Lab 08 – Multithreading

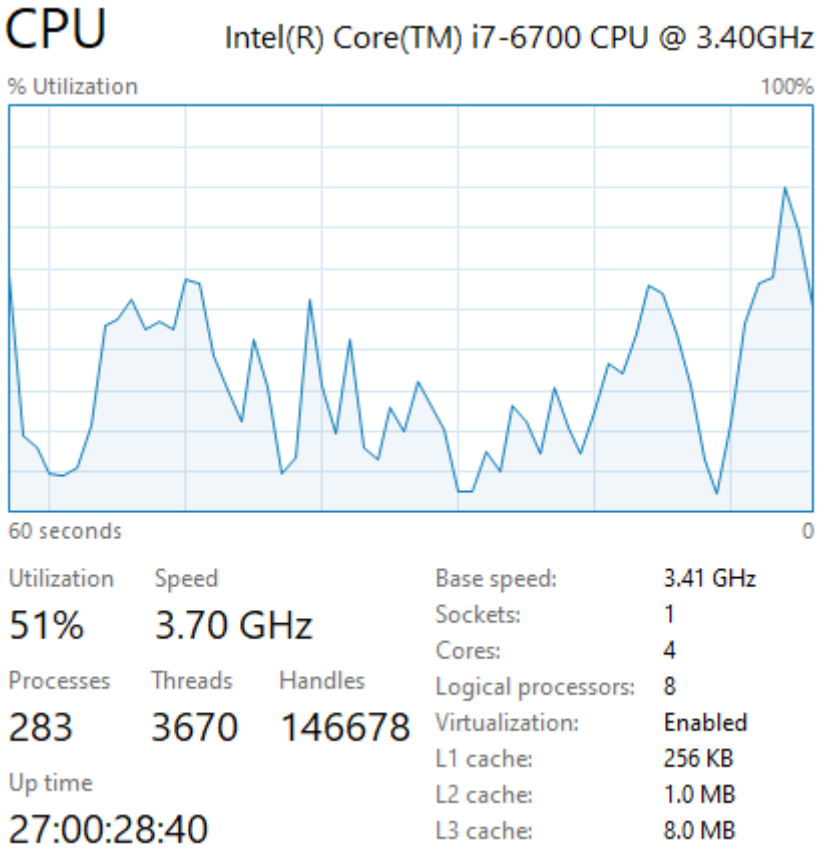
Objectives:

- Understand the concept of concurrency
- Create threads and assign tasks to threads to execute
- Recognize the advantages of using multiple threads
- Learn to use different types of thread pools

1. In Mac or Windows, you can find your number of available processors as follows:

   For Mac: Enter this command in a terminal: **`sysctl -n hw.ncpu`**

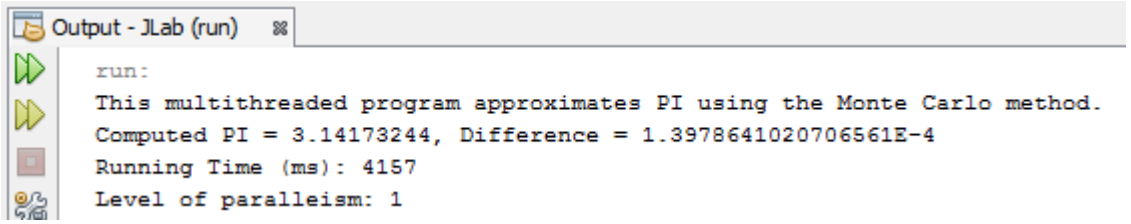
   For Windows: Press Ctrl + Shift + Esc to open Task Manager and then select the Performance tab to see how many cores and logical processors your PC has.

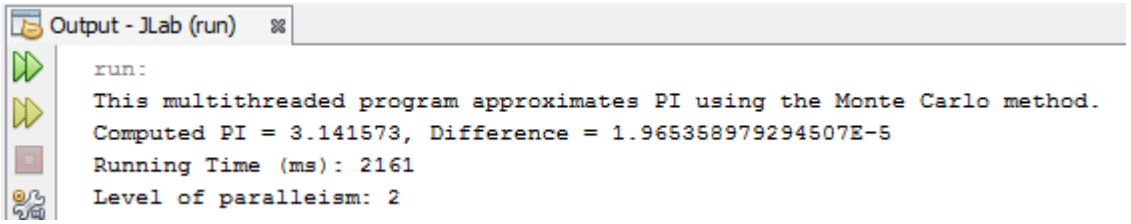


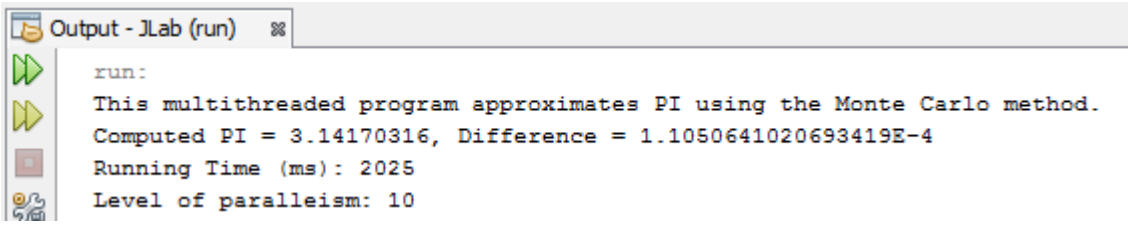
   Run the following Java method to obtain the number of processors in your system. Is the returned number indicating your physical cores or logical cores?

