

Software Tools (BINF*6210) – Fall 2025 – Assignment #2
DUE DATE: Friday, Nov. 14, 2025, 11:59 PM Eastern

PLEASE INCLUDE THE FOLLOWING AT THE TOP OF YOUR ASSIGNMENT:

Your name: _____ Your Student ID: _____

Names of your group members:

Name of peer who was primary author of the script you edited: _____

Your GitHub link: _____

GitHub link of peer: _____

OVERVIEW

This assignment focuses on developing skills in collaboration, using a key software tool for collaborative coding and version control, and further improving your R programming skills. For this assignment, you will work in a group of 3-5. You will swap code from your Assignment #1 or new code that you are developing for your final assignment, work together to improve code, use GitHub to manage the code edits, and individually prepare a short write-up about the process. Each person's assignment submission is graded individually.

ASSIGNMENT-SPECIFIC LEARNING OUTCOMES

By the end of this assignment, you should be able to:

1. Use GitHub, which is an important software tool for collaboration and version control. You should be able to perform fundamental steps, including: fork, branch, commits, pull request, and merge. (GitHub also serves as an important code repository and record of your work. Once you develop one or more polished projects and set them to public, we suggest adding your GitHub link to your CV.)
2. Demonstrate an ability to read R code, interpret code, and constructively critique and edit code of others. This includes identifying areas where code efficiency can be improved and/or adding new features to the code.
3. Collaborate with others productively and respectfully.
4. Reflect upon your coding and analysis practices in bioinformatics and identify areas for your further skills development.

This assignment is primarily targeted towards our course-level learning outcome #3: *to conduct reproducible analyses and use software tools for version control and collaboration*. It is anticipated that, during the course of working on this assignment, you will also improve your skills in the other four learning outcomes as well, particularly programming skills.

INSTRUCTIONS

1. Respect! This is a group project. It is imperative that you treat your team members with respect, and you should expect to be treated with respect in turn. Interpersonal skills are vital for success in your professional life, and this is a great opportunity to work on these skills. Being respectful includes making the time to meet with your team and working on this project in a timely fashion. You need to give time for your team members to complete their “merge” and to write their reflective paragraph about their original script. As well, critiques and edits to the projects of others should be made in a constructive manner. Your comments/critiques should always be directed at the script/task and not at the person. (Example: “I have noticed a place in the code where we can reduce redundancy by using an apply function; what do you think of this edit?”. NOT: “I can’t believe you didn’t use an apply function in the first place; what were you thinking?” Very different tone! All of us, including course instructors, are learning new things all the time. Also, it’s important to remember that various class members have diverse prior knowledge to share. Some class members have extensive molecular biology background, while others have greater strengths in programming and/or statistics, for example. This is therefore an opportunity for reciprocal peer help. Recognizing and drawing from diverse skill sets is a valuable skill in itself, especially in an interdisciplinary field such as bioinformatics. Please treat this assignment and your peers as you would in a professional work assignment.

2. Formatting and originality. Please read all instructions carefully and save a copy of the instructions for your reference. However, do not include the learning outcomes and the instructions in your file for submission. The written portions should be written in your own original words. If you wish to do so, you may upload your assignment to CourseLink in advance and check your own TurnItIn score and locations of matches, then submit an update if needed. You should work together on the code improvements with your group members. Your grade will be an individual grade based upon your short report.

3. Submission instructions. Please submit your assignment in PDF format to the labeled Dropbox folder on CourseLink by the due date and time. Each person will submit an individual file with your portion of the assignment. This will consist of 5 written paragraphs, plus supporting evidence such as small code blocks of old vs. revised code and/or new figure(s) pasted in, as suitable. You may insert screen shots from GitHub for your supporting evidence. You may wish to work in Word or another word processor that accepts inserted figures and then convert to PDF for submission.

4. Clarifications regarding group work. This assignment involves group collaboration. You should work together with your group members on this assignment. Specifically, you should work together to discuss the scripts, and you are encouraged to work together directly on the code, e.g. using Zoom and screen sharing or an in-person group meeting if everyone can meet and feels comfortable meeting in person. The write-up portion must be written in your own words. If you wish, you may exchange write-ups

(except paragraph 5) for peer commenting/proofreading, but you must not write the assignment for another person or copy/paste prose from any source.

