

CSCB07 – Software Design

Lab 2

Objectives

- Getting familiar with Java (loops, arrays, I/O)
- Getting familiar with Git (branching)

Logistics

- This lab is worth 2% of the course grade and it will be supervised by your TA during the tutorial session of Week 4 (Sep 25-29, 2023). If you encounter any problem while doing the steps listed in the next section, ask the TA for help.
- Attendance will be taken during the tutorial. If you are unable to attend, please send your TA an email explaining why and make sure to submit the deliverables by the due date. Failing to do so might result in a grade of zero on this lab.
- The lab should be done individually.
- The due date is Oct 1, 2023.

Instructions

1. Clone the repo of Lab 1 using the command **git clone RepoURL** (*RepoURL* should be replaced by the actual URL of your repo)
2. **cd b07lab1**
3. Create a branch named **lab2** using the command **git checkout -b lab2**
4. Modify *Polynomial.java* as follows:
 - a. Replace the array representing the coefficients by two arrays: one representing the non-zero coefficients (of type *double*) and another one representing the corresponding exponents (of type *int*). For example, the polynomial $6 - 2x + 5x^3$ would be represented using the arrays `[6, -2, 5]` and `[0, 1, 3]`
 - b. Update the existing methods accordingly
 - c. Add a method named **multiply** that takes one argument of type *Polynomial* and returns the polynomial resulting from multiplying the calling object and the argument. The resulting polynomial should not contain redundant exponents.
 - d. Add a constructor that takes one argument of type *File* and initializes the polynomial based on the contents of the file. You can assume that the file contains one line with no

whitespaces representing a valid polynomial. For example: the line `5-3x2+7x8` corresponds to the polynomial $5 - 3x^2 + 7x^8$

Hint: you might want to use the following methods: *split* of the *String* class, *parseInt* of the *Integer* class, and *parseDouble* of the *Double* class

- e. Add a method named **saveToFile** that takes one argument of type *String* representing a file name and saves the polynomial in textual format in the corresponding file (similar to the format used in part d)
 - f. You can add any supplementary methods you might find useful
5. Modify *Driver.java* to test the updated code
 6. **git add *.java**
 7. **git commit -m "modified Polynomial.java and Driver.java"**
 8. **git push --set-upstream origin lab2**
 9. Submit the URL of your repo to "Lab 2" on Quercus

Evaluation

- Git: 0.5 pt
- Java code: 1.5 pts
- A 10% penalty will be applied for missing the tutorial with no valid excuse