CMPS 10
Introduction to Computer Science
Spring 2012

Grading Criteria

Theory and Concepts: 49%
- Class Presence (Required; see below) and Participation 10%
- Class Participation (or S/U homework or reading assignment) 11%
- Interactive Story (roughly due 5th or 6th week) 11%
- Peer-to-Peer Learning Evidence 1-page (due 7th week) 5%
- Why Computer Science 1-page PPT (due 8th week) 5%
- In-Class Presentation (1 min) (due anytime before 9th week) 2%
- In-class quizzes or “graded” homework 5%

Programming: 51%
- 4 Programming Assignments roughly 5, 5, 6, 7 (roughly due 3rd, 4th, 6th, and 7th week) 23%
- Final Programming Project 28%

Extra Credit
Over and above performance in “Interactive Story” or “Final Programming Project”

Revised Grading Criteria 04/25/2012
(3 programming assignments (instead of 4) and further refinement of Class Participation Points)

Presence, Participation, Collaboration, Homework, and Quizzes: 38%
- Class Presence (Required; see below) and Participation 10%
- Homework 6%
- Lab Participation and Engagement with Tutors/TA 5%
- Peer-to-Peer Learning Evidence 1-page (due 7th week) 5%
- or Instructor’s Discretion
- Main Homework: Why Computer Science 1-page PPT (due 8th/9th) 5%
- In-Class Presentation (1 min) (due anytime before 9th week) 2%
- In-class quizzes or “graded” homework 5%

Interactive Story 19%
- Interactive Story Part 1 4%
- Interactive Story Part 2 4%
- Interactive Story (roughly due 7th week) 11%

Programming 43%
- 3 Programming Assignments roughly 5, 7, 9 (roughly due 3rd, 5th, and 7th week) 21%
- Final Programming Project (due June 13, W, 7:30pm) 22%

Extra Credit
Over and above performance in “Interactive Story” or “Final Programming Project”
Revision based on mid-Quarter feedback: Common Grade Criteria for all students: **79%**

**Presence, Participation, Collaboration, Homework, and Quizzes:** **22%**
- Class Presence (first 10 classes) 5%
- Class/Lab Participation/Engagement/P2P/Effort/Interview 5%
- Homework (first 6 homework) 6%
- Quizzes (3 quizzes) 6%

**Interactive Story** **8%**
- Interactive Story Part 1 4%
- Interactive Story Part 2 4%

**Programming** **21%**
- 3 Programming Assignments roughly 5, 7, 9 21%
  (due 3rd, 5th, and 7th week)

**Final Programming Project** *(due June 13, 7:30pm)* **28%**
(objective is to create something that you are proud of).

Options for Remaining 21%
Objective is to allow students flexibility to showcase what they are most skilled at/enthused about:

(i) create value for peers through computer science education, and demonstrate awareness of importance of evolving role of computer science on society (Option 1),
(ii) create an interactive story telling of “Why Computer Science Matters” (Option 2), demonstrate the importance of computer science for society using a creative and scientific process (incorporating text, images, sounds, and links),
(iii) create a final impactful project of your choice relevant to the course (Option 3),
(iv) use traditional pathway of examination to demonstrate your competence (Option 4),

Option 1: Interactive story removed but small tasks maintained
- Presence 5%
- Homework 2%
- Quiz 3%

Main Homework: Why Computer Science 1-page PPT (due before 10th week) 5%
In-Class Presentation (1 min) (due anytime before 9th week) 2%
Peer-to-Peer Learning Evidence (due before 8th week) 4%

Option 2: Small tasks removed; interactive story maintained;
- Presence 5%
- Homework 2%
- Quiz 3%

**Final Interactive Story Proposal** *(proposals due 8th week)* **11%**

Option 3: Small tasks removed and Interactive Story replaced with student proposals for ONE BIG TASK
- Presence 5%
- Homework 2%
- Quiz 3%

**Final Conceptual Project Proposal** *(proposals due 8th week)* **11%**

Option 4: Presence, homework, quiz, and interactive story replaced with examination
- Examination (June 6, 5:00-6:45pm) 21%
(still **required** to participate in-class in-person on May 30 for class survey at the beginning)
Examples of “Conceptual Project” (Intellectual Creative Innovative Thinking)
Answer a question that is of value to instructor, peers, or society

1. What is “Computational Thinking”? Is it as important as Reading, Writing, and Mathematics? Why or Why not?

2. Take a clear and strong position on that “Introduction to Computer Science” class must be taught on-line. Justify your position.

3. Why is the 21st century learner fundamentally different than the previous generation of learners? What role has computers played in this transformation? What critical pedagogical changes need to be incorporated in order to address this emerging need?

4. A critic of “Scratch” being used in the university curriculum just opined, “You are short-changing the students by teaching them an easy programming language “Scratch”. You ought to be teaching a more advanced programming language.” How would you justify, “Why Scratch in an Introductory Computer Science Class at the college level?”

5. A critic of “TouchDevelop” being used in the university curriculum just opined, “Programming language on a mobile platform and that too on Windows! Naaaaaaa. Why not iPhone or Android or just a regular programming language.” How would you justify, “Why TouchDevelop in an Introductory Computer Science Class at the college level?”

6. Why is College Board working on introducing “AP Computer Science Principals”? What it different than “AP Computer Science”? Why shouldn’t be just teach the old regular computer science – algorithms, operating systems, programming languages, hardware, and databases?

7. “Computer Science” is a science. “Computer Gaming” is entertainment. “Business Management” is about entrepreneurship and making profits. An interactive story telling project belongs to a creative writing class and not to a computer science, computer gaming, or business management class. Argue against this thesis.

8. Discover the efforts at the University of Washington, Seattle (by Larry Snyder) and at the University of California, Berkeley on teaching “Joy and Beauty of Computing” (aka Computer Science Principles). Can these efforts be adopted at the University of California Santa Cruz? Why or why not? What will be the main challenges?

9. What is a “Turing Machine”? Describe a problem, if any, that can be solved by computers and cannot be solved by a Turing Machine.

10. Reflect on the creative thinking of a few leading computer science entrepreneurs of our generation. What problems did they set out to solve?


12. Provide value-added concrete feedback on CMPS 10 class on how to improve teaching? Examples include but not limited to (i) concrete revision for Quiz 2 (potentially a new handout) to teach the “importance of commenting in a computer program (rather than “shock and awe” methodology), (ii) concrete revisions for homework to describe requirements for “Interactive Story Telling” Projects, (iii) a list of excellent questions on scratch/touchdevelop or a handout summarizing essential concepts of scratch/touchdevelop.