100 points. Individual Work Only. Due Feb 6, 2017 at 10 AM.

Objectives:

- use a Linux system and command-line tools
- use a software version control system
- submit jobs through a batch system
- review a technical paper (optional for CS 432, required for CS 632 and CS 732)

Instructions:

Part #1. Use a Linux system and command-line tools

Windows Users:

1. Download and install PuTTY – a free SSH client. Download this from: http://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html. Select the Windows Installer (putty-0.67-installer.msi), double click on the .msi file and follow the instructions to install in the directory of your choice. You can also create a shortcut on your desktop.
2. After you have successfully installed PuTTY, open the application, you will be prompted with a window to enter the Host Name, type *moat.cis.uab.edu*, and click on *Open* (you can also enter a session name, say CIS, under Saved Sessions and click on *Save*). You will be prompted to enter username and password, enter your blazerid and password. The first time you login you will see a popup, click OK. If successfully, you are now logged in to a CIS Linux system.

Mac/Linux Users:

1. Open the terminal application, you can find it under Applications -> Utilities -> Terminal.
2. Type *ssh <blazerid>@moat.cis.uab.edu*, make sure to replace *<blazerid>* with your individual *blazerid*. If this is your first login you will be prompted with the question "*Are you sure you want to continue connecting (yes/no)?*" type *yes* and then enter your password (blazerid password). If successful, you are now logged in to a CIS Linux system.

After successful login, you can now use Linux command-line tools to access various utilities and execute them. Here are couple of simple commands to try:

pwd – shows your current working directory

mkdir cs432 – creates a directory called "cs432"

cd cs432 – change directory to "cs432"

pwd – shows the new working directory

mkdir hw1 – creates a directory called "hw1"

cd – you are now back to the root of your home directory

pwd – shows the current working directory

ls –lR – recursively lists the files in long list format

Part #2. Compile and execute Homework-1 on a Linux system

1. Copy the solution to Homework-1 to the CIS Linux system. On Windows systems, you can use the psftp tools that came with the PuTTY installation or use any other file transfer program such as sftp or WinSCP or FileZilla.

   On Mac/Linux systems, you can use *scp* from the terminal to copy files to the CIS Linux system. For example: *scp hw1.c <blazerid>@moat.cis.uab.edu:cs432/hw1/* will copy the local file *hw1.c* to the CIS Linux system in the directory cs432/hw1/ (assuming you have already created these directories).
2. Login to the CIS Linux system and compile the solution to Homework-1 by typing "gcc –O –o hw1 hw1.c"
3. Execute the program by typing "./hw1 100"
4. If your program has any errors, fix that using a text editor such as *nano* or *vi*.

Part #3. Create a software repository to manage your source code

1. Login to the CIS git server at https://gitlab.cis.uab.edu using your blazerid and password.
2. Create a new group (say, *parallel_computing* or *cs432*) and add a new project (say, *hw1*) on the CS git server.
3. Login to the CIS Linux system and execute the following commands (assuming you have the solution as *hw1.c* in the directory *cs432/hw1* and the git group is *parallel_computing* and project name is *hw1*):

   ```
   cd cs432/hw1
   git init
   git remote add origin https://gitlab.cis.uab.edu/parallel_computing/hw1.git
   git add hw1.c
   git commit –m "Initial check in"
   git push -u origin master
   ```
4. After this if you make changes to the file hw1.c you can update the repository using:

   ```
   git commit -m "Appropriate commit message"
   git push origin master
   ```

The URL above (https://gitlab.cis.uab.edu/parallel_computing/hw1.git) will be different for you depending on your group and project names, you can get the URL by selecting HTTPS and clicking on the copy icon next to the URL on your project page on the git server.

Part #4: Login to the dmc cluster at Alabama Supercomputing Center (ASC) and submit jobs

1. Use an SSH client (as described in Part #1) and login to the *dmc* cluster (Host Name: dmc.asc.edu) at Alabama Supercomputing Center (ASC) using the username and password provided to you. First time you login, you will be prompted to change the password, please choose a secure password, and remember that password. Please review *Chapter 7. Accessing the Supercomputers* (pages 41–48) in HPC User Manual available on Canvas or the ASC website at https://www.asc.edu/files/HPC-User-Manual. If you are using Linux for the first time there is also a quick overview of using Linux systems in Chapter 8 (pages 60–89) of the user manual, please review that.
2. Edit the file *.bashrc.local.dmc* in the home directory and add "module load intel" to the end of that file (you have to use an editor such as *nano* or *vi* – see pages 80–87 of the user manual for more details). Logout and login again using the new password you selected.
3. Now that you have successfully logged in, check out the solution to homework-1 that you uploaded in Step #3, and compile it on the *dmc* cluster using the following commands:

