

Computer Science 435

Project J1: The Common Meeting Time Problem

Due: Mon. Sep. 26, at the beginning of class

Points: 50

The Common Meeting Time problem is the problem of finding a list of the times when a group of people are all available to meet. Write a concurrent program in Java to find the solution to the common meeting time problem for three people.

Input: The input to your program will be a file containing three lines of data. The name of the file will be specified as the first and only argument on the command line. Each line of the file will consist of an unsigned integer, followed by zero or more integers: the first number will tell you how many integers follow on that line. Each of the integers after the first represents a “time” when that person is available to meet. You can assume that no time is listed more than once in a list, and that there is a maximum of 1,000,000 values per line, but do not make any more assumptions about the input.

For example, suppose that the first person can meet at times 3, 2, 4, 1 and 5; the second can meet at 2, and the third can meet at 9, 13, 17, 6, 5, 4, 3 and 2. This information would be represented in the file like this:

```
5 3 2 4 1 5
1 2
8 9 13 17 6 5 4 3 2
```

Program Structure: Structure your program as follows. The main method should be responsible for reading the contents of the file into three lists. You may use either an array or an `ArrayList`, as you see fit. However, if you use an array, you probably want to wrap the array in a class along with the number of elements it contains.

The main method should create a thread for each value T in the first list. Communicate the value T to each thread when you create its associated `Runnable` object, or later using a mutator method, again as you see fit. Each thread is responsible for searching for its value in the second and third lists. If a matching value is found in both lists, a thread should (atomically) print out the string “T is a common meeting time.”.

If no thread succeeds in finding a common meeting time, your program should output the string “There is no common meeting time.”

What to turn in: When you have finished with the program, send an email message to me that contains, as an attachment, a copy of your source files, in tar format. When untarred, make sure your source files unpack in a directory named `<lastname_firstinitial>.tar/.tgz`.

Include in your email any special instructions that I will need to follow to untar, compile, and execute your program on the departmental Linux server sand.

Then, print a hard copy of the source to turn in. Make sure the main class file is on top. You must submit your source hard copy within one class day of your electronic submission to receive full credit. I will use the date of your electronic submission as the official submission time, unless you fail to turn in the required hard copy promptly.