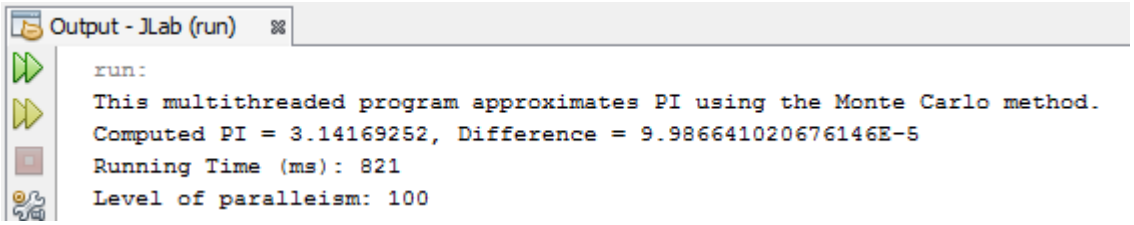
   `Runtime.getRuntime().availableProcessors()`

2. You have used the *Monte Carlo* approach to estimate the PI value in Lab 02. Now rewrite your simulation program to perform the same PI estimation using multiple threads. Here are the parameters for your simulation:
    - Number of iterations (random samples): 100,000,000
    - Level of parallelism (*p*): 1, 2, 10 and 100
      (divide your task into *p* subtasks and execute them on a pool of *p* threads)

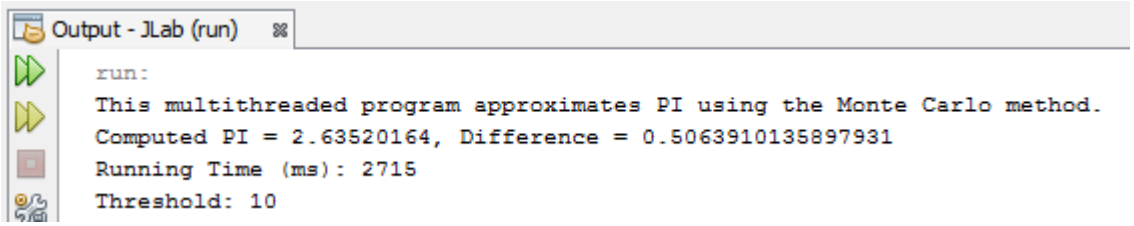


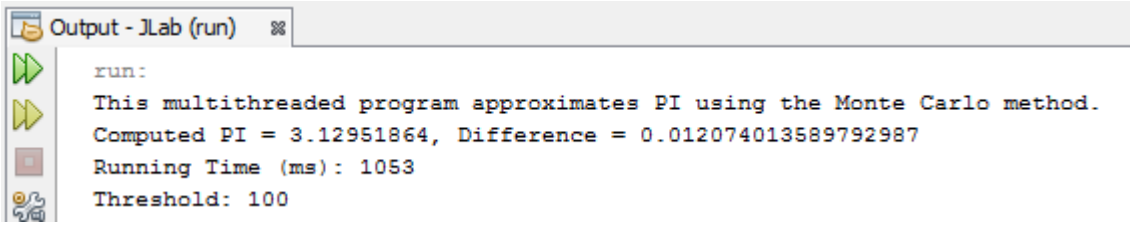


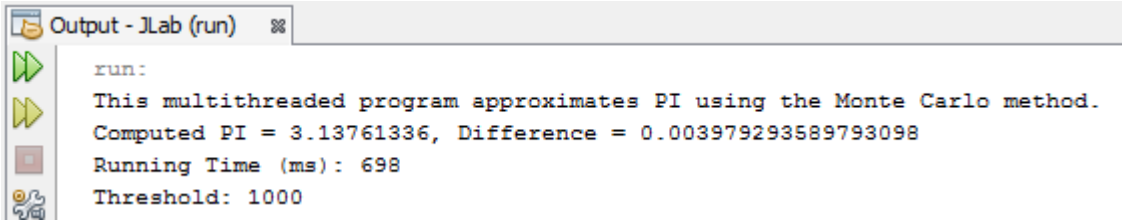


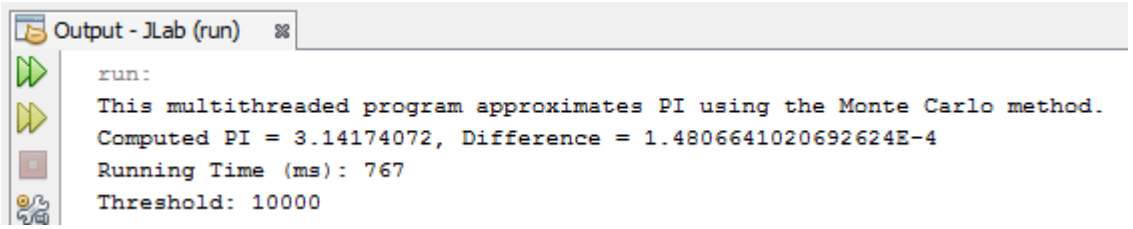


3. Revise your PI simulation program again by adopting the **Fork-Join framework**. Here are the parameters for your simulation:
    - Number of iterations (random samples): 100,000,000
    - Threshold for the subtask's size (*t*): 10, 100, 1000, 10000
      (recursively fork your task until the size reaches the threshold)









- END -