Experiences of Successful Students

COMP 105
Fall 2018

To succeed in 105, you not only have to work hard; you have to work in the right way. This document presents the study practices used by the COMP 105 students who earned the highest grades in the Spring 2018 semester: A or A+. (Not every student who earned an A or an A+ chose to respond.)

I followed what the instructor and TAs suggested—try to work on assignments every day, but for at most 2 to 3 hours. When I did longer chunks of work (4-5 hours in a day), towards the end I was significantly less productive. If you choose to do a longer chunk of work, break it up by leaving Halligan and eating a meal, or going to the gym and doing a workout. Time management is a significantly important skill in this course, and if you do not manage your time well, it can make finishing assignments on time a living hell.

In terms of just understanding the course material, following the algebraic-law protocols at the beginning of the semester really helped me understand how to write functions in a cleaner and more straightforward way. It may have taken a bit more time at first but it helped me understand the material better. The best way to study for the exams is to do well on the homeworks—if you put in a lot of effort into learning the material when doing the homeworks and do well on them, you won’t have to learn as much material right before the exam. In helping you understand the new material for a homework, CQs are your friend, and it makes a big difference when you can get through all of them before an assignment, as you will understand how to begin to tackle the larger problems better.

105 is a long but winnable battle. When you finish an assignment, give yourself something. You’ve earned it! After you’re done with your pint of ice cream or movie or whatever else you enjoy, crack open the next assignment as soon as possible. Try (truth be told I did this maybe half the lectures) to read a bit of what the next subject is before Norman discusses it in lecture. When reading, cover as much ground as you can before revisiting tough parts—there are many chances for that in lecture, recitation, and the Halligan kitchen. Understanding is a multi-step process of reading, listening, questioning, and doing, and some parts are going to make very little sense before they start making sense.

Take Norman and the course staff at their word. They are very fine people and teachers and they will do whatever they can to help you. Ask about what you don’t know, but in between questions, take some time to check in with Piazza and with the book. If you’re with your partner, don’t be afraid to go slowly—the answers derived on the whiteboard are many, and in one instance, a partner and I even sat there reading for an hour straight. The amount of code written is merely a heuristic for, not a measurement of, your success. And if at any point you feel angry, sad, or frustrated, walk away. You’ll remember you’re free to be human, that computer science is challenging because it’s worth pursuing, and you’ll come back stronger for it.

I didn’t really do anything special or extra in this class aside from extra credit on one of the assignments. What was really important for me was attending every recitation and lecture, and taking notes during the lectures. As the semester went on there were fewer and fewer people showing up. You can certainly get away with missing classes, but it places an extra burden on you to learn more out of class.

Plan to spend more than 10 hours per week on homework once they get harder. I didn’t start any of them super early, but I also didn’t start any of them on the last day—that would’ve been a disaster. I kept track of when I had time to work—it helped me schedule meetings with partners, and it also helped me make sure I had enough time to finish each assignment.

The study strategies I used:

1. Doing all the recommended readings thoroughly. I think this is probably the most important thing that got me through the assignments. I always gave myself plenty of time to do all the listed readings, and made sure that I understood the connections between the readings and the assignments. I then did reading-comprehension questions to test my knowledge. This way, I felt that I built a solid foundation before I started each assignment.

2. Starting early. Starting early gave me a chance to really know and love the assignments instead of stressing through and hating them.

How I tackled the work:

1. I always wrote unit tests as I went. Knowing that what I already wrote was bug-free gave me lots of confidence and a good foundation to tackle the next problem.

What enabled me to succeed:
1. Trust in the class. When I got stressed out, I would tell myself that 105 is a reasonable class and is designed so that if I started the work early, I could finish. I think this trust got me through many stressful times.

2. Attending recitations. I really appreciated the recitations. They were very useful hands-on experiences, great starting points for the assignments. I always felt more confident about the assignments after attending recitations.

What was most tip for me was to start each new homework right after finishing the current homework, even if the lectures hadn’t covered the materials yet. This strategy not only provided me more spread out time to work on each homework, but also helped me gain more from each lecture since doing the reading comprehension helps me understand materials discussed in the lectures much more quickly.

I would also recommend each student to try their hardest to understand the material of each homework deeply rather than just try to finish the homework without deeply understanding the materials. This helped me tremendously in exam preparation.

An unusual approach that works only for rare students. —Editor

The key to success in the course is simply to read through your textbook. If one reads the textbook containing the relevant material they should be more than prepared to take on the exams/assignments. There’s no trick to it, no need for TA’s or other students or even lectures; everything is in the book, it’s about your willingness to search for it. The class isn’t particularly hard, the material is extremely logical, and the professor lays out the material extremely well. As to study strategies, studying isn’t really necessary given the amount of repetition on the homework. They really nail the ideas into your head. If you’re on top of your homework, you’ll succeed in the course.

