Table of Contents

1. HW 1 Overview: Getting Started Using the Student Cluster
   - HW 1.1: Course Tools & Accounts
   - HW 1.2: Unix operations
   - HW 1.3: Install Pacheco Test Code Library
   - HW 1.4: Running Serial Programs on Student Cluster
   - HW 1.5: Timing Serial Programs

2. General Instructions
Problems 1.1 and 1.2 are simple, in the case of duplicate last names).

This will usually be done after the second day of class.

What to turn in: some image or evidence that you logged on: such as a listing of your directory on tuckoo
Due: 01/29/15

This homework involves demonstrating that you can log onto the student cluster. Tasks:

- access blackboard information for this class
- join Google Group and mailing list
- obtain a user account and logon on the student cluster
- create the correct homework directory structure
- perform simple unix operations to get you familiar with the system
- install the Pacheco demo codes into your directories
- compile, run, test, time a serial C program (histogram)
HW 1.1.a: Course Tools

**Description:** setting up and using required course tools.

- Access your Blackboard account
- Respond to the Google Group invitation.
- **What to ”turn in”:** image of BB access which will also appear in the ”last accessed” column; image of Google Group welcome email (keep these small (1/2 page is fine)).
Comment: Include images of your Blackboard login and an email from the Google Group.
HW 1.1.b: Accessing the student cluster

**Description:** obtain a user account, and logon using ssh.

- Once you are confirmed in the class, we will create an account using your last name (or using first name initials + last name in the case of duplicate last names).
- This will usually be done after the second day of class.
- **What to turn in:** some image or evidence that you logged on: such as a listing of your directory on tuckoo

image of any login such as the information below
HW 1.1.c: Getting cluster information:

Comment: solution is below

- \texttt{[gidget:~] mthomas\% nslookup rohan.sdsu.edu}
  \texttt{Server: 10.0.1.1}
  \texttt{Address: 10.0.1.1#53}

  \textbf{Non-authoritative answer:}
  \textbf{Name: rohan.sdsu.edu}
  \textbf{Address: 130.191.3.100}

- The cluster is on the internal SDSU campus network with no external login allowed. You can access the cluster from any on campus machine, including the ROHAN Academic Computing system, rohan.sdsu.edu.

- You create the account using your WebPortal account. See:
  - Home Page: \url{http://www-rohan.sdsu.edu/}
  - Create Rohan Account:
    \url{http://www-rohan.sdsu.edu/raccts.shtml}
HW 1.1.d: Remote SSH login

Comment: solution is below

- Launch **SSH** terminal on your computer
- **SSH** onto rohan:
  
  \%ssh rohanUserName@rohan.sdsu.edu

- **SSH** onto tuckoo:
  
  \%ssh tuckooUserName@tuckoo.sdsu.edu

- locate the class homework source code directory: /COMP605
HW 1.2.a.: Test these Unix operations:

- `cat /etc/motd`
  
  Note 1: try these from your home directory

- `whoami`
- `date`
- `uname -a`

- `cd ~`
- `pwd`

- `ls`, optional arguments `[-al, -R]`

- `mkdir`

- `chmod`, test arguments such as `[-R]`

Create, compile and run a serial "Hello user" program in C (where `user` is your username).

Compiler commands: use the specialized parallel library compiler commands

- C code: `mpicc -o myprogram myprogram.c`
- C code: `mpif90 -o myprogram myprogram.f90`
- where is the command installed?

  `%locate mpicc`

What to turn in: evidence that you completed this:
- images, text file which contains the output captured, session output.
HW 1.2.a. Comment: just run these commands and capture image

HW 1.2.a.: Test these Unix operations:

- `cat /etc/motd`
  
  Note 1: try these from your home directory

- `whoami`, `date`, `uname -a`

- `cd ~`, `pwd`

- `ls`, optional arguments `[-al, -R]`

- `mkdir`

- `chmod`, test arguments such as `[-R]`
HW 1.2.a. Comments: Include evidence you ran the code.

- Create, compile and run a serial "Hello user" program in C (where user is your username).

