CSSE220 Academic Integrity Guide

1) Penalties for Violating Academic Integrity – they are severe. Possible penalties:
- Automatic F in the course (*Generally assigned for severe or repeated infractions*)
- Drop 1 letter grade
- -100% score on assignment (*Why less than 0%? because a student not completing an assignment earns a 0% and copying/sharing is worse than doing nothing at all*)
- 0% score on assignment
- partial credit on an assignment *(Generally assigned for honest students reporting mistakes)*

Yes, you can get an automatic F for cheating on one assignment one time.

2) The Kind of Assignment Matters
There are two kinds of assignments in CSSE220:
1. **Collaborative (any help is OK)**
2. **Individual (specific guidelines for acceptable help)**

Collaborative Assignments (anyone can provide any kind of help):
- *Daily Quizzes*
- *In-Class Exercises*

Individual Assignments (only help allowed is defined in #5 below):
- Coding Homework *(unless explicitly stated otherwise in the HWs instructions)*
- Design Problem Homework
- Exams - *NEVER communicate with anyone (other than an instructor) during or after the exam*

3) How to Get Help:
If you are stuck *we want* you to be able to get help when you need it! Good sources for help:
1. TA Help Hours - Posted on Moodle
2. Post a question on Piazza *(send to Instructors Only when posting code)*
3. Arrange a meeting with your instructor
4. **MUST CITE HELP from ANYONE or ANYTHING other than provided CSSE220 Materials**

4) Citing Help on Individual Assignments
- On any **Individual Assignment**, you MUST credit anyone or anything from which you receive help outside of CSSE220 Materials - place this citation near or at the top of what you submit.
- We will try to provide an appropriate place for you to provide this citation, but you are responsible to add a "Help Citation" comment if you do not find a provided location.
- If it turns out you got too much help, but you cited the sources that you got help from, it lets us assign one of the weaker penalties or possibly no penalty at all.
- If you end up **getting or receiving help** that is not cited, it is ***MUCH*** more likely to be penalized.

5) How to SAFELY HELP other students
We love to have students in CSSE220 help each other. To make sure they learn and you give appropriate help, be sure to:
1. **Never give help until AFTER completing an assignment!** In order to help another student, you should be sure to be competent in the material yourself, which is hard to determine until after you have successfully completed (submitted on Moodle) it yourself.
2. **NEVER share or show your own code!** If you have completed (submitted on Moodle) an assignment already it is OK for YOU TO HELP by looking at someone else's code, but only AFTER you have submitted your own code. However, it is NEVER OK to share or show your code.
3. Particularly when a student peer is having trouble with syntax (generally in weeks 1-3), it is OK to point out the problem to them and recommend a fix or resource to help them fix it.

4. **Make sure that they cite the help from you.** When you provide help make sure you have the student list you in their files to be submitted listing the ways that you helped them. You should write the comment together to make sure that you both agree with the nature of the help received.

6) Examples of Acceptable and Unacceptable Help:

6.1) Acceptable help
Examples of *acceptable* help from **3.4.a-c** of *Getting Help* above:
- *"So you loop across the array elements, getting each element and seeing if it's in the hashtable. If it is, you get the value and increment it…"*
- "*The assignment said we must use recursion and you are using a loop*"
- "*You said you are getting an array out of bounds exception, that usually means your 'for' loop is executing one too many times*"

6.2) Too much help?
Examples of too much (*unacceptable*) help for **3.4.a-c** above:
- Talking in code: *"So you write a for loop, 1 to array length. Your key variable is gonna be array[i]. You check if hashMap.get(key) is null, if not, you get the value with get, then hashMap.put(key, oldValue + 1)…"*
  Never "talk in code" to give or receive help
- "*I found the solution online and just copied and pasted right into Eclipse.*"
- "*You do the first two operations and I will do the last two.*"

6.3) NEVER List:
1. Never communicate BEFORE/DURING/AFTER exams about exams with other students.
2. Never show another student completed code or look at another student's solution code, ***even for a second***. (If you have a reason you want to share code after you and another student have BOTH submitted an assignment, first confirm with your instructor to get permission)
3. Never take and send photos of code or otherwise share files
4. Never code together with another student on an individual assignment
5. Never sit next to someone as you both work on the assignment, working through any problems you have as a group

7) Uh-oh, I think I messed up
What to do if you realize that you might have provided or received more help than you should have:
1. **Don't Panic!** If you are honest with us, the consequences can be greatly reduced if you are forthcoming rather than if we discover it ourselves
2. **Email your instructor** ASAP and explain what happened and how much of the assignment you were working on was impacted by the help you received
3. Depending on the specific instances, you might not receive any penalty or you may even be given a chance to do a replacement assessment
4. **Honesty is the best policy**, the more you tell us the more accommodating we can be

8) What about ChatGPT and other Generative AI Tools?
It is expected that any work submitted for assessment represents the intellectual work of the individual(s) submitting the work. Any attempt to pass off the intellectual work of another (including the work generated by Large Language Models (LLMs) like ChatGPT) as their own or without proper attribution is an example of academic misconduct and is subject to the penalties described in the Rose-Hulman Academic Rules and Procedures and Student Handbook documents.