 - 6 pm
 - and by appointment
- Email: csturner@calpoly.edu
 - don't count on email (or cellphones!)
 - watch for spam filtering (use calpoly accounts)

Texts

- **Recommended:**
 - Johnson, Computer Ethics, 3d Ed., Prentice-Hall
 - Petroski, To Engineer is Human
 - Yourdon, Death March
 - Baase, A Gift of Fire
 - Martin, Schinzinger, Ethics in Engineering
- *Very important to writing (and grade in 300)*
 - Turabian, A Manual for Writers
 - Strunk and White, The Elements of Style

Participate by volunteering: short reports on current issues

- Try this:
 - Go to a LUG meeting
 - Read 2600 magazine
 - Read (usenet) comp.risks
 - Peruse Slashdot
 - Read the business section of the newspaper
 - Listen to NPR
 - Bring your own work experience
 - Make friends with local hackers
 - Watch videos on You Tube
 - Play WOW

Assignment and Reading

- Review Standish Group's "Chaos Report" tonight
 - referenced on my webpage
- Read, in detail, "Unskilled and Unaware"
- Look over the SE Code of Ethics
 - Linked from my webpage
 - Oral QUIZ in class on the code during the second week of classes.
 - You need to read and understand the major topics and some details
 - Be prepared to discuss a few in detail during class

Assignment and Reading (cont'd)

- Plan to read assigned papers in detail
 - take notes and look for the important points
 - why did the author write this paper?
 - what are the main points of the paper
 - what are the main arguments supporting the authors thesis?
 - why is the paper considered important even today?
 - what is the author's pedigree, position today?
 - how much is the paper cited in other works?
 - what do you find strong / weak about the paper?

Lab 1 Assignment

- Prepare 1 page “future alumnus” report
 - give me a vision of what you hope to achieve in the 10 years beyond graduation.
 - where will you live?
 - what will you be doing?
 - what will you have achieved?
 - Include a photo at the top
 - due at the end of lab on Friday, week 1

Prerequisites

- Prerequisite for this class
 - CSC 205 (308) or CSC 309
 - no exceptions
- Make sure you are on the roll,
 - and you know the drop dates

Look at Course Website

- Tour the website
- Note that class pronouncements have priority over web pages
 - I may announce things in class that are not on the web and may not ever make it to the web.

General Course Themes

- Review course description from catalog
- Define terms as we encounter them
 - there is a lot of ambiguity out there
- Spot relationships between technical and social realms
 - and *communicate clearly* about it

Grading

- See website for assignment / exam details
- Goals: (How to get an A, B, C, D or F)
 - *consistently*
 - develop communication skills
 - writing effectiveness is **assumed** (spelling, grammar, clarity and style)
 - develop research skills (not wikipedia)
 - develop critical thinking (not opinion)
 - see higher implications of low level technologies
 - look at computing in a situated context
 - computer scientists have no inherent right to do CS and SE, they perform service for society (who supports them)

Grading (cont'd)

- see tradeoffs and able to make legitimate arguments for alternative designs and outcomes
- become familiar with Codes of Ethics
 - why do we have Codes?
 - how do we use a Code of Ethics?
 - are we “Professionals” - what does that mean?
- become familiar with current topics in computing ethics
 - and their social implications
- Not necessary (possible) to reach “correctness”
 - must be satisfied with rough methods for ethical analysis

Grading (cont'd)

- compare this to “correctness” for software?
 - are we any “better”?
- Perspective on grades
 - evaluation is part of life
 - but not all of it :-)

Underlying Questions and Definitions

- What is “ethics”
- What are “codes”
- Who *should* care
 - why should anyone care anyway?
- What is an “employee”
- What is a “professional”
- What is a “system” - “emergent behavior?”
- Digital vs. Continuous
- Duty to meet a “contract” or “solve a problem?”

Software / Computing

- What are YOU doing here?
 - Why do we get to do computing?
 - Who pays for this?
 - Who suffers costs / enjoys benefits?
 - Who has “authority” to direct, restrict, guide?
 - What are the issues of consequence?

Ultimate Goals for CSC 300

- You'll know the SE Code of Ethics
 - and how to use it
- Broad general knowledge of issues and tradeoffs in computing and ethics
 - familiarity and ability to argue reasonably for alternative designs
- A high quality 20-30 page paper in some area of computing ethics
- A set of CSC 300 lab reports to show ethics experience
 - developed by you in groups

Intro Cases to think about

- Final exam on professor's display
 - you are invited but unobserved
- Internet gambling program flaw
 - illegal to gamble in your state
- Avionics control systems contract
 - impossible to meet software requirements
- Wardriving and mapping to put on web

Thoughts on Analysis of Issues

- Who are the stakeholders?
 - direct and indirect
- What obligations are at stake?
 - legal, ethical, fiduciary...
 - what level of obligation is at stake?
 - professional or employee
- What are the tradeoffs made for a given solution?
 - the benefit (always?) has a cost

Thoughts regarding Case Studies

- How do we proceed?
 - Look at the undisputed relevant **FACTS**
 - no argument from any side: the background
 - Find the **ISSUES**
 - what are the questions inherent in the story?
 - How **IMPORTANT** are the **ISSUES**?
 - should anyone care?
 - List the **STAKEHOLDERS** and their interests
 - who are the players and their interests?

- List the TRADEOFFS made for any given solution
 - who wins, who loses, by what means?
- Look at extant ARGUMENTS (what do *other rational people* think about the issues?)
 - you've not yet decided on an answer, just survey what other smart people think
- Analyze to come to your own solution
 - based on previous analysis and basic rules

Anatomy of a Logical Argument

- Collect general principles and rules
 - codes of ethics, general ethical principles, laws, morals, commonly held beliefs
 - use strongest forms first!
- Collect relevant facts that raise an issue, form a question from that issue
- Apply the principles or rules to the facts
- You now have an answer to the question

- Facts: Dr. Turner wrote some code for a medical linear accelerator for a Varian machine. He later did a few modifications and got it running on the AECL Therac-25 to sell to them, explaining that he “wrote this specifically for your machine.” Bugs showed up that killed patients. AECL does not want to pay Turner for his work.

- Issue - does Turner deserve to be paid for the software?
- One Relevant rule (from SE Code) Software Engineers should not engage in deception regarding their software.
- Issue - did Dr. Turner engage in deception?
 - Yes. Prove it by using the “facts”
- Therefore, Dr. Turner’s actions were unethical
 - show this explicitly
 - whether he gets paid is a legal issue

“Correctness”

- What is “correctness” in Software Engineering?
 - meet spec?
 - “satisfy” “customer”?
 - capture a “market”?
- must be satisfied with rough methods for ethical analysis too
 - *compare this with software “formal” correctness*
 - See Leveson, Parnas, Hamlet, Knight, Kaner
 - » complete testing absolutely impossible
 - » formal proofs impractical and of limited value
 - » pointers back to “requirements” problem (validation?)

Computer “Science” ??

- Define “science”
 - consider theme central to “The Structure of Scientific Revolution” by Thomas Kuhn
 - natural science
- Sciences of the Artificial
 - “design science”
 - see Herb Simon’s work and others built on it.

Karl Popper's falsifiability criterion (epistemology)

- Any respectable scientific theory must be falsifiable, subject to showing it is untrue
 - “God is love” is not falsifiable
 - not a perjorative criteria
 - there are different ways of “knowing”
 - “The new Cal Poly IP policy explicitly favors ‘open source’” is falsifiable
 - so it can be “tested” for its truth objectively
 - just like the rules for Software Requirements