



The University of
Chicago
Department of
Computer Science

CMSC 16200 – Honors Introduction to Computer Science 2
Winter Quarter 2007
LAB SESSION

Lab instructor

Name: Borja Sotomayor

E-mail: borja@cs.uchicago.edu

Office: Ryerson 257C

Lab: Mondays 3:30-4:50 in JRL A01C (Maclab, A-level of Regenstein Library)

Office hours: By appointment. I'm usually available from 9am to 6pm (except Thursdays, from 9am to 4pm). Walk-ins are fine, but please take into account that I might have to politely turn you away if I am too busy. Whenever possible, please give me a few hours notice before dropping by, to make sure I'm available.

Lab website

The lab website can be found in the *LabTA/Labs* section of the class wiki:

<http://brick.cs.uchicago.edu/~stuart/Courses/CMSC-16200/>

The main purpose of the lab website is to provide PDF versions of the handouts, along with any additional files needed to solve the lab exercises. A bibliography for the lab exercises is also included.

Mailing list

The course has a mailing list which can be used to ask questions and share useful information with your classmates. In fact, we encourage that *all* questions about homework assignments, labs, and course contents in general be sent to the mailing list, and not directly to the instructors and TAs. The reason for this is that, this way, all your classmates will be able to benefit from the reply to your question. In some cases, some of your classmates might even pitch in to provide their insights into questions or issues raised in the mailing list.

You can subscribe to the mailing list in the following web page:

<http://mailman.cs.uchicago.edu/mailman/listinfo/cmssc16200>



Lab organization

Throughout the quarter, the lab sessions will be structured in different ways, but they will all share these common traits:

- Attendance to the lab sessions is *mandatory*.
- You will have 48 hours to solve the problems presented during the lab session (i.e., the deadline is Wednesday at 5pm)

In some labs, you will also be asked to hand in, at the end of the lab session, whatever work you have done up to that point. During the additional 48 hours you will be able to polish up your solution, solve additional exercises, and document your work. You will be evaluated *only* on the basis of the work handed in on Wednesday. The reason you are required to hand in your work at the end of the lab is because you will usually be asked to write documentation explaining how you've improved your solution in those 48 hours. So, take into account that it is perfectly OK to hand in an incomplete (or even nefariously bad) solution at the end of the lab session, as long as you document what steps you have taken to improve it.

The main difference with the homework assignments is that most of the labs will involve some degree of teamwork. In particular, the labs will be organized in one of the following three manners:

- **As a group:** You will be asked to solve an exercise as a group of three during the lab session. You must hand whatever work you have done at the end of the lab session. After the first handin, you will be expected to continue working with your group to polish up your solution.
In this case, you will be graded as a group, not individually.
- **Peer review:** You will be assigned an exercise which you must solve individually during the lab session. You must hand whatever work you have done at the end of the lab session. After the first handin, you will be arranged into groups of 4 and members will be expected to discuss and explain their solutions. The group, in turn must provide feedback and constructive criticism. Based on this, you will polish up your solution, clearly documenting the feedback you received. You are free to disregard suggestions made by your classmates, but must explain why you chose to do so (e.g., you might have thought of a better solution later on).
In this case, you will be graded individually.
- **Mini-Project:** You will be arranged into groups of 4 and will be assigned a complex exercise composed of several components that can be developed and tested



individually, but must be pieced together at the end to produce a working program. During the lab session, you are expected to come up with an “attack plan” explaining what task has been assigned to each group member, outlining the specific subtasks of each task, and how you will piece all the components together. If you have time to start developing during the lab session, you must also hand that in. After the first handin, each member must individually develop the component assigned to him/her, and the group must meet at the end to put everything together.

In this case, you will be graded both on the basis of your individual work and on the basis of the final solution handed in by the group.

Finally, don't be intimidated if:

- The explanation of these three types of labs seem too diffuse or abstract. Lab handouts will clearly specify what you need to hand in (e.g., you will not be simply asked for an “attack plan”, we will provide a simple template of an attack plan for you to fill in).
- It seems like an overwhelming amount of work. We will not be giving you 48 hours worth of work! If you plan your time correctly, and leverage all the skills in your group, you should be able to finish most labs with only a couple of extra hours of work (beyond the lab session).



Handing in the lab

For a lab, you will be expected to hand in the following items:

- **Lab grading sheet.** Each lab handout includes a lab grading sheet where you will need to write down your name and student ID. The instructor will write down your grade in this sheet and return it to you. You can keep the rest of the lab handout.
- **Source code and documentation.** All the code you've written, and any accompanying documentation (if required by the lab). You will hand this in using the `hws submit` command (described below)
- **Printout of source code** [Optional] If you provide a printout of your code and documentation, the lab instructor will return it with annotations and comments, along with the grading sheet. If you do not, you will only get back your grading sheet (which will make it difficult for you to see what you did right and what you did wrong)

hws submit

This command is available if you log into any of the Linux machines in the Maclab. Make sure all the files you want to hand in are inside a directory, and then run `hws submit` like this (where `dir_name` refers to the directory you want to submit):

```
hws submit cmsc16200labs dir_name
```

For example, assuming that you are currently inside your home directory, and that you placed all the files for a particular assignment in directory `/home/myusername/hw01`, you would run `hws submit` like this:

```
hws submit cmsc16200labs hw01
```

Printing code

An easy way to print code from a UNIX system is using the `enscript` command. This command will automatically format the code for you, and can handle most programming languages. For example, to print out BASH code, you could run `enscript` like this:

```
enscript hello.sh -P printer_name -Esh
```

Where `printer_name` is the printer you want to send the code to. See the `enscript` man page for more details on using this command, and for a list of languages that `enscript` can handle.