5. Choose one script per person. Each person will be the primary author of one script and the secondary contributor to a second script. You will choose either your Assignment #1 or new code that you are developing for your final assignment to share with your peers for collaboration. If you want to do so, it is permissible (but not expected) to make improvements (e.g. typo corrections or clarification of commenting) compared to your original assignment #1 submission prior to sharing with your group. However, extensive revision is NOT expected prior to sharing your script for the group project; timely sharing is expected. This assignment involves improving your group members' scripts collaboratively. The grade for Assignment #2 is based upon improvements to the original script and the associated write-up, not the quality of the original script. Upload your chosen script to a GitHub repository. Add your team mates as well as the TA to your GitHub. Note that even though you will be primary author of one script and make edits to one other script, your whole group of 3-5 should look at all of the scripts from your group together to discuss and share ideas for improvement.

6. GitHub process. Each student should “fork” the repository you will be editing, from the repository of the primary author of that script. i.e. You will be making a copy to your own repository. Then, “branch” the repository to create a development branch. Download the repository and work on the script. You should make regular updates to GitHub to the development branch as “commits”. Once you have discussed the proposed edits with your team, submit a “pull request” and “merge” these edits into your own master branch. Then, submit a “pull request” to the primary author. The primary author should consider/discuss the edits and then “merge” them into the master repository of the original script author. There will be an in-class tutorial to go over this process.

7. Attribution. It is essential that commenting be added to the top of each revised script with attribution to all contributors. A brief description should be added clarifying the role of each person in contributing to the script.

8. Instructions for each student in your role as secondary contributor. Each student: you should make three substantive edits (or a set of related edits) to the coding sections of the script. You should work on the code together with your group, but you will be the one to submit the edits to GitHub. Edits to improve the code could be of many types. These are just a few possible examples:

- make a true correction to the code so that it works properly, i.e. if the original version did not work as intended
- make an improvement to the data acquisition or data filtering steps to make the code more suitable to address the study question
- add a data summary step or a visualization to check the data for errors or biases
- increase the computational efficiency of the code or reduce code redundancy. Examples could include rewriting repetitive lines of code and instead using a vectorized function, a member of the apply family of functions, a loop, or writing a new function (as suitable to improve that particular section).

- edits to increase the generality and reusability of the code. An example would include using a function to get a vector length or to subset a data set, rather than hard coding in values that are specific to one data set.
- improvements to an existing visualization (e.g. improving axis labeling and using colour/symbols in a more effective way to show something useful). A figure edit could also involve switching from a base R plot to a ggplot2 plot to improve visual appeal, for example.
- adding a new visualization to the project. An example could include adding a map figure, if the original project lacked such a figure.
- Adding a statistical test. For example, perhaps the original project used a graphical/exploratory approach, and a statistical test could be added to test the idea.
- Add a taxonomic group and perform a comparison between the results using the original taxonomic group and a different taxonomic group. (I would recommend this one only after you consider all of the above options for code improvements first.)

When adding to making three main, substantive edits, you are also encouraged to make additional minor improvements throughout (e.g. improve the commenting or the formatting of the code, if needed). You will choose three main edits to highlight in your written document. You should paste in lines or small blocks of code, showing the old vs. new version of the code. You may use screen shots from GitHub if you wish. You are also requested to paste in the figure if your edit involves revision of a figure or creation of a new figure. Explain each chosen edit in a short paragraph written in your own words. Why did you make this edit? Why is this an improvement compared to the original version? You will have three written paragraphs for this part, each one describing one of your substantive edits to the script.

9. Reflection on your own original script. Each person will also write up a fourth paragraph as a response to the edits proposed to your own code. Do you think the proposed edits represent improvements to your script? What did you learn from the collaboration process overall? In what areas do you think you still need to improve? What will you work on going forward?

10. Short paragraph on teamwork (to be written up individually and confidentially, included at the end of your written assignment). Please provide a short paragraph summarizing your group work experience. Did all team members make themselves available to meet and discuss the assignment? Did all members of your group treat one another with respect? Did you contribute to your group to the best of your ability? Did your group members contribute to the best of their ability? Ensure that you comment upon the contributions of your team members. Please note that this course includes students from a variety of undergraduate programs. Variability in experience with molecular biology, biodiversity/ecology, statistics, and/or computer programming is expected. Your group should seek to benefit from the diverse expertise of your group. Was there any aspect of the group work that didn't go well? Overall, what did you learn from the teamwork process?

11. Summary: To be clear, your final assignment will consist of:

PDF file containing:

-At the top, include your name, student number, group members, and GitHub links.

- Three short paragraphs about your three main edits to someone else's script. Paste in small code blocks (and figures, if applicable) to support your points.
- One paragraph reflecting on the edits made to your own original script and what you learned during this process.
- One paragraph about your teamwork experience.

12. Enjoy! I hope that you have fun with this assignment and enjoy trying GitHub. I also hope that you enjoy and benefit from improving your coding and research skills by working together with your peers.