For homeworks, I did the reading comprehension questions (CQ) first. I patiently studied the book, following guidelines outlined in the CQ. Consequently, I was able to tackle the problems. Often, I made mistakes on my first try. But I was only able to make an effort because I somewhat knew the material. For better results beyond my personal efforts, I discussed my solutions with TAs who appreciated my honest efforts and guided me to better or clearer solutions if needed.

As someone who’s worked in multiple 105 study groups, I realized that success in 105 is tied to both your mindset and motivation to learn the coursework. Because the material in 105 is unique and mostly theoretical, it comes across as immensely difficult. When I felt like skipping a required section or sub material, I did not. Unlike in many other classes where you can get away, in 105, you’re likely to fall into a trap. 105 builds on previous material, so I took my time to learn the skills each homework (particularly reading CQ) was to teach me. Also, I spent more time learning than writing code. When pressed for time, I was able to write meaningful code because I understood the theoretical rules I was translating or the problems I was trying to solve.

Besides working hard and giving yourself enough time to complete assignments, the most unique advice I could give would be the following:

Be the best programmer you can possibly be before taking the class. COMP105 was the last class I took to complete the major and there should be no rush to take it if you’re not ready for it. After all the other coursework, internships, and side projects I completed over the past few years, I approached the class with more confidence in my abilities, more relaxation when dealing with stressful, long, and difficult assignments/concepts, and was just simply a better computer scientist/programmer/problem solver. I don’t think I would have been able to work towards the grade I did had I taken this class my sophomore or junior year.

1. Start working on the homework as early as possible. It worked very well for me that finishing reading before recitations under the guidance of the reading-comprehension questions, which helped me make the best of recitations.

2. Focus on reading the textbook. I always read the relevant contents repeatedly and I benefited a lot from a thorough understanding of the topics.

3. Ask on piazza or go to TA’s office hours when you get stuck. It can help you save a lot of time.

4. Lectures and recitations are very important. I never missed any of them.

5. Don’t use your tokens on the first three assignments.

I tended not to immediately go to TAs for help but first try to figure out the problem on my own. When I got stuck, I first went back to the materials to make sure I understand the question correctly. Then I would experiment with several approaches I considered reasonable. If none of these work, I would then go to TAs as my last resort. However, I would ask them freely for general, conceptual type of questions for clarification purposes. I realized the most important part for these assignments was understanding the questions correctly. Once I got the idea of what I’m asked to do, the rest would go smoothly.

I also found it helpful to start reading the homework questions early. I didn’t find it helpful to start coding at once. I spent more time reading and understanding the questions at the early stage of each assignment. I also found it slightly easier to start programming after the recitations.
The main strategies that allowed me to succeed in COMP105:

- **Starting every homework as early as possible.** Even if we hadn’t completely finished covering everything in lecture, I made an effort to complete the reading comprehension within the first two days of the assignment being released. This allowed me to have the rest of the week to slowly chip away at the assignment.

- **Going to Halligan to work.** I never did COMP105 homework by myself. I asked TAs for help early on in the assignment, which helped me understand concepts faster and enabled me to avoid a lot of pitfalls along the way.

- **Pacing myself.** Understanding and debugging problems can take a long time. I tried not to work for more than a couple hours at a time before calling it quits for the day. (Note: I was able to do this by starting early. It’s a lot harder to pace yourself when the deadline is looming.)

- **Bookmarking useful sections in the textbook.** There’s a lot of information in there, and it’s hard to remember where certain pages are. It was also very useful when I was studying for the midterm and final.

To study for 105 I primarily followed the study guide and rewrote important information by hand on a sheet of paper, organized by topic to help me conceptualize the material. I found that my enjoyment of the class and my productivity were much higher when I started assignments soon after they were released and worked with my partner for two to three hours per day, though when I had a lot of other commitments this strategy wasn’t always possible. Working this way helped me feel like 105 was just a part of my regular schedule rather than a big assignment I needed to complete, and it allowed me to feel less stressed if we didn’t make as much progress in a day as we’d hoped. I also found working in Halligan with the TAs and other 105 students to be really enjoyable and social, which kept the work from feeling like “work.”

My approach to much of the work for this class (though this might not apply as much to some of the earlier assignments that consisted of many short functions) was to consider the work as an iterative process. Many times I would start the a problem by writing the algorithm/approach up in plain English or pseudocode. Then if I was still confused I would try to implement that algorithm roughly in a language I’m familiar with like C++ just to think about the finer points. From there it was writing code in the language that was relevant to the assignment, but not trying to do it all at once. Many times I would write something that was pretty close to the solution, but it wasn’t right so I would just start over. Already having attempted the problems it was easier to do the second and third and so on times. For me I guess it was just being okay with scrapping what I had, in order to start fresh.

