# LEARNING BATCH COMPUTING ON SOONER

Henry Neeman, Director
OU Supercomputing Center for Education & Research (OSCER)
University of Oklahoma Information Technology

1. Log in to `sooner.oscer.ou.edu`, using Secure Shell (`ssh`) and the username and password that you've been given. Details for how to do this can be found at:

   **http://www.oscer.ou.edu/ssh_install.php**

   If the PC you're using has **Windows** as its operating system, then look for the description of how to download, install, run and configure PuTTY.

   If the PC you're using has **MacOS or Linux** as its operating system, then bring up a terminal window and at the Unix prompt enter the following command:

   ```
   ssh  -C  -X  -Y  yourusername@sooner.oscer.ou.edu
   ```

   except that `yourusername` will be replaced by your username.

2. If this is your **FIRST TIME** logging in to Sooner:

   a. Please **IMMEDIATELY** change your password. Details for how to do this are found at:

      **http://www.oscer.ou.edu/password_change.php**

   b. Please **IMMEDIATELY** do the following:

      ```
      echo youremailaddress@yourinstitution.edu > ~/.forward
      ```

   **NOTES**

   i. You should replace **`youremailaddress@yourinstitution.edu`** with your e-mail address.

   ii. After your e-mail address comes a blank space, then a greater than symbol, then a blank space, then tilde slash period **`forward`**, with no spaces between them.

3. The first time you start working on Sooner, copy the directory named `MPI_exercises` from Henry's home directory:

   ```
   cp  -r  ~hneeman/MPI_exercises  ~
   ```

   Note the tilde (~) at the end of this command. It means "my home directory" and is **EXTREMELY IMPORTANT**.

The rest of this document will apply to multiple times logging in to use Sooner, not just the first time.

4. Check to make sure that you’re in your home directory:

   ```
   pwd
   ```

   This is short for “print working directory.” The output should be something like this:

   ```
   /home/yourusername
   ```

   where `yourusername` will be replaced with your user name.

5. You should now have your own copy of the `MPI_exercises` directory, as a subdirectory of your home directory. Check to make sure that you do:

   ```
   ls
   ```

   Note that this command is lower case L followed by lower case S (that is, “ell ess” which is short for “list”), **NOT** “one ess.”

   You should see a list of one or more files and subdirectories, including `MPI_exercises`.

6. Change directory into your `MPI_exercises` directory, like this:

   ```
   cd  MPI_exercises
   ```

   This command shouldn’t produce any output, just the Unix prompt.

7. Make sure that you’re in your `MPI_exercises` directory, like this:

   ```
   pwd
   ```

8. List the files in your current working directory (`MPI_exercises`), like this:

   ```
   ls
   ```

9. Now change directory into your `Greetings` subdirectory, like so:

   ```
   cd  Greetings
   ```

10. Using the appropriate commands from above, make sure that you’re in your `Greetings` subdirectory, and that it has two subdirectories named `C` and `Fortran90`.

11. Now change directory into your `C` subdirectory or your `Fortran90` subdirectory, like so:

    ```
    cd  C
    ```

12. Using the appropriate commands from above, make sure that you’re in the appropriate subdirectory, and that it has multiple files in it.

13. One of the files in this subdirectory should be named `readme.txt`. Look at the contents of this text file:

    ```
    cat  readme.txt
    ```

14. Follow the directions in `readme.txt` to compile (make) your executables. The command to compile will be:

    ```
    make_cmd
    ```

    If for some reason that doesn't work, try this:

    ```
    ./make_cmd
    ```

    Notice the dot-slash `./` at the beginning of this latter version. It means "located in the current working directory" and is **EXTREMELY IMPORTANT**.

15. Using your preferred Unix text editor (for example, `nano`, `pico`, `vi`, `emacs`), edit the batch script file `hello_world.bsub`.

    In particular:

    a. Change every instance of `yourusername` to your user name.

    b. Change every instance of `youremailaddress@yourinstitution.edu` to your e-mail address.

16. While you're editing the batch script file, carefully read its contents.

    Notice that the batch script file has many comments (lines that begin with a pound sign `#` but **NOT** with `#BSUB`), which describe how to submit batch jobs to the batch queue, how to monitor batch jobs in the batch queue, and, if necessary, how to kill batch jobs in the batch queue.

    **NOTE**: You are **ABSOLUTELY FORBIDDEN** to run programs interactively; instead, you **MUST** run **ALL** programs in the batch queue.

17. Why can't you edit these files on your Windows PC, and then upload them to Sooner? See the last item in this document for the answer.

18. Once you're done editing and reading, submit the `hello_world.bsub` batch script, using the `bsub` command as described in the batch script file you just read. This will cause the batch scheduler to run your batch job, although you may have to wait a while before your batch job starts running.

19. Check the status of your batch job using the `bjobs` command, as described in the batch script file. You may need to check its status repeatedly.

20. You can check the contents of standard output (stdout) and standard error (stderr) using the `bpeek` command:

    ```
    bpeek  JOBID
    ```

    replacing `JOBID` with the appropriate batch job identifier that you saw from the `bjobs` command.

21. You should run the `bpeek` and/or `bjobs` commands repeatedly until your batch job is no longer listed, which will indicate that it has completed (not necessarily successfully).

22. Once the batch job finishes (which you'll know because it no longer shows up when you do the `bjobs` and/or `bpeek` commands), find out which of the files in your current working directory have been created most recently:

    ```
    ls  -ltr
    ```

    This command is lower case L, lower case S, space, hyphen, lower case L, lower case T, lower case R (that is, "ell ess space hyphen ell tee are"), meaning "list with long listing (lots of information), with files listed sorted by time, with the most recent at the bottom."

23. Examine the contents of the `stdout` and `stderr` files created by the most recently completed batch job. Is the output what you expected? Why or why not?

24. Repeat steps 15-23 for `greetings.bsub`.

25. You're welcome to play with the other example programs in the other subdirectories. You can go up a level in the directory tree by using this command:

    ```
    cd  ..
    ```

    Notice that this command is lower case C, lower case D (for "change directory"), followed by a space, followed by two periods ("dot dot") with no spaces between them, where "dot dot" means "the parent of the current working directory I'm in."

26. Once you've done this, you should check what directory you're in, and what files and subdirectories are in them, using the appropriate commands described above.

27. Why can't you edit these files on your Windows PC, and then upload them to Sooner?

    In principle, you can, but it's a **TERRIBLE IDEA**. Here's why:

    In Windows, almost all text editors embed hidden special characters (for formatting and so on) in the text file that you're editing, and also they express carriage returns differently than in Unix (including Linux).

    Furthermore, in some of these files (especially batch script files), **where** the carriage returns occur is **EXTREMELY IMPORTANT**, and the text editors in Windows cannot be relied on to keep those in the proper places.

    So, if you edit a file in Windows, there's no guarantee that, when you upload it to a Unix machine (including Linux), it'll be usable.

    There's a command that can fix some of these problems:

    ```
    dos2unix  filename
    ```

    This is a very handy command, but it can't fix everything, so you take **HUGE RISKS** if you choose to edit a file in Windows and then try to use it in Unix. So **DON'T**.