Final Project Technical Update

> To help final project development, implementation, and debugging, the TAs have provided some tools in this update. Meanwhile, the test case contribution format, compiling methods, and resource limitations have been slightly modified. Please read the following sections and the updated project description carefully and strictly follow the instructions.

Sample *main.cpp* File (*main1.cpp*, *main2.cpp*)

The sample main.cpp files are provided to you for testing and debugging. The file *main1.cpp* is for Problem 1, and the file *main2.cpp* is for Problem 2. Please refer to the **Compiler, C++ Version and Operating System** section in the updated final project description for compiling and execution commands.

You may modify the sample *main.cpp* files for development purpose. The actual grading program will have similar function calls and structure, but the TAs will include additional grading modules.

Test Case Validator (*testcase_validation.py*)

This tool is provided for you to check the test cases you designed are valid. You can run it with Python version >= 3.8. The TAs have tested the tool before release, however, we do not guarantee the correctness or completeness of the software. Passing *testcase_validation.py* checks could not be an excuse of an invalid test case submission. If you find any bugs, you are welcome to open a public discussion on the eeClass system.

You are encouraged to search online for detailed Python usage and intallation tutorials. Youtube tutorial videos and GeeksforGeeks explain Python better than the TAs.

CPU Benchmark Helper (*benchmark.cpp*)

This tool is provided for you to evaluate your CPU performance. Please compile this file using g++ without any optimization flags. Running the program takes 32.38 user CPU time on TAs' lab server. You could run it on your computer and use the result to estimate the CPU usage of your solution on TAs' lab server.

In this update, the TAs also set the time and space resource explicitly in section **Resource Limitation** of final project description. For each test case in Problem 1 and Problem 2, the maximum user CPU time is 5 seconds with 500MB memory usage. Please test your solution and make sure it can perform under these resource constraints in the worst case scenario.

Test Case Contribution Format

To simplify file input and output, the test case contribution format has been slightly modified. Please omit the commas previously required when representing sets. Moreover, please add a white space between brackets ('{', '}') and numbers. Here is a comparison of old and new format:

Old Format (Depricated)

```
5 7
1 2 10 5
1 3 20 8
2 3 15 6
2 4 25 10
3 4 30 12
3 5 15 6
4 5 20 8

insert 1 2 {1, 2, 3, 4, 5} 10
insert 2 3 {1, 2, 3, 4, 5} 5
insert 3 4 {1, 2, 3, 4, 5} 15
stop 2
insert 4 1 {1, 2, 3, 4, 5} 5
rearrange
```

New Format

```
5 7
1 2 10 5
1 3 20 8
2 3 15 6
2 4 25 10
3 4 30 12
3 5 15 6
4 5 20 8

insert 1 2 { 1 2 3 4 5 } 10
insert 2 3 { 1 2 3 4 5 } 5
insert 3 4 { 1 2 3 4 5 } 15
stop 2
insert 4 1 { 1 2 3 4 5 } 5
rearrange
```

Compile and Execution

To simplify development and testing, the TAs have modified the compiling and execution methods. After we unzip your submitted zip file, we will put the grading programs (*main1.cpp* and *main2.cpp*) into the same folder. The folder should look like this:

```
{student id}_final_project.zip
    ├ main1.cpp
    ├ main2.cpp
    ├ Problem1.cpp
    ├ Problem2.cpp
    ├ Problem1.h (Optional)
    ├ Problem2.h (Optional)
    ├ Problem1_test_case.txt
    ├ Problem2_test_case.txt
    └ Report.pdf
```

Then, the TAs will compile it with the following commands:

```
g++ -o main1 main1.cpp -std=c++2a
g++ -o main2 main2.cpp -std=c++2a
```

Finally, the TAs will execute *main1* and *main2* to grade your score. Please make sure you include all custom functions or constants in the *.cpp or *.h files.

Which TA Is Best to Answer Your Questions?

- TA Peter (陳唯中): The main designer of the final project and the head TA. You can ask/discuss with him about...
  - Algorithm and data structure suggestions
  - Solution ideas and possible approaches
  - Network and multicasting
  - CPU benchmark tool
- TA Kevin (彭冠文): The designer of grading and testing environments. You can ask/discuss with him about...
  - Problem 1 and Problem 2 public instance method parameters and return types
  - Sample main.cpp files
- TA Alice (簡映榕): The designer of grading and testing environments. You can ask/discuss with her about...
  - Problem 1 and Problem 2 public instance method parameters and return types
  - Z-score and project score calculations
  - Plagiarism checks
  - Test case validation tool