- What to turn in: evidence that you completed this: images, text file which contains the output captured, session output.

image the commands
### Directory Creation

**use '-p' option for multiple dirs**

```
[comp605tst] pwd
/home/mthomas/605/comp605tst [comp605tst] [comp605tst]
mkdir -p hw/hw1 [comp605tst]
[mthomas@tuckoo: /comp605tst] ls -al total 12 drwxrwxr-x 3 mthomas mthomas 4096 Feb 14 08:52 .
  drwx——— 5 mthomas mthomas 4096 Feb 14 09:06 ..
  drwx——— 3 mthomas mthomas 4096 Feb 14 08:52 hw
[mthomas@tuckoo: /comp605tst] ls -al hw/hw1 total 8
  drwx——— 2 mthomas mthomas 4096 Feb 14 08:52 .
  drwx——— 3 mthomas mthomas 4096 Feb 14 08:52 ..
```

### Change directory access:

```
[mthomas@tuckoo: /605] cd comp605tst/
[comp605tst] ls -al total 12 drwxrwxr-x 3 mthomas mthomas 4096 Feb 14 08:52 .
  drwx——— 5 mthomas mthomas 4096 Feb 14 09:06 ..
  drwx——— 3 mthomas mthomas 4096 Feb 14 08:52 hw
[comp605tst] chmod 750 hw
[comp605tst] ls -al hw/hw1 total 8
  drwx——— 2 mthomas mthomas 4096 Feb 14 08:52 .
  drwx——— 3 mthomas mthomas 4096 Feb 14 08:52 ..
```

/verbatim
HW 1.3.a: Install the Pacheco demo codes

Description:

- tar files are located in /COMP605/pacheco_examples
- you may want to read the Unix man pages or Web pages to learn about the commands for tar and gzip
- you will want to locate the histogram code in the IPP codebase (ch2)
- What to turn in: evidence that you completed this: directory listing.
HW 1.2.a. Comments: Directory manipulation

Directory Creation: use `-p` option for multiple dirs

```bash
[comp605tst] mkdir pacheco
[comp605tst] cd pacheco/
[comp605tst/pacheco] cp -R /COMP605/pacheco\_examples/* .
[comp605tst/pacheco] ls /COMP605/pacheco\_examples/
total 16
drwxr-xr-x 4 mthomas mthomas 4096 Feb 14 2014 .
drwxr-xr-x 14 mthomas mthomas 4096 Jan 20 02:43 ..
drwxr-xr-x 2 mthomas mthomas 4096 Feb 14 2014 intro-par-pgming-pacheco
drwxr-xr-x 2 mthomas mthomas 4096 Feb 14 2014 par-pgm-mpi-pacheco
[comp605tst/pacheco] ls /COMP605/pacheco\_examples/intro-par-pgming-pacheco/
total 872
drwxr-xr-x 2 mthomas mthomas 4096 Feb 14 2014 .
drwxr-xr-x 4 mthomas mthomas 4096 Feb 14 2014 ..
-rw-r--r-- 1 mthomas mthomas 880640 Feb 14 2014 ipp-source.tar
```

Change directory access: remember to set the homedir as well as hw dirst

```bash
[comp605tst/pacheco] ls -al
total 16
drwxrwxr-x 4 mthomas mthomas 4096 Feb 14 10:39 .
drwxrwxr-x 4 mthomas mthomas 4096 Feb 14 10:38 ..
drwxr-xr-x 2 mthomas mthomas 4096 Feb 14 10:39 intro-par-pgming-pacheco
drwxr-xr-x 2 mthomas mthomas 4096 Feb 14 10:39 par-pgm-mpi-pacheco
[comp605tst/pacheco] ls -al intro-par-pgming-pacheco/
total 872
drwxr-xr-x 2 mthomas mthomas 4096 Feb 14 10:39 .
drwxrwxr-x 4 mthomas mthomas 4096 Feb 14 10:39 ..
-rw-r--r-- 1 mthomas mthomas 880640 Feb 14 10:39 ipp-source.tar
```
HW 1.4.a: Compile & Run Serial Pacheco Histogram Code