Doing well on COMP 105 homeworks is critical and carefully budgeting your time helps. I always completed at least the reading-comprehension questions before recitation. Since the content of a recitation is usually similar to that of a homework problem, I found it easier to complete the coding portion of the homework after recitation. Be sure not to start the homework too late, however; since everyone codes at their own pace, you will have to find the start time that works for you and your partner. When I was stumped on a homework problem for more than an hour, asking a TA or posting a conceptual question on Piazza often helped me find the answer faster.

The two things that helped me the most in 105 were reading the book and starting early on assignments. The book is easily the most helpful tool offered, as it has just about everything that’s needed to understand each language and concept so to tackle the homework. Even if you understand everything perfectly, the assignments still take many hours to complete, so starting early is crucial. These homework problems are not cramammable and coding while panicked is the least productive way to attempt them.

Some people just don’t benefit from going to lecture and that is completely fine—I personally find it hard to keep my focus during lecture. It is important though to put in enough time into this class especially since by not going to lecture, you’ve taken the onus of your learning more deeply upon yourself. Spend time with TAs after recitation has ended, ask lots and lots of questions in office hours, and just really engage with the material! If you can motivate yourself to really sit down and get to know the coursework, 105 is an amazing and enriching journey!

More concretely, no matter what kind of a learner you are, PLEASE go to lectures starting from lambda calculus and onwards. That material is really important both in 105 and beyond and the best learning happens in lecture simply because it is difficult material to wrap your head around and you could do with all the exposure to it you can gather! Finally and most importantly, give yourself time to learn, know when to call it a day, and know how and when to really push yourself.

P.S. A lot of the extra-credit material in this class is very interesting and you will probably not get a chance to enjoy those later on so seize the day!

Here’s what I did:

- I asked what I thought were “dumb” questions (that didn’t turn out to be dumb at all, much to my surprise!). When you’re insecure about where you are on a project, it’s challenging to speak up, but flailing with syntax or something silly is beside the point. When I was stuck on something that wasn’t useful to my learning, I made sure to get help.
• I would have not done nearly as well if I had not gone to Halligan as often as I did. I logged a lot of hours for the class, but not nearly as many as I spent debugging in COMP 40 (or even COMP 15!). Eventually, I got to the point where I only worked in Halligan. I never wasted time being stuck, instead asking questions to my classmates and TA’s about what I was having trouble with. I stopped convincing myself that I would work better at home (which is almost never the case), but focused on working productively in Halligan. I would say I spent around 20 hours per week on 105, but there were only a handful of nights this semester when I stayed up past midnight coding (for me, a great feat).

• Though it was a hard lesson learn, I learned to do the reading questions early. More importantly I asked the TA’s about the CQ’s, and I made sure that I actually learned something before moving to the coding assignment.

• Starting early is very necessary (not new advice but still helpful). Doing the CQ’s before recitation, if nothing else, is really helpful. If you go to HALLIGAN early, you get more access to the TA’s with much less chaos.

• I learned to schedule 105 in spans of 2 to 3 hours. I went to Halligan between classes, but always with a time that I wanted to stop working. There were some nights that I had to work past my “end time,” but mostly I ended up leaving when I wanted to do other work, and while I did 105, I was extra productive.

• I worked with other people throughout the class—working alone can be scary and unnecessarily challenging.

• Taking a day off of 105 every week doesn’t hurt either!

The most useful I did was to do the reading before attending lecture. Neither the reading nor the lectures alone can teach you the material; I needed both. Although the readings weren’t always clear to me before lecture, they enabled me to keep up during lecture. Being completely lost for 75 minutes is no fun, and doing the readings in advance gave me the base knowledge I needed in order to tackle the harder concepts during lecture. Also, because the book can’t be resold, there is no reason not to mark it up. Highlighting not only helped me slow down and absorb more while initially doing the readings, it also made it easy for me to review the important sections before the midterm and final. Highlighting now helped streamline my studying later.

As for homework assignments, I started early and did the reading comprehension first. Although it was sometimes tempting to dive right into the bulk of the work (the programming and theory sections), the reading-comprehension questions helped me get moving in the right direction. And they were a good starting point for asking questions of the TA’s. Because some comprehension questions were multiple choice, I had to remind myself that just getting correct answers by process of elimination was beside the point: when I talked to TA’s until I fully understood the answers, I did much better. Really understanding reading-comprehension questions saved me hours of frustration on the programming and theory parts.

Unlike many other CS courses, COMP 105 is not something that you can complete by brute force. Before I could jump into the homework problems, I had to invest time in mastering the basics.