DEPARTMENT OF COMPUTER SCIENCE, HANYANG UNIVERSITY

(DOCUMENT VERSION 1.0)

# Wait-free Snapshot

**Concurrent Programming**

Programming Project #2

Final due date: Oct 12, 2022 (HARD DEADLINE)

## 1 TASK OVERVIEW

Implement the **Wait-free snapshot** algorithm that you have learned in the previous class.

## 2 TASK DETAILS

You have $N$ worker threads, and each thread has its own *local integer value*. The value can only be updated by the owner thread, but it can be read by other threads. This task is to implement the wait-free snapshot algorithm in the textbook.

1. Each thread keeps *updating* its local variable with a random value for *1 minute*. The

*update* should be accompanied by the *scan* function (making a snapshot) as the algorithm describes.

2. Parameterize the number of threads using the command line parameter.

3. Count the number of total updates (*throughput*) that all threads made.

4. Adjust the number of threads with 1, 2, 4, 8, 16, and 32, respectively, and analyze the performance difference. (The performance of your program does not have to be fast, but it must be correct.)

5. Write a report containing the analysis on your Wiki page of the hconnect project.

## 3 Test protocol

Your program should take one command-line argument. An example of the execution is:

| |
|---|
| $ ./run *N* |
| *N*: Number of worker threads |

After your program is finished, the number of total updates made by all worker threads needs to be printed on the standard output.

## 4 Submission

You should upload your project into **project2** directory of your “hconnect” repository. You must create a proper **Makefile** in the project2 directory. The name of your executable file must be **run**, and it must be in the project2 directory as well. You also need to upload your assignment report to your gitlab wiki page of your hconnect project. Please set the name of the wiki page as project2.

**Please enjoy this project and have fun!**