CSE100 Algorithm Design and Analysis
Lab Policies and Guidelines, Fall 2018
Last update: September 10, 2018

In each lab assignment, you will be asked to implement a specific algorithm that you have learned in class (in most cases). *You must follow the input/output formats* that are specified in each lab assignment description. We will provide an example in the description, but for more examples, please see `t*`, `o*` files under `testfiles` directory you can find after unzipping `labXX.zip`. Each `t`x file contains an input, and the corresponding output is in file `o`x.

Submission Before the posted deadline, submit your source code through the assignments page of CatCourses. *You must submit only your cpp file (which includes the main function) and the file name must be* ***`(your ucm email id).cpp`*** (i.e. `jsmith5.cpp`)— we apologize for this restriction, but we want to make our grading job as simple as possible. You will have about two weeks to complete your assignment from the release date. You're allowed to resubmit as many times as you want before the deadline, however, **NO LATE SUBMISSIONS** would be allowed afterwards.

Grading We will compile your code using `GNU C++` Compiler[1] against `C++ 2011` standard[2]. You can program your code in your favorite `IDE`, but make sure that you compile it using `GNU C++` compiler with `C++ 2011` standard and test it in the lab computers before submission; we will give you a testing toolkit.

We will run an *automatic* grader to test the correctness of your code. We will test it against number of examples, *some of which* will be shared with you. Your score will be proportional to the number of the test examples your code passes. If you do not see your score there within a week after your submission, please let your lab session TA know.

How to Test your Code by Yourself You will be provided with the same automatic grading tool we use. The only difference is that we will disclose only some examples while the entire test suit has many more. By checking your code with the grader before you submit, you will be able to know if your output format is correct.

Disclaimer. We have tested this script in a few different systems, but we can only guarantee that it works in the CSE100 lab computers, so ensure you test your code there before you submit it.

You need to use Linux Terminal, Mac Terminal (for Mac users), or Cygwin (for Windows users) with properly installed `GCC`. The recomended way is to use UCM Linux machine which is also available online using VPN and/or SSH. To test your code, unzip `labXX.zip` to your working directory and go inside the `labXX` folder. You will see several files. File named `Grader.sh` is the file you will use to test your code. After you finish writing your solution, compile it in terminal using:

```
$ g++ -std=c++11 -o a.exe source_file.cpp
```

Then you can run grader using:

```
$ ./Grader.sh
```

If you implementation is correct, you will see the following output:

[1] `https://gcc.gnu.org/`
[2] `https://en.wikipedia.org/wiki/C++11`

```
Test 1: correct.
Test 2: correct.
Test 3: correct.
Test 4: correct.
Test 5: correct.
Total correct: 5/5
```

If you want to test your code for just one test file, you can try:

```
$ ./a.exe < testfiles/t1
```

Troubleshooting and FAQ.

Q: `./Grader.sh: Permission denied`.
A: You need to change file permissons to be executable:
`$ chmod +x ./Grader.sh`

Q: `./Grader not found`.
A: We have updated the grader file, now it should have `.sh` ending, therefore use `./Grader.sh`

Q: I have changed my solution and uploaded new one to catcources, and system added a number to the file name. Is it all right?
A: Don't worry, just make sure you are entring right `UCM user name`, i.e. `jsmith5.cpp`, everything else will be handled automatically.

Q: My solution is exactly right, but grader says it is incorrect. What should I do?
A: Remove or add `endl`, make sure there is no additional outputs from your solution.

Academic Integrity You can discuss the lab assignment or your programming approaches with your classmates. However, your code must be your own work. We will run a tool to catch cheating. If you get caught, you will get 0 points for the lab assignment. If you get caught twice (in labs or exams), you will surely get an F. Note that both parties who showed their solutions and copied them will be punished.