Minutes: August 8, 2012

* Irina’s paper: Development of a Laboratory Kit for Robotics Engineering Education
	+ Robotics program at the Worcesters Polytechnic Institute
	+ Robotics to introduce undergraduates to computer science – hoping to convince them to undertake graduate research projects in robotics
	+ Purposes
		- Introduce fundamentals of different engineering disciplines
		- Apply skills in real life
		- Using imagination and innovation to improve society
	+ 3 course levels
	+ Robotics approach was good, not because of the immediate feedback provided by the robots, but the multiple disciplines involved with it.
	+ Most students loved the courses
* Assessment for USC cs101 CLASSES – 3 sections
	+ Combine methods from research papers reviewed by the group
	+ Surveys, individual interviews, tests
	+ Must be able to obtain statistically significant data
	+ Use surveys across all sections
	+ Use interview responses as anecdotal fodder for publications/papers/etc.
	+ Cannot compare exams scores if questions do not match (eg: question specifically pertaining to Scribbler)
	+ Instead, compare answers to specific questions instead of overall exam scores
	+ Questions should be phrased in such a way that students from all sections (robot AND non-robot) can answer it
		- Eg: one concept is taught across all sections using the same guidelines – will students from the robot sections answer questions on that concept more accurately?
	+ The tests need to be the same length (to eliminate the factor of time allotted per question), but a designated number of questions need to be the same across all section tests (say 10 out of 15, for example)
* Blue pleo vs. pink pleo
	+ Blue is relaxing for people who are nearsighted
	+ Red/pink is a more exciting color
* Potential Scribbler projects
	+ Group alphabet project: each group assigned a letter (or two letters if they are easy to draw). Letters to be designed in the same fashion (x-y coordinate system) – combine code for all letters in the end to create an alphabet for everyone to use
	+ Starter project: having each group have the robot draw a team logo?
* Teaching pointers
	+ Best taught in a visual manner
	+ Idea:
		- Labeling the box ‘pointer’, labeling the robot ‘variable’
		- Typing \* makes the robot move out of the box (returns the variable)
		- Typing & makes the robot move back into the box (gives pointer)
* Practical concerns
	+ How to get everyone to run C++?
	+ How do we evaluate assignments? What data are we looking to collect?
		- Ask students to log how much time they spend on the project
		- Not all variables can be accounted for – not all should be considered. Which ones are the most important to collect data on?
		- Maybe don’t tell the students to log times – code the robots to run a timer each time they turn on and save it in an accumulating log
		- Eg: draw two concentric hearts, and have the students code the robot to draw a heart in between them (so there will be three concentric hearts) – assign grades based on how well they did that
* Need a YouTube channel for the class!
* Places to draw data from
	+ Surveys (2: pre-course and post-course)
	+ In-class activities [labs] (2 2-hr sessions per week)
	+ Homework
	+ TA Office hours
	+ Us (research group)
	+ Professor office hours
	+ HW projects
	+ Tests (2 – midterm and final)
	+ 3 hour lab to be required – looking for a studio classroom – meet 3:00 – 5:50 on Fridays
		- Group(s) to present on what they’ve worked on during the week (student team presentations)
			* Find out what they liked / what they learned
			* Forces students to form study groups and work together
			* How-to presentations / handout worksheets
				+ Have students fill in worksheet – force them to pay attention
				+ Maybe require that the presentations cannot have words – serve only as a visual aid, forcing the students to actually present their material
* Demo (as a final project?)
* What to have by the end of summer
	+ Toolkit (C++)
	+ Demos for the first day of class – showcase capabilities of robots
	+ Tutorials – video and written
	+ Documentation for code – using Doxygen?
* Website options
	+ How/where to post everything listed above? Wiki? Wordpress? YouTube? USC course website? Combinations and connections?
		- Usable, friendly, easy to find
* Progress
	+ Scribbler team:
		- toolkit prepared and functional – only need to add more demo functions
		- Ideally: change image for VMWare to bypass current configuration process
	+ Pleo team:
		- Sensor worksheet completed
		- Achieved a way to get real-time feed of sensors
		- Having trouble with Qt – Witty is also a struggle. Ideally, Witty to replace Qt