- run the code for different variables:
  usage: 
  
  ```
  usage: ./histogram
  < bin_count >< min_meas >< max_meas >< data_count >
  ```

- use the following test cases (8 combinations):
  - `bin_count = [5, 20]`
  - `min_meas = 1`
  - `max_meas = [500, 3000]`
  - `data_count = [50, 500]`

- **What to turn in**: evidence that you completed this:
  - screen image, or text copy of output.
HW 1.4.b: Running the histogram program

[mthomas@tuckoo pacheco]$ cd intro-par-pgmProg-pacheco/
[mthomas@tuckoo intro-par-pgmProg-pacheco]$ ls
total 876
drwx------ 3 mthomas mthomas 4896 Mar 1 16:19 .
drwx------ 4 mthomas mthomas 4896 Mar 4 11:22 ..
drw------- 8 mthomas mthomas 4896 Feb 14 2014 ipp-source
-rwx------ 1 mthomas mthomas 88640 Oct 16 2012 ipp-source.tar
[mthomas@tuckoo intro-par-pgmProg-pacheco]$ cd ipp-source
[mthomas@tuckoo ipp-source]$ ls
total 52
Drwx------ 8 mthomas mthomas 4896 Feb 14 2014 .
Drwx------ 3 mthomas mthomas 4896 Mar 1 16:19 ..
Drwx------ 2 mthomas mthomas 4896 Jan 19 2011 ch2
drw------- 2 mthomas mthomas 4896 Apr 27 13:57 ch3
drw------- 2 mthomas mthomas 4896 Apr 23 13:48 ch4
drw------- 3 mthomas mthomas 4896 Nov 1 2012 ch5
drw------- 2 mthomas mthomas 4896 Jan 15 2012 ch6
-rwx------ 1 mthomas mthomas 14515 May 26 2011 INDEX
drw------- 7 mthomas mthomas 4896 Feb 14 2014 ipp-source
-rwx------ 1 mthomas mthomas 1694 Jan 7 2011 README
[mthomas@tuckoo ipp-source]$ cd ch2
[mthomas@tuckoo ch2]$ ls
total 28
Drwx------ 2 mthomas mthomas 4896 Jan 19 2011 .
Drwx------ 8 mthomas mthomas 4896 Feb 14 2014 ..
DRwx------ 1 mthomas mthomas 8638 Jan 19 2011 histogram.c
[mthomas@tuckoo ch2]$ mpicc -o histogram histogram.c
[mthomas@tuckoo ch2]$ ./histogram 10 1 1500 180
1.000-150.980: XXXXXXXXXX
150.980-300.800: XXXXXXXXXX
300.800-450.700: XXXXXXXXXX
450.700-600.600: XXXXXXXXXX
600.600-750.500: XXXXXXXXXX
750.500-900.400: XXXXXXXXXX
900.400-1050.300: XXXXXXXXXX
1050.300-1200.200: XXXXXXXXXX
1200.200-1350.100: XXXXXXXXXX
1350.100-1500.000: XXXXXXXXXX
[mthomas@tuckoo ch2]$
HW 1.4 Comments: Running Histogram

```
[[pachecho/histo] ./histogram 5 1 100 100
1.000-20.800: XXXXXXXXXXXXXXX
20.800-40.600: XXXXXXXXXXXXXXXXXXXX
40.600-60.400: XXXXXXXXXXXXXXXXXX
60.400-80.200: XXXXXXXXXXXXXXXXXXXXX
80.200-100.000: XXXXXXXXXXXXXXXXXXXXXXXXXX
```
HW 1.5: Timing the Histogram code

- Time how long the code takes to run as a function of Problem Size:
  \[ \text{ProbSize} = \text{data\_count} \]
  Wallclock Time: \( T_{\text{wall}} \)

- use the following test cases:
  - \( \text{bin\_count} = 10 \)
  - \( \text{min\_meas} = 1 \)
  - \( \text{max\_meas} = 50000 \)
  - Vary \( \text{data\_count} = 10^n \), where \( n = 0, 1, 2, N_{\text{max}} \)
  - What is \( N_{\text{max}} \)? Explain the limit.