TIPS FOR SUCCESS

- Start this assignment right away! This assignment is intended to be worked on in small blocks of time a couple of times per week over 3 weeks. Do not leave this assignment to the last minute! That wouldn't be fair to your group members.
- Follow this timeline for this assignment:
 - Get a student GitHub account prior to Oct 28 (GitHub lesson). (Note that approval can take a few days, so start this process early.)
 - Individually, read over the Assignment #2 instructions before that class, and bring your questions to class. The instructor will assist in group formation. Then, you will work on the assignment over three weeks.
 - Week 1:
 - Group meeting #1: decide how you will swap scripts; get things set up in GitHub so that each person can work on their assigned script; share ideas you have for potential script improvements to consider.
 - After meeting #1, each person should work on making code improvements on their designated script.
 - Week 2:
 - Group meeting #2: share how things are going; show/discuss your edits to date; help one another with coding issues, as needed.
 - Each person works further on code improvements and drafts paragraphs 1-3 for the written assignment.
 - Week 3:
 - Each person should finish their edits and GitHub pull request at least three days before the assignment due date.
 - Group meeting #3: discuss any final issues and prepare for submission.
 - Each person completes final write up and submit individually by due date.
- We encourage groups to work together to resolve problems and help one another. However, if a group member is unavailable to complete their activities, then groups should contact the instructor to discuss the situation.
- Note: In terms of grading, remember that the grade is assigned based upon code improvements and the quality of the write-up, not the starting point for that script. Groups with mixed levels of experience will be fine. You should discuss ideas together as a group, and experienced programmers should especially take care to share ideas about how your own script can be improved.

- We hope that you have fun and enjoy learning with and from peers!
- This will also be a valuable experience for building your online professional profile. We encourage you to share one or more projects publicly on GitHub by the end of your program and list this on your resume or CV when applying for jobs.

Context for Assignment #3 Grading Rubric

The below grading rubric is intended to help you to be as successful as possible on Assignment #3. Instructor judgement will also be applied during grading. For example, high-quality or novel code edits not explicitly mentioned in this instructions document would also be considered during grading.

Note that each level is cumulative. That means you should meet the criteria of the lower level to move on to receiving a grade at the next level. Aim to complete all components at the first level (70%) first.

I have started the rubric at 70%, because I am assuming that everyone wants not only to pass this specific course (65% is the passing grade for an individual course at the graduate level at the University of Guelph). I assume you also want to do well overall, pass your graduate degree (at least 70% overall GPA required), and achieve a level of competency that will help you in your further studies and future career. You can do it! Your grade may fall in between two levels. This rubric is intended as a guide to help you to be successful.

I hope that you enjoy completing this assignment!

Assignment Component	Good (Grade of 70%)	Very Good (Grade of 80%)	Excellent to Outstanding (Grade of 90-100%)
Edits 1-3/ Paragraphs 1-3 (/20) X 3	14/20 *an edit is made that improves the code *edited code works as intended *a relevant statement is made about why this edit was made *writing is readily understandable	16/20 *edit involves applying programming skills (e.g. for iteration, generating a visualization, etc.) *written paragraph clearly explains why the edit represents an improvement to the script *writing flows well, few errors	18-20/20 *novel edit/addition is made that significantly improves the script *creativity is expressed *may be an advanced edit (e.g. advanced visualization, statistical test, writing a new function, etc.) *writing flows very well, very few to no errors
Reflection - Paragraph 4 /20	14/20 *reflection component is addressed *statements are included about whether the edits by your peer improved the script *statement about what you learned *writing is readily understandable	16/20 *reflection component is well addressed *thoughtful comments about the code edits to your script *reflection on your own practices and skills *reflection on areas you want to work on going forward *written paragraph flows well, few writing errors	18-20/20 *reflection component is very well addressed *critical reflection on the code editing process *thoughtful self-reflection on your progression and areas you would like to work on to meet your future goals *writing flows very well, very few to no errors

Teamwork – Paragraph 5 /20	14/20 *reflection included on teamwork process * comments included about your <u>and</u> your team members' contributions *your team members (in their paragraph 5) mention that you contributed to group *writing readily understandable	16/20 *teamwork reflection paragraph well developed *thoughtful comments included about your <u>and</u> your team members' contributions *your team members mention that you contributed well to group *writing flows well, few errors	18-20/20 *teamwork reflection paragraph very well developed *thoughtful comments about your <u>and</u> your team members' contributions *your team members mention that you contributed very well to the group *deep self-reflection on teamwork and collaboration processes; what considerations you will make moving forward in your graduate program and career *writing flows very well, very few to no errors
Total Graded out of 100% and Valued at 15% of Course Grade			