**Expectations**

Choose an open-source project and refactor the project with the following refactoring techniques.

**Set I**- **all** of the following techniques must be applied

* + Extract method - Duplicate Code
  + Long Method
  + Feature Envy
  + Switch Statements
  + Message Chains
  + Comments
  + Data Class
  + Rename method/variable - Alternative Classes with Different Interfaces & Comments

**Set II** - at least **three** kinds of refactoring techniques must be applied from the following list

have added two more refactorings in the set II of assignment 3 (

**push-down field/method /intermediate level – Rebellious Hierarchy,** [**Refused Bequest**](https://refactoring.guru/smells/refused-bequest)

**move field – Feature Envy**).

* + Move method – Feature Envy, Parallel Inheritance Hierarchies
  + Message Chains
  + Inappropriate Intimacy
  + Data Class
  + Pull-up variable/method –, [**Duplicate Code**](https://refactoring.guru/smells/duplicate-code)
  + Extract class – Multifaceted Abstraction, Large Class
  + Divergent Change
  + Data Clumps
  + Primitive Obsession
  + Temporary Field
  + Inappropriate Intimacy
  + RlpObjecttypeList
  + Change bidirectional association to unidirectional association – Cyclic Dependency, Cyclic Hierarchy & Cyclically-dependent Modularization
  + Replace conditional with polymorphism – Switch Statements

East student will identify an open-source project, fork and clone it, and make all the necessary changes. Once done, the student will raise a pull-request to the original repository. It is mandatory to raise a pull-request. If the refactoring carried out by the student is accepted by the repository owner/maintainer, the student will get bonus marks.

The student will carry out all the necessary changes to make the refactoring successful. It includes changes in the code, tests, and documentation.

**Hints**

* Check the guidelines published by the project to learn the mechanism of the contributions and to increase your chance of getting pull-request accepted (look for *contributions.md* or similar).
* Choose your project wisely. Consider important factors such as whether it is actively maintained and number of developers.
* Sometimes, project maintainers ask for additional changes to accept your initial changes. Be ready to make those changes.

**Constraints**

* Each student must work on a unique project. Hence, if you identify a Java project for yourself, make an entry in this[excel sheet](https://dalu.sharepoint.com/:x:/r/teams/CSCI5308Winter2022/Shared%20Documents/General/Assignment3_Projects.xlsx?d=w6333a620710e4b0db9c9f4860f73d521&csf=1&web=1&e=UoPowv)to claim the project. Obviously, if the project that you want to analyze is already claimed by someone else, you need to find another project.
* Your Java project must have at least 10 thousand lines of code and must have at least 50 stars on Github (hint: use GitHub's advanced search)
* It must compile.

**Delivery**

* The delivery mode is Brightspace; no need to put the code in Gitlab (rather provide us the links).
* Please include a short description (such as location (package/class/method, line no, where applicable) of each refactoring in the text file along with the link of the previous version of the file(s). You may follow this template:
  + Refactoring name
  + Location (file, class, method, ...)
  + Link of the files(s) of the previous commit (before refactoring)
  + Link of the files(s) of the commit with refactoring changes (after refactoring)
* Indicate the status of your pull request. If it is accepted, provide the link of the commit.
* At the time of deadline, the status of your pull request will be taken. If it gets accepted/rejected later, it has no influence (we will consider it is submitted but not accepted) on marks. Hence, start early; give some time to the project maintainers if you aim for the bonus marks.

**Rubric**

* *Prerequisite:*the program must compile. If program fails to compile due to any reason, the student will get 0.
* Each successfully applied refactoring in Set-I will attract 1 mark each.
* Each successfully applied refactoring in Set-II will attract 2 marks each (max 6).
* If a student shows that his/her pull-request is accepted by the project, the student will get 2 marks as bonus (assuming all the refactorings are accepted). If the student raises multiple pull-requests and only some of them are accepted, in that case, following rules apply.
  + Pull-request containing only Set-I refactorings are accepted = 0.5 bonus mark max
  + Pull-request containing Set-II refactorings are accepted = 0.5 bonus mark for each kind of refactoring (max 1.5 marks).

**Relevant links**

[Getting started with contributing to open source](https://stackoverflow.blog/2020/08/03/getting-started-with-contributing-to-open-source/)

[Basic etiquette for open source projects](https://developer.mozilla.org/en-US/docs/MDN/Contribute/Open_source_etiquette)