- Modify how the code prints out the results: you don’t need to "plot" the histogram

- You only need to printout the bins, ranges, and the count (not all the X’s).

- Plot your test results using excel, Matlab, by hand: \( T_{\text{wall}} \) vs ProbSize
HW 1.5 Comments: Timing Histogram

[pachecho/histo] ./histodat 5 1 10000 10000
T_wall in seconds: 0.001017 seconds
T_getargs in seconds: 0.000019 seconds
T_mem in seconds: 0.000069 seconds
T_gendat in seconds: 0.000238 seconds
T_genbins in seconds: 0.000000 seconds
T_whichbin in seconds: 0.000691 seconds
Initialization Timein seconds: 0.000326 seconds
Sum Times in seconds: 0.001017 seconds
Suggestions for timing:

- Use C/Fortran internal timer
- You can find example timer code in `/COMP605/tools/code_timer.c`
- Avoid using the unix `date` function, it is not very sensitive
HW 1.5 Comments: Table of Serial Histogram Timing for critical blocks/calls to functions. Table has clear labels and uses scientific Notation. Note: it is missing time units

<table>
<thead>
<tr>
<th>Size</th>
<th>Bin gen</th>
<th>Data gen</th>
<th>Get arg</th>
<th>Mem alloc</th>
<th>Wall time</th>
<th>Which bin</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^0$</td>
<td>1e-07</td>
<td>3.2e-06</td>
<td>0</td>
<td>6.8e-06</td>
<td>1.13e-05</td>
<td>0</td>
</tr>
<tr>
<td>$10^1$</td>
<td>1e-07</td>
<td>3.1e-06</td>
<td>0</td>
<td>0</td>
<td>4.5e-06</td>
<td>8e-07</td>
</tr>
<tr>
<td>$10^2$</td>
<td>1e-07</td>
<td>5.1e-06</td>
<td>2e-07</td>
<td>3e-07</td>
<td>1.53e-05</td>
<td>9.5e-06</td>
</tr>
<tr>
<td>$10^3$</td>
<td>1e-07</td>
<td>2.53e-05</td>
<td>1e-07</td>
<td>5e-07</td>
<td>0.0001194</td>
<td>9.18e-05</td>
</tr>
<tr>
<td>$10^4$</td>
<td>4e-07</td>
<td>0.0001972</td>
<td>0</td>
<td>4e-07</td>
<td>0.001004</td>
<td>0.0008054</td>
</tr>
<tr>
<td>$10^5$</td>
<td>1e-07</td>
<td>0.0017872</td>
<td>2e-07</td>
<td>1.8e-06</td>
<td>0.0091417</td>
<td>0.0073517</td>
</tr>
<tr>
<td>$10^6$</td>
<td>3e-07</td>
<td>0.017798</td>
<td>9e-07</td>
<td>3.2e-06</td>
<td>0.09127</td>
<td>0.073466</td>
</tr>
<tr>
<td>$10^7$</td>
<td>3e-07</td>
<td>0.18205</td>
<td>1e-06</td>
<td>1.47e-05</td>
<td>0.91711</td>
<td>0.73504</td>
</tr>
<tr>
<td>$10^8$</td>
<td>2e-07</td>
<td>1.8158</td>
<td>1.3e-06</td>
<td>1.8e-05</td>
<td>9.1649</td>
<td>7.3492</td>
</tr>
</tbody>
</table>

**Table 1.** This table shows the average run time for critical blocks of the histogram.c. Each value is averages over 10 runs.

