

# Academic Integrity – CPSC 501

## Definition and References to University policies

As part of the University of Calgary, the Department of Computer Science wholeheartedly embraces the University of Calgary statement on Academic Integrity:

Academic integrity is the foundation of the development and acquisition of knowledge and is based on values of honesty, trust, responsibility, and respect. We expect members of our community to act with integrity.

Research integrity, ethics, and principles of conduct are key to academic integrity. Members of our campus community are required to abide by our institutional code of conduct and promote academic integrity in upholding the University of Calgary's reputation of excellence.

See <https://www.ucalgary.ca/pubs/calendar/current/k-3.html> for additional details. This document elaborates by providing tools to adhere to these principles in situations that are specific to computer science.

The [Student Success Centre](#) has excellent [resources](#) that can guide you to learning with integrity at the UofC. Especially read the [handbook](#) they have provided or attend the [workshops](#).

## Why is Academic Integrity Important?

The [handbook](#) explains why academic integrity is important. This includes

- Reputation of institution and credibility of degree. When you go for an interview, you want your Computer Science degree from the University of Calgary to be respected.
- Maximize your learning while at the University of Calgary. If the work you submit is not your own, we can't help you improve your knowledge and skills.
- Get in the habit of acting professionally. Others will not appreciate when you take credit for their work and companies are liable for copyright infringements.
- Your grades fairly represent your understanding and effort in your courses.

## How to Act with Academic Integrity in CPSC 501

**Exams:** These will be **out-of-class(Final)/in-class(Midterm)** **in-person individual assessments.** Students will complete this on their own as individual work. Students **will** be given exam preparation material and have a chance to ask questions about the exam topics before the exam. Students **will** be able to ask questions of the instructor or teaching assistants during the exam. Students **will not** be able to discuss the exam with other students during the exam. Exam aids, 'cheat sheets', or outside materials will not be allowed in exam unless

explicitly noted by instructor in detail. After each of midterm and final have been returned students are free to discuss their answers with other students and again with the instructor or teaching assistants.

**Assignments:** Each assignment is an individual assessment. Any developed material for the assignment should be maintained privately (for example private online git hosting) during the semester and not shared with others or shared publicly. The source for the assignment 1 refactoring project must be something that the student has the rights to use. Examples of this include a student's own prior project, a prior group project with other contributors agreeing, or an online source. Online sources for this project must have a license that allows for use, and students are not allowed to use an online project source for assignment one that was 'designed' for refactoring. Small parts of code for other three assignments can be sourced from online resources, but credit must be given (example from a tutorial about designing an I/O interface or networking). Larger parts of code, or parts that reproduce the fundamental assessed part of the assignment should not be sourced from others. If such fundamental parts are not the student's own work, then the grade will be reduced accordingly (including a possibility of a 0 grade if majority of work was not the student's work).

**General:** The overarching principle of academic integrity is giving credit where credit is due. If anyone had any input into a solution that you created, then that input should be acknowledged in work you submit. Not crediting anyone with contributions to your solution is equivalent to the claim that you created the solution alone without anyone contributing anything. If this claim is not valid, this is considered academic misconduct.

Contributions can include copying code or text from a website or textbook, having a friend, family member, or tutor helping you debug or design code you write, getting ideas from other sources, discussions with classmates, etc.

*Before using sources, make sure that these sources are allowed in the course you take.* In some courses, it is important that you show you can create solutions on your own. In other courses, it is important that you can collaborate with others or incorporate existing work in your own work. **[Sources are allowed for minor things but students will lose marks if they source something that was core to the assessment instead of completing the work themselves]**

All contributions and sources must be cited. Below you find some examples on how you can cite these. These are examples only. The instructor for your course may have a specific requirement for citations, so make sure to find the requirements for your course. In general, it is important that you provide enough information that we can find the source.

- Citing sources. You have likely learned how to cite sources in essays and papers in many other courses, so we'll focus here on citing sources of code you use.

*Example of citing code copied from an on-line source:*

```

# Minimum finding algorithm from
# https://www.geeksforgeeks.org/python-program-to-find-smallest-number-in-a-list/
# accessed May 16, 2022

# Assign first element as a minimum.
min1 = l[0]

for i in range(len(l)):

    # If the other element is min than first element
    if l[i] < min1:
        min1 = l[i] #It will change

```

*Example of citing code copied from a textbook* (this includes an e-textbook or a copy of (part of) the textbook found on-line.

```

/* Code for printing a formatted table from
 * Think Java by Downey and Mayfield
 * 2nd edition, Nov 29, chapter 6
 * accessed on-line through learning.oreilly.com
 */
for (int x = 1; x <= 10; x++) {
    for (int y = 1; y <= 10; y++) {
        System.out.printf("%4d", x * y);
    }
    System.out.println();
}

```

Some on-line texts include the page numbers from the paper based textbook. Make sure to include a page number if one is available.

