CISP 299: Experimental Offering in Computer Information science - Programming (Robot Programming in C)

Tak Auyeung, Ph.D.

August 14, 2004

1 About the Instructor

Name	Tak Auyeung
Office	#7 in Liberal Arts #133 (where Math used to be)
Phone	916-484-8250
Email	tauyeung@drtak.org
Website	www.drtak.org/teaches/ARC

2 About the class

Course Number	CISP 299
Course Title	Experimental Offering in Computer Information Science - Programming (Robot Programming in C)
Prerequisites	none
Number of Units	4.0
Hours	54 lecture, 54 lab
Class Section Number	18317
Course Description	This course investigates how autonomous robots can be programmed in the C language. Software techniques to interface with the input/output devices of a microcontroller are discussed. Special techniques to control motors and read sensors are demonstrated. Another topic is parallel threading using a real-time kernel. This class is project based; students complete the software component of mini-Sumo robots.
Class Meeting Time	MWF 1200-1250
Lab Meeting Time	MWF 1300-1350
Classroom	Rm 121
Lab	Rm 152 section C

3 Generic Information (All Classes)

This section contains information that is common for all classes, unless otherwise noted.

3.1 During Lectures

- Absolute no-nos
  - Eating *any* kind of food, including chewing gum and candies. If you have a medical condition that requires you to eat during lectures, please let me know ahead of time.
  - Drinking any kind of liquid. If you have a medical condition that requires you to drink during lectures, please let me know ahead of time.

  - Talking that distracts me or other students. This includes talking to another student, on a cell phone, and etc.
  - Behaviors that are disruptive, insulting, intimidating or otherwise unacceptable in a classroom. I may optionally warn, or ask people to leave a classroom without warning. In case of non-compliance, campus security will be called in.
  - Missing classes. If a class meets n times in a week, you can miss n classes before I drop you without warning. Exceptions can be made with my consent.
  - Do not print the classnotes in room 152. You have been warned! (Since you don't need to buy a textbook, print the classnotes somewhere else for a modest price. If you don't need a printout, save the money!)

- Not my problem
  - Falling asleep or otherwise not paying attention.
  - Forgetting to bring classnotes or books in an open-book exam.

- Encouraged
  - Participate in discussions. Answer questions, ask questions, correct my mistakes, etc.
  - Letting me address you before speaking.

3.2 During labs (only for classes with labs)

- No-nos
  - Doing anything that is prohibited, implicitly or explicitly, in the lab usage agreement.
  - Same policy about food and drink.
  - Using any lab resource for *anything* other than school work related to CIS, office technologies or business courses. If a student is spotted using a lab computer for anything that is not related to coursework, lab technicians and I may reboot/disable a computer remotely with or without prior warning.
  - Same policy about unacceptable behaviors.
  - Hogging my attention while other students are waiting. While there is at least one other student waiting, I can only give a student up to 10 minutes at a time. You can always line up again for another 10 minutes.

- Not my problem
  - Missing labs. Labs are optional (unlike lectures), but you may be able to do your homework assignments quicker and more effectively during scheduled labs.
  - Falling asleep, dozing off.
  - Leaving behind diskettes, USB flash drives, etc. Be sure to take all personal belonging with you. No one but yourself is responsible for your belonging.
  - Forgetting to save files before exiting applications or shutting down a computer.

- Encouraged
  - Asking questions related to homework assignments.
  - Helping other students without considered cheating.
  - Doing your CIS, business or office technology homework assignments.

3.3 Cheating

- Definitions:
  - "cheat" (general definition): engage in deceitful behavior; practice trickery or fraud (according to WordNet 2.0)
  - "cheat" (specific definition): to submit any work (quiz, examination, homework) that is not completed according to rules.
  - Unless otherwise noted, all submitted work should be completed independently and originally.
  - Independent means "by oneself", by whomever submits the work.
  - Original means "not derived or copied or translated from something else" (WordNet 2.0)
- But what about helping/getting help (for homework assignments, not quizzes or examinations)?
  - Discuss concepts and ideas, not answers.
  - Do not share specific applications of concepts and ideas for homework assignments.
  - Help others by asking questions, not by spelling out answers.
  - It's okay to clarify material already included in classnotes and textbooks.
  - Use examples that cannot be directly applied to homework assignments.
  - *Never* share files or print-outs of complete homework assignments with other students when work can still be submitted for points.
- How does Tak know?
  - I read all source code and look for similarities.
  - I check to see if homework performance is consistent with examination performance.
  - I proctor examinations and look for unusual behavior.
  - I compare examination answers and look for unusual similarities.
  - I reserve the right to ask a student explain submitted work.
  - I keep my ears open to reports and complaints from fellow students and other professors.
- So what happens to cheating students?
  - Any submitted work that is the result of cheating does not count to the final grade. No make-up will be offered.
  - Incidents are reported to the area dean.
  - Students who repeatedly cheat can be expelled.
- What if a student is "wrongfully accused"?
  - All complaints and appeals should be forwarded to the area dean.
  - The dean's decision will be implemented.

3.4 Resources

- Classnotes are available from my website
- Lecture recordings are also available (whenever I remember to record and upload)
- In-class notes are also available (not guaranteed to be complete)
- Email is an effective method to get in touch with me, better than phone messages.