Source: Anna Ma, COMP605 student, Spring, 2016
HW 1.5 Comments: Prediction/Estimate of Serial Histogram Timing

To determine the max data size, we must first figure out where in the code is the largest memory cost. This occurs when we dynamically allocate memory for the variable data. This memory allocation costs the problem size times the memory needed for a single float value. Since a float value costs 4 bytes, and the available number of Gigs (which is obtained from the command df) is 77, which roughly translates to $7.7 \times 10^{10}$ bytes. Therefore, with simple computation we can solve for $n$:

$$10^n = \frac{(7.7 \times 10^{10})}{4}$$

$$n = \left\lfloor \frac{\log((7.7 \times 10^{10})/4)}{\log(10)} \right\rfloor$$

$$n = \lfloor 10.28 \rfloor$$

$$n = 10$$

Source: Anna Ma, COMP605 student, Spring, 2016
HW 1.5 Comments: Plot of Serial Histogram Timing for critical blocks/calls to functions. Plot shows a family of curves representing the different calls to functions.

![Plot showing serial histogram timing with various functions such as Bin generation, Data generation, Get arguments, Memory allocation, Wall clock, and Which bin.](data_count_vs_timing)

**Figure 6.** Plot of problem size vs critical time on log log axis (averaged over 10 runs).

Source: Anna Ma, COMP605 student, Spring, 2016
Put homework into a directory:

```
HOME/ < your_username > /hw/hw1
```

- include the source code(s), compiled binaries
- see sections above for what to include
- Write a simple report (this can be TEXT, Word, PDF Doc).
- See each section for what to turn in.
- Turn in hard copy (condensed/minimal number of pages) at start of class.

Once the submission timeline has closed

**DO NOT CHANGE THE FILE TIMESTAMPS!**
HW directory listing example

```
[mthomas@tuckoo hw_dir_exj3 ls -R]
.
  total 12
  drwx------ 3 mthomas mthomas 4096 Aug 28 12:23 .
  drwx------ 2 mthomas mthomas 4096 Aug 28 12:22 ..
  drwx------ 5 mthomas mthomas 4096 Aug 28 12:22 username

./username:
  total 20
  drwx------ 5 mthomas mthomas 4096 Aug 28 12:23 .
  drwx------ 3 mthomas mthomas 4096 Aug 28 12:23 ...
  drwx------ 2 mthomas mthomas 4096 Aug 28 12:22 dev
  drwx------ 5 mthomas mthomas 4096 Aug 28 12:22 hw
  drwx------ 2 mthomas mthomas 4096 Aug 28 12:23 misc

./username/dev:
  total 8
  drwx------ 2 mthomas mthomas 4096 Aug 28 12:22 .
  drwx------ 5 mthomas mthomas 4096 Aug 28 12:22 ..

./username/hw:
  total 20
  drwx------ 5 mthomas mthomas 4096 Aug 28 12:23 .
  drwx------ 5 mthomas mthomas 4096 Aug 28 12:23 ..
  drwx------ 2 mthomas mthomas 4096 Aug 28 12:24 hw1
  drwx------ 2 mthomas mthomas 4096 Aug 28 12:23 hw2
  drwx------ 2 mthomas mthomas 4096 Aug 28 12:23 hw3

./username/hw/hw1:
  total 8
  drwx------ 5 mthomas mthomas 4096 Aug 28 12:23 ...
  -rw-rw-r-- 1 mthomas mthomas 0 Aug 28 12:24 data1.dat
  -rw-rw-r-- 1 mthomas mthomas 0 Aug 28 12:24 file1.c
  -rw-rw-r-- 1 mthomas mthomas 0 Aug 28 12:24 file2.c

./username/hw/hw2:
  total 8
  drwx------ 2 mthomas mthomas 4096 Aug 28 12:23 .
  drwx------ 5 mthomas mthomas 4096 Aug 28 12:23 ..

./username/hw/hw3:
  total 8
  drwx------ 2 mthomas mthomas 4096 Aug 28 12:23 .
  drwx------ 5 mthomas mthomas 4096 Aug 28 12:23 ..

./username/misc:
  total 8
  drwx------ 2 mthomas mthomas 4096 Aug 28 12:23 .
  drwx------ 5 mthomas mthomas 4096 Aug 28 12:23 ..
[mthomas@tuckoo hw_dir_exj3]
```