Handing in printed matter

You can hand in your grading sheet, documentation, and optional code printout in my office (Ryerson 257C, slide it under the door if I'm not there)



Grading

Each lab will be graded on the basis of the code you hand in (correctness, elegance of the solution, etc.) and the documentation. Each lab will specify the point weight of each.

Late Submissions

- You are allowed one 24-hour extension, without any penalty, to be used at your discretion on one of the labs. The lab instructor must be explicitly notified when you use your free extension. Otherwise, your lab will be counted as a late submission.
- Late submissions will have a 50% point penalty if submitted no later than 24 hours after the deadline. After 24 hours, the lab will not be graded and you will not get any points for it.
- You are not allowed to hand in a portion of your work before the deadline, and the remaining work after the deadline. Handing in *any* work after the deadline automatically flags your entire submission as late.
- Four late submissions (not counting the free 24-hour extension) will result in a zero in your final lab grade.

Academic honesty

The University of Chicago has a formal policy on academic honesty which you are expected to adhere to:

http://www.uchicago.edu/docs/studentmanual/academic_honesty.shtml

In brief, academic dishonesty (handing in someone else's work as your own, taking existing code and not citing its origin, etc.) will *not* be tolerated in this course. Depending on the severity of the offense, you risk getting a hefty point penalty or being dismissed altogether from the course.

Even so, collaboration between students (outside your group) is certainly allowed (and encouraged) *as long as you don't hand someone else's work as your own*. If you have discussed parts of an assignment with someone who is not in your group, then make sure to say so.

As for consulting other sources, this can be a thorny issue, as programmers are always encouraged to not "reinvent the wheel" and reuse as much code as possible. However, in this course you are also required to show that you are capable of effectively coding a solution to a given programming problem. So, in no case will you be allowed to hand in an existing solution to a problem (e.g. taken from a website) even if you *do* cite its origin. In general, you *are* allowed to look at existing code in search of inspiration, as



**The University of
Chicago**
Department of
Computer Science

CMSC 16200 – Honors Introduction to Computer Science 2
Winter Quarter 2007
LAB SESSION

long as you cite the sources you consulted. For large programming problems (which will turn up later in the course), you *will* be allowed (and encouraged) to directly use code taken from other sources (web sites, programming books, etc.) as long as this code is used to solve a particular subproblem and not the entire problem.

If you have any questions regarding what would or would not be considered academic dishonesty in this course, please don't hesitate to ask the instructor.



Practical information

Obtain a CS account

Lab sessions will take place in the Linux section of the Maclab. Before using those machines, you need to request a CS account. This account will allow you to access certain computing resources in the Department of Computer Science, most notably the Linux machines in the Maclab. You can claim your CS account here:

https://www.cs.uchicago.edu/info/services/account_request

Knowing your way around a UNIX system

Although you will have the option of doing your homeworks and labs on UNIX or Mac in the Maclab, all lab instructions will be given from a UNIX system. If you are completely new to UNIX, we encourage you to use the KDE desktop, which provides a graphical interface very similar to the ones found on Windows and Mac systems. However, at a certain point, you will need to perform certain actions from the UNIX *command line interface* (or “console”). The first lab in the course will provide a basic introduction to the UNIX console. However, if you want a more complete introduction, you can take a look at the following tutorials:

<http://support.uchicago.edu/docs/misc/unix/general/feet.html>
<http://support.uchicago.edu/docs/misc/unix/tutorial/>

Working from home

Although the Maclab provides an excellent work environment, with all the software you need to complete the lab exercises, you are certainly free to work from home. However, take into account that you will need to hand your homework in using the `hwsubmit` command described above. Since this command is only available in your CS account, you will need to *log in remotely* to a CS machine using SSH. Instructions on how to do this are available here:

http://www.cs.uchicago.edu/info/services/new_users_guide

Also, take into account that there are two ways of working from home:

1. If you have a UNIX system at home (such as a computer with GNU/Linux installed in it), you can do your homework assignment entirely in your machine, using whatever tools you prefer. Once your assignment is complete, you simply have to copy your files to your CS account home directory and submit them using `hwsubmit`. Just in case, you might want to make sure that your code works fine in your CS account, as it



**The University of
Chicago**
Department of
Computer Science

CMSC 16200 – Honors Introduction to Computer Science 2
Winter Quarter 2007
LAB SESSION

will be graded in one of the CS Linux machines.

2. Regardless of having a UNIX system or not at home, you can do all your work using your CS account. To do this, you will have to log into your account using SSH and then do the assignment using the tools and commands available in your CS account.