# CS341/CM339 Fall 2009

# Assignment Guidelines

## General Requirements

The solutions you hand in must be your own work. In particular, **you are not allowed to look up the solutions in the literature or on the internet**. Violation of this rule is equivalent to cheating.

Assignments will involve mostly written work. Please write legibly and staple the pages of your solutions securely (typeset solutions are preferred, but not required). Handwritten solutions should be done either in pen, or (if you prefer to use a pencil) make a photocopy of the assignment and submit the original and photocopy together.

**Assignments done in pencil but not accompanied by a photocopy will be ineligible for mark appeals.**

Put your full name and ID number on the first page, and put the **first two characters of your last name** in big capital letters on the top right-hand corner of the first page for ease of sorting and searching (for example, John Doe would put letters DO on the top right-hand corner). Also, specify on the first page of your assignment the section of the course (1 or 2) in which you want to pick up your assignment.

For programming questions, the written part and **both printouts of the source code and electronic submission are required**. More details on programming questions can be found below.

Assignments are to be placed in the CS341/CM339 assignment box located on the 3rd-floor of MC next to the elevators by the bridge to DC. Please ensure that you place the assignment into the correct slot by 5pm on the day they are due. Late assignments will receive a penalty of 20%. NO LATE assignments will be accepted after 5:00pm on the day following the due day. In case of genuinely extenuating circumstances such as serious illness, please let us know as soon as possible.

We highly recommend that you start early and hand in your assignment when it is completed, instead of waiting until the last moment.

## Non-programming Questions

Ensure that your solutions are complete and mathematically precise, and at the same time, easy to understand and to the point. Your solutions will be judged not only for correctness but also for the quality of your presentation and explanations. You should provide reasonable justification of any steps in a solution unless the question states otherwise.

In questions that involve the **design of an algorithm**, you should (unless stated otherwise) design the best algorithm that you can. The first criteria for marking is the algorithm's correctness, the second criteria is its efficiency. Thus, an algorithm which is slow but correct will receive substantially more marks than an algorithm which is fast but incorrect. Inefficient algorithms will not earn full marks but part marks will usually be given. In your solution, enclose the following:

- Describe the main idea first if that is helpful.
- Present clearly written pseudo code (at a level of detail mimicking the style of the lectures, the model solutions, or the book).

- Give a correctness proof/argument if it is not immediately obvious.
- Include an analysis (usually, of the running time).

## Programming Questions

Programming questions will usually involve designing an algorithm and implementing it. The above mentioned guidelines and requirements for algorithm design questions are to be fully applied to programming questions as well. The rest of this section deals with the instructions for the programming part of the questions.

Your program should be implemented in C++, or Java, within the undergrad UNIX environment.

**If you are working on a different platform, it is your responsibility to ensure that your program runs properly on the undergrad environment.**

The program should reside in a single file with the name of the file having the form: **filename.c** (if in C++), or **filename.java** (if in Java), as given in the assignment question.

Your program should be submitted both **on paper and electronically**. For electronic submission, use the **`submit`** command, e.g., by copying the required file(s) to the current directory and typing

```
submit cs341  assignment_number
```

(see man page for the details). Submit only your source file(s) -- no executable files please!

We will compile your program using one of the command lines:

```
g++  filename.c  -o  filename
```

or

```
javac  filename.java
```

(main class should be named: filename.class). We will run your program using one of the command lines:

```
./filename  <input_file  >output_file
java filename  <input_file  >output_file
```

Since we will compile/run/test your program using an automatic script, it is vital that you follow our instructions to the letter. In particular, your program should read from the standard input and write to the standard output, in the format exactly as specified in the assignment question. It is a good idea to test your program thoroughly yourself (on your own input) to ensure its correctness. Some additional pre-testing might be available. If so, this will be announced in the course newsgroup.

Marking of programming questions will be based on both correctness (as determined by our test runs) and coding style (documentation, design, clarity, etc.).

## How to Get High Marks on Your Assignments

*Adapted from the text by Prof. Jeff Shallit:*

1. Write legibly, in pen, not pencil. If possible, type your solutions, or use a word processor. If you use pencil no appeals will be accepted.

2. Write the answer to each question on a separate sheet of paper. If you have left some problems unsolved, say so. Staple your pages together in order; do not use paper clips.

3. Make your answers clear and concise, but don't omit details. Be sure your arguments will convince a skeptical (but intelligent) TA.

4. If a final result is requested, show all your work to get that solution, and put a box around the final answer.

5. Do not try to fool the TAs! The following phrases are tip-offs: "Obviously. . . ", "It can be easily shown that. . . ", "It is clear that. . . "

6. When doing a proof by induction, state precisely what your induction hypothesis is and on what variable you are doing the induction.

7. If you are unsure whether your solution/proof is correct or not, you need to work out more details and write them down in your solution.

8. Make your solution look more like the text from the text book rather than the concise notes from the class.

## Cheating

*Adapted from the text by Prof. John Wainwright:*

Cheating on assignments and projects includes copying another student's solution and submitting it as your own, allowing another student to copy your solution, or collaborating excessively with another student. The standard penalty for cheating on an assignment or project is as follows: a grade of **minus 100%** will be assigned for the submission, with a minimum deduction of 5% from the final course grade.

With assignments it is possible, and indeed desirable, to discuss methods of solution of problems with classmates, TAs, and instructors. However, the solution that you submit must be worked through by yourself and written in your own words. We recommend that you:

(i) Do not take any written notes during a discussion.
(ii) Wait several hours before writing your solutions (to let details fade).
(iii) Never show your write-up to other students.
(iv) Always acknowledge any discussions you have with others (or any consultation with other sources).

All academic offenses are reported to the Associate Dean for Undergraduate Studies. A second academic offense will lead to the student being suspended from the university for at least one term and may lead to expulsion.