C S 272/463 Introduction to data structures
Fall 2019
Lab 1: Eclipse tutorial & file operations

1 Learning objectives

Objective 5 and Objective 7 in course syllabus.

- Objective 5: You will use this lab homework to practice your Java knowledge that you should have learned from your prerequisite courses. If you did not take any prerequisite courses, you may want to read this file (`https://www.cs.nmsu.edu/~hcao/teaching/cs272/JavaOverview2006.pdf`) to get an understanding about Java.
- Objective 7: You will need to learn how to use a new IDE, Eclipse.

2 Requirements

2.1 Task

- Get familiar with Eclipse IDE.
- Get familiar with Java file operations.

2.2 Detailed instructions for program design and implementation

1. (40 points) Get familiar with Eclipse IDE.
   Follow the steps in the Eclipse tutorial (`https://www.cs.nmsu.edu/~hcao/teaching/cs272/eclipsetutorial.html`) and load the two java files into eclipse, and test them.
   Create a file named `eclipse_test.txt` to keep running results.
   You are given two java files (`https://www.cs.nmsu.edu/~hcao/teaching/cs272/lab/lab1/Welcome.java` and (`https://www.cs.nmsu.edu/~hcao/teaching/cs272/lab/lab1/addition.java`)), download them and put them to the `(default package)` in Eclipse.
   (1) (20 points) For `welcome.java`, after `Line 8`, add one line to print the current system time. The function for getting the current system time is `System.currentTimeMillis()`
   Copy the running results to `eclipse_test.txt`.
   (2) (20 points) For `addition.java`,
   - Change the variable `number` to 20, copy the running results to `eclipse_test.txt`.
   - Keep the variable `number` as 10 and change `Line 14`
     `step++;`
     to
     `step *=2;`
     Copy the running results to `eclipse_test.txt`.
2. File operations.
   You are given a data file (`https://www.cs.nmsu.edu/~hcao/teaching/cs272/lab/lab1/core_dataset.csv`). This data file was obtained from `https://www.kaggle.com/rhuebner/human-resources-data-set`.
   It is a comma-separated values (CSV) file. The first row of the dataset is the metadata, showing what information of an employee is recored. Each row contains information of one employee. Download the file and have a quick look at the content.
   Create a java file named `EmployeeFileOp.java`. In this java file,
   (1) (25 points) Create a read function to read each row of the employee information and extract the the `Employee Name, Employee Number, State, Zip, Age`, and `Sex` information of each employee.

(2) (25 points) Create a write function to write the `Employee Name, Employee Number, State, Zip`, `Age`, and `Sex` information of an employee with age **less than or equal to 30** to a file named `young_employee.csv`. The `young_employee.csv` should keep the first row of the dataset.

(3) (10 points) Create a main function to properly call the read and write functions that you created.

For this question, you can use this file (`https://www.cs.nmsu.edu/~hcao/teaching/cs272/lab/lab1/FileOperator.java`) as reference. The `FileOperator.java` will need to use this data file (`https://www.cs.nmsu.edu/~hcao/teaching/cs272/lab/lab1/test.txt`).

3 Submission instructions

- Submit through canvas a zip file consisting of `eclipse_test.txt`, `EmployeeFileOp.java`, and `young_employee.csv`. Please do NOT submit the `.class` files.

4 Grading criteria

(1) The score allocation is already put in the questions.

(2) Please make sure that you test your code **thoroughly** by considering all possible test cases.

(3) 5 points will be deducted if submitted files (including files types, file names, etc.) do not follow the instructions.

(4) At least 20 points will be deducted if your code cannot be run on CS servers.