3.5 Grading

- Examinations
  - No make-up exams unless I approve of reasons prior to the exam
  - All exams account for 80% of your final grade
  - First exam accounts for 20% of your final grade
  - Second exam accounts for another 20% of your final grade
  - The final exam accounts for 40% of your final grade
  - All exams are open book and open notes, but no cheating is tolerated
  - All exams are, by default, multiple choice. However, I reserve the right to change the form of questions.
  - All exams have some "extra credit", see below
  - Letter grade equivalence: given t_i is the total number of questions of exam i, b_i is the baseline number of questionsof the same exam, and r_i is the number of questions answered correctly:
    - F: $r_i < 0.1 \times b_i + 0.2 \times t_i$
    - D: not F and $r_i < 0.3 \times b_i + 0.2 \times t_i$
    - C: not D and $r_i < 0.5 \times b_i + 0.2 \times t_i$
    - B: not C and $r_i < 0,7 \times b_i + 0.2 \times t_i$
    - A: $r_i \geq 0.9 \times b_i + 0.2 \times t_i$
- Homework Assignments/Projects
  - All homework assignments/projects account for 20% of your final grade
  - +2% per day early
  - -20% per day late
  - I will use the last submission for grading
  - I reserve the right *not* to grade any submission that is not submitted according to specifications
  - Letter grade equivalence (s_i is the adjusted score of a submission i after early/late adjustments, p_i is the number of points for the work):
    - F: $s_i < \frac{p_i}{8}$
    - D: not F and $s_i < \frac{3 \times p_i}{8}$
    - C: not D and $s_i < \frac{5 \times p_i}{8}$
    - B: not C and $s_i < \frac{7 \times p_i}{8}$
    - A: $s_i \geq \frac{7 \times p_i}{8}$
- Final grade
  - Weighted average of all exams and homework assignments/projects.
  - Letter grade is based on boundaries determined by weighted average of all exams and homework assignments
  - Want exact formulae, here they are:
    - let $R = 0.2r_1 + 0.2r_2 + 0.4r_3$, where r_1, r_2 and r_3 are the number of correct answers of the first, second and final exams, respectively.
    - likewise, let $T = 0.2t_1 + 0.2t_2 + 0.4t_3$ and $B = 0.2b_1 + 0.2b_2 + 0.4b_4$, subscripts indicate exam number (1 means first, 2 means second and 3 means final)
    - Let S be the sum of adjusted scores of all homework assignments and projects
    - Let P be the sum of total points of all homework assignments and projects
    - F: $\frac{R}{B} + 0.2\frac{S}{P} < \frac{0.1B+0.2T}{B} + \frac{P}{8}$
    - D: not F and $\frac{R}{B} + 0.2\frac{S}{P} < \frac{0.3B+0.2T}{B} + \frac{3P}{8}$
    - C: not D and $\frac{R}{B} + 0.2\frac{S}{P} < \frac{0.5B+0.2T}{B} + \frac{5P}{8}$
    - B: not C and $\frac{R}{B} + 0.2\frac{S}{P} < \frac{0.7B+0.2T}{B} + \frac{7P}{8}$
    - A: $\frac{R}{B} + 0.2\frac{S}{P} \geq \frac{0.7B+0.2T}{B} + \frac{7P}{8}$

4 Schedule

8/16	Introduction, take role, rules
8/18	Microcontrollers, embedded systems, development tools
8/20	More development tools
8/23	Binary digital input and output
8/25	Transition detection, debouncing
8/27	More software debouncing techniques
8/30	Timer device, concepts and configuration
9/1	Interrupt service routines (in general), timer ISR
9/3	Applications of timer ISR as a timing basis
9/8	Stepper motors, stepper motor driver, basic stepping concepts
9/10	Stepper motor motion control, trapezoid profile, physics formulae
9/13	Stepper motor control implementation, frequency division
9/15	Acceleration logic, distance control logic, code organization
9/17	Practice exam 1
9/20	More code organization for stepper motor control
9/22	Exam 1
9/24	Differential steering, ccncept, angular and linear components
9/27	Implementation of differential steering, combination of speed control profiles
9/29	Phototransistors as sensors, advanced techniques
10/1	External interrupts, edge sensing
10/4	Analog to digital convertor (ADC), configuration
10/6	ADC ISRs, techniques to scan multiple channels, software layer separation
10/8	Triangulation sensors (GP2D12), noise problems, glitch filtering
10/11	More filtering, averaging noise, circular buffers
10/13	DC motors, concept of pulse-width modulation (PWM), configuration of hardware-based PWM
10/15	More on PWM, R/C servo control, problems with PWM
10/18	Rotation encoding, quadrature encoding, software techniques
10/20	Quadrature encoder interface techniques, filtering
10/22	Practice exam 2
10/25	Partial Integral Differential (PID) control loop, concepts
10/27	Exam 2
10/29	PID implementation issues, bit-width resolution, fixed-point computation
11/1	PID constant tuning, introduction to Micromouse competitions
11/3	Basic robot design issues, configuration, component selection
11/5	More on Micromouse design issues
11/8	Maze exploration and mapping algorithms, depth-first search
11/10	More mapping algorithms, floodfill
11/15	Map representation, set representation and implementation
11/17	Floodfill algorithm implementation
11/19	Introduction to micromouse simulator, interface description
11/22	Introduction to real-time kernel (RTK), basic threading
11/24	Multithreading concepts, round robin, priority queue, scheduling
11/29	Utilizing an RTK, thread coordination, semaphores
12/1	More RTK concepts, relationship of ISR and RTK, coordination of ISR and RTK
12/3	Application of RTK to robotics, such as Micromouse
12/6	Practice final exam
12/8	Student presentations
12/13	Final exam 1015-1215