CS 420
HOMEWORK ASSIGNMENT H2

DUE DATE: Wednesday, September 19

In the following exercises, "displaying the current time" means displaying the time in milliseconds since start of program execution.

1. Simple Multi-Threaded programming - 4.

Write a Java application that runs three new threads (identified as `Thread1`, `Thread2` and `Thread3`). Two threads should sleep for 500 milliseconds before displaying the current time with their thread identifier on the terminal screen, while the third should sleep for 1300 milliseconds before displaying the current time with its thread identifier. Also, make the last statement in your application (the `main`() method) also display the current time with a label identifying it with the `main`() method. Note how closely do the sleeping periods match against the reported times and explain your observation. Also explain why the time printed by the `main`() method is located where you observe it.

2. Simple Multi-Threaded programming - 5.

Modify your solution for exercise 1 on homework 1 so that the three threads have different priorities (`Thread.MIN_PRIORITY`, `Thread.MAX_PRIORITY`, and `Thread.NORM_PRIORITY`). Note if this makes any difference in the correspondence of reported and actual times as compared to the version where all threads have the same priority and explain your observation. Print the numeric value of each thread's priority so the user can see what priorities are assigned to the threads.

3. Simple Multi-Threaded programming - 6.

Modify your solution for exercise 1 so that the `main()` method uses `join()` methods to delay the printing of the time at the end of the `main()` method until all its child threads have terminated.

General Instructions:

- Homework submissions must be prepared using computer document preparation applications such a word processor or similar editor. Handwritten solutions are not acceptable – neatness, readability and grammar count!
- Homework submissions will be clearly marked with the student's name, date and assignment identification at the top of the first page.
- All homework is to be completed by each student individually and represent that student's original, unassisted work. Any material copied in any way from other sources must be clearly identified and attributed.
- The non-programming problem solutions are printed on paper and submitted at the start of class on the due date.

Programming Instructions:

- Put a block of comments at the beginning of every physical file containing program source code that includes your name, the course name and number, information identifying what functions the program is designed to perform, and instructions how to execute the program. (required)

- For the programming problems, place the Java source files, class files, batch files and all other files necessary to execute each program you write into a separate Windows folder that is named `Prob1`, `Prob2`, etc.

- Place a batch file in each problem folder that will execute the program in that folder.

    - For example, The `prob1.bat` file is a text file with the following format (where `TheProgam` is the name of the class file containing the `main()` method):

      ```
      java TheProgam
      pause
      ```

    - Double-clicking the `prob1.bat` file should cause your program to execute.
    - Don't forget to put the `pause` command in the last line

- Place the programming problem folders into a zip file. Create the zip file so that the folder structure (path) is also recorded by selecting the "save full path info" option. Use your email ID as the zip file name. Example of file structure for submission:

  – Zip file named `brixiusn.zip` contains folders named `Prob1`, `Prob2 and Prob3`. Each folder has source code, class files and any required data files or batch files. The zip file records the path information for each file.

- Submit the zip file in the Canvas course site using the assignment submission capability. You can also add comments when you submit a file for the assignment.

- Each assignment must be submitted on Canvas by the start of class on the day the assignment is due.

- Submit only one zip file with your entire assignment.

- If you have already submitted a homework assignment and then decide you must resubmit the assignment before it is due, you can submit another zip file to replace previous submissions. You can also use comments with the submission to further explain your submission to the grader. You may resubmit as many times as you find necessary before the assignment due date.

- DO NOT FORGET TO ALSO SUBMIT THE PRINTED NON-PROGRAMMING PROBLEM SOLUTIONS AND SOURCE CODE AT THE START OF CLASS ON THE DUE DATE.