- **As with all course work, resubmitting code that you developed for another course is generally not allowed.** However, in some courses this may be acceptable **[for assignment 1 previous code can be used as start of code base but not refactoring steps, for other assignments permission must be granted by instructor]**. Always make sure to check with the instructor before reusing your own code as part of a course submission. If it is allowed, don't forget to cite yourself. If you use code that you already submitted for a previous assignment, course, or other purpose, you are required to cite that!
- **Discussions and collaborations must also be acknowledged.** If you are working in a small team, you may informally mention that a specific colleague helped you. Getting help from others and then taking all the credit yourself will likely leave you in a place where no one wants to help you anymore.

In some courses we require that you work alone. Especially in the first year, we need to assess your ability to come up with solutions on your own and your readiness to start working in a team. In most courses, discussions and collaborations are allowed or encouraged. Make sure to check what the expectations are in each course you take.

In all cases, it is important that you cite such collaborations and make it clear which work was done individually and what was done in collaboration with others. Here are some examples on how you can cite discussions and collaborations.

```
// My mom is a software engineer and she explained how to use printf in Java.  
System.out.printf("For reading level for text '%s' is %d", userInput, readingLevel);
```

```
# User fabstudent on the course Discord explained this minimum finding algorithm  
# that I implemented here.
```

```
minSoFar = myList[0]  
for num in myList:  
    if num < minSoFar:  
        minSoFar = num
```

- Clarity in own work and work by others. In some courses, you will be required to work in teams. **[There are no team assessments in the course]** When submitting team work, if each of you focussed on distinct work, make sure it is clear what each team member contributed. You may also have played different roles on the team (as a whole or for different sections) for example, author, editor, reviewer, etc. Make it clear how each team member contributed. If this is not clarified, this implies that all team members worked collaboratively on all components.

In addition to crediting the work done by others, we have other expectations. It is important that you follow the guidelines and requirements set for each course and that you respect the work provided by the instructor. And that you follow the guidelines set by the instructor regarding work you complete for the course.

- **Similar to all courses on campus, all materials provided by the course instructor are for your use in the course only and should not be shared with others. This includes code or any other resources the instructor provides to help you complete course work. or code you write as a solution to assigned work.**

- Materials provided should never be uploaded to websites, even if this is required to get access to such websites. (Note that such websites often have a goal to get materials for free and financially benefit from such free materials that you paid a lot of money for as a student.)
  - Code that the instructor provided should never be shared. If you completed a project for a course and you want to include this code in your portfolio, either remove the instructor's work or ask for permission to share the work that includes the instructor's materials in your portfolio.
- Never make code (or other materials) you create for a course available to others while the course is in session. If you want to use a public resource to store and manage your work, such as GitHub, Google Docs, etc, make sure that this is stored privately: only you (or your team for team work) have access to it. If you want to make your solutions public (for example, to include in your portfolio), make sure to ask the course instructor for permission first.

## Challenges

It can at times be challenging to adhere to academic integrity when studying and completing course work. At times you may feel that you have too much to do, not enough time to do it, or the tasks may feel too challenging. You may be under a lot of pressure from parents or scholarship requirements to get high grades in a course. You may struggle to communicate with your teachers in a course.

During those times it will be tempting to let integrity slide and instead submit work that is not your own or try to hide that you got more help than you should have from others.

In the face of such challenges, it takes courage to act with integrity. The [handbook](#) provided by the Student Success Centre provides tools to help you continue to complete your course work with integrity. If you feel like it is not possible to meet your goals with integrity, please contact the course instructor or the department's assistant head for Academic Integrity. Contact information is available on the [department's website](#):

## Consequences of breach

Breaches of academic integrity are taken very seriously. See the [University of Calgary calendar](#) for the process.

Because the Faculty of Science is very large it uses a "pre-process" to reduce confusion and deal with things that turn out not to be academic misconduct at all, before this process actually starts. The "preprocess" goes all the way back to the marker or TA who first saw you work and noticed something unusual.

The marker or TA must inform the instructor of any (and all) evidence of possible academic misconduct - it's just part of the job - but the marker or TA does not do anything more than that.

The instructor looks at what has been noticed. If she/he can say, "Oh! I see what happened! That's not academic misconduct!" then things stop there.

Otherwise the instructor adds contextual information that helps people outside the course to understand the material, and passes it to an "Assistant Head, Academic Integrity" in the Department of Computer Science.

The Assistant Head then sends you a request for a meeting. During that meeting the Assistant Head will try to make sure that you understand this process. You will be shown the material suggesting that academic misconduct might have happened. You will have a chance to say what happened. If the Assistant Head can say, "Oh! I see what happened! That's not academic misconduct!", then you and the instructor are notified and your grade is restored.

If the Assistant Head is unable to say that then the information is forwarded to the office of the Dean of Science. The University of Calgary calendar describes the process from this point forward.