   ```
   git clone https://gitlab.cis.uab.edu/parallel_computing/hw1.git
   cd hw1
   icc –O –o hw1 hw1.c
   ```

   The URL above will be different for you depending on your group and project names, you can get the URL by selecting HTTPS and clicking on the copy icon next to the URL on your project page on the git server.
4. Use an editor (e.g., *nano* or *vi* – see pages 80–87 of the user manual) and create a file called *myscript.sh* and add the following lines (assuming you named your executable *hw1*):

   ```
   #!/bin/bash
   ./hw1 1000
   ./hw1 1000
   ./hw1 1000
   ```

5. Change the file permission for myscript.sh to have execute permissions using the following command:

   ```
   chmod +x myscript.sh
   ```

6. Submit the job to the cluster using the following command:

   ```
   run_script myscript.sh
   ```

You will be prompted with several questions, follow the sample session provided below (the options will be slightly different based on the queue selected). Make sure you choose the "class" queue, 1 core, and the appropriate maximum runtime and memory based on the problem size you are using. Please do NOT execute programs on the login node, you MUST submit jobs to the batch system to execute your programs.

```
-bash-4.2$ run_script myscript.sh
This runs a script in the current directory via the queue system
Report problems and post questions to the HPC staff (hpc@asc.edu)


Choose a batch job queue:

Queue                      Wall Time      Mem  # Cores
-------------------------- ---------- -------- --------
express                      4:00:00     16gb      1-4
small                       60:00:00      4gb      1-8
medium                     150:00:00     16gb     1-16
large                      360:00:00    120gb     1-64
knl                         60:00:00     94gb    1-256
benchmark                   24:00:00    120gb     1-64
gpu                        360:00:00     20gb      1-2

Enter Queue Name (default <cr>: small)

Enter number of processor cores (default <cr>: 1 )

Enter Time Limit (default <cr>: 60:00:00 HH:MM:SS) 00:10:00

Enter memory limit (default <cr>: 1gb )

The job name will be myscriptshSCRIPT (can be changed in .asc_queue
file)

Run on; uv, dmc, ivy-bridge, etc (default: any)
dmc

============================================================
=====         Summary of your script     job           =====
============================================================
  The script file is: myscript.sh
  The time limit is 00:10:00 HH:MM:SS.
  The memory limit is: 1gb
  The job will start running after: 2017-01-22T22:47:40
  Job Name: myscriptshSCRIPT
  Virtual queue: small
  QOS: -p dmc,uv,knl --qos=small
  Constraints: --constraint=dmc
  Queue submit command:
sbatch -p dmc,uv,knl --qos=small -J myscriptshSCRIPT --begin=2017-01-
22T22:47:40 --requeue --mail-user=puri@cis.uab.edu -o
myscriptshSCRIPT.o%A -t 00:10:00 -N 1-1 -n 1 --mem-per-cpu=1000mb --
constraint=dmc

Submitted batch job 66889
```

7. You can check the status of your job using the “squeue” command. See the user’s manual for more details.

8. Once you are comfortable with using Linux and submitting the jobs, run the Game of Life program from Homework-1 on the dmc cluster and complete the table below.

Test Case #	Problem Size	Max. Generations	Time Taken HW1	Time Taken dmc cluster
1	1000x1000	1000		
2	5000x5000	1000		
3	5000x5000	5000		
4	10000x10000	1000		
5	10000x10000	10000		

Make sure you comment out any print statements you might have to print the board when you execute with larger problem sizes. Also, execute the program three times and use the average time taken. If your program creates large output files, please make sure you write the files to /scratch/$USER directory. You have to first create the /scratch/$USER directory using the command: mkdir /scratch/$USER.

9. Compare the timing results from Homework-1 with the execution time on the dmc cluster. If there is difference in the execution time between the two runs, provide a rationale for this difference.

Part #5: Review technical paper (optional for CS 432, required for CS 632 and 732)

Read the paper titled “Exponential Laws of Computing Growth” by Peter J. Denning and Ted G. Lewis published in the January 2017 issue of Communications of ACM available at Canvas and answer the following questions in your report:

1. What is Moore’s Law?
2. What is Dennard scaling?
3. What is S-curve model?
4. Is Moore’s Law dead now?
5. According to the authors, what are the three levels of exponential growth in the computing ecosystem? Will this growth be sustained for the next decade?

Report:

Include the results from Part #4 along with the performance analysis and answers to questions in Part #5 in the report. Submit the report as a Word or PDF file.

Submission:

Upload the report (.doc or .pdf file) to Canvas in the assignment submission section for this homework. There is NO need to turn in any printed copies in class.