Syllabus for CSI 35

1. GENERAL COURSE INFORMATION

   Instructor: Kerry Ojakian (*Email*: ojakianteaching@gmail.com).

   Website: http://fsw01.bcc.cuny.edu/kerry.ojakian/TeachingPages/TeachingMainPage.html

   See website for other course information (book, office hours, etc)

2. TESTS

   - There will be 3 tests during the semester, and a fourth exam (the final exam), after the semester. The lowest of your 3 in-class tests will be dropped completely. The final exam cannot be dropped.
   - For each test, you are responsible for all material covered prior to it. The final exam will be comprehensive, covering all the course material.
   - There are no make-up exams.

3. IN-CLASS WORK

   - We will have regular "in-class" assignments (sometimes done in groups). They will be worth 5 points, with a possible grade of 3, 4, or 5.
   - If you are not present for a class when an in-class assignment is being worked on, you will receive a 0 for that assignment (except under extenuating circumstances cleared by the professor). However, if you are present and put in some work, you will get at least a 3.
   - Your lowest in-class assignment grade will be dropped.

4. HOMEWORK

   - There will be two kinds of homework: Paper-and-pencil homework, and a small amount of WeBWork homework. The WeBWork homework is done online, on a computer. For the paper-and-pencil homework, there will be 6 in total, two due before each of the exams.
   - Late homework is not accepted. However, your lowest handwritten homework grade will be dropped. I also essentially drop your lowest couple of WeBWork homeworks; in fact I do better: I simply add 5 % points to your WeBWork homework score at the end of the semester (though your score can't go above 100%).

5. GRADING

   Your final numerical grade will be determined as follows:

   (a) Final Exam: 30%
   (b) In-Class Tests: 42% (i.e. your 2 highest tests count 21% each)
   (c) Homework: 18%
   (d) In-Class Assignments: 10%

   From the final numerical grade, the guaranteed minimum term grade is as follows:

   A- for 90 or above, B- for 80 or above, C- for 70 or above, D- for 60 or above

6. AI Help

   Yes, you may use ANY online computer tools and AI for all homework and classwork! However! - 1) You may not simply copy a verbatim solution, and 2) On tests you may use NO electronic devices whatsoever!

   Look above: Tests are worth 74% of your grade! Use computers/AI to learn or you will fail the tests ...

7. EXPECTATIONS

   Respect. You are expected to show respect for the other students, the professor, and the class environment. In particular, behavior which disrupts the learning of other students will not be tolerated (for example, engaging in a conversation while I am lecturing).

   Class Attendance. You are responsible for everything discussed in class. If you miss class, it is *your responsibility* to find out what you missed. There will be in-class work, which you can **not** make up if you miss class.

8. ACADEMIC INTEGRITY

   Academic dishonesty (such as plagiarism and cheating) is prohibited at Bronx Community College and is punishable by penalties, including failing grades, dismissal and expulsion. For additional information and the full policy on Academic Integrity, please consult the BCC College Catalog.

9. ACCOMMODATIONS/DISABILITIES

   BCC respects and welcomes students of all backgrounds and abilities. In the event you encounter any barrier(s) to full participation in this course due to the impact of a disability, please contact DisAbility Services as soon as possible this semester. A Disability Services specialist will work with you to review the barriers you are experiencing and explain the eligibility process for establishing academic accommodations for this course. You can reach DisAbility Services by email at disabilityservices@bcc.cuny.edu or by phone at (718) 289-5874.

10. LEARNING OBJECTIVES

    Objectives: A successful student in this course will learn to 1. classify basic discrete structures, 2. use graphs and trees as models and tools for studying computational complexity, 3. analyze finite and infinite structures using mathematical reasoning and tools of first order logic, 4. design and analyze algorithms, in particular those based on recursion and iteration, 5. prove formal statements using mathematical induction, 6. use mathematical induction in